

'Example: Abandon method

'This example script has not yet been created.

'Example: Abilities property

'This example script has not yet been created.

'Example: AbsoluteOn property

'This example script has not yet been created.

'Example: AbsoluteXPos property

'This example script has not yet been created.

'Example: AbsoluteYPos property

'This example script has not yet been created.

'Example: Accelerators property

'This example script has not yet been created.

'Example: AccessRights property

'This example script has not yet been created.

```
'Example: Activate method
' This example prompts for a document name then cycles through all open
' documents and attempts to find the one requested.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim DocName As String
DocName = Inputbox("Enter the name of the document to activate:")
Forall Doc In .Documents
    If Doc.Name = DocName Then
        Doc.Activate
        .ActiveDocWindow.Show
    End If
End Forall
```



```
'Example: ActiveDocument property
''This example retrieves the name of the currently active Word
Pro document,
'assigns it to a variable and prints the name in the Output panel
of the Script Editor.
'You must have a document open for this script to work.
'Paste this script into Sub Main in the Globals section.
Dim DocName as String
DocName = CurrentApplication.ActiveDocument.Name
Print DocName
```

'Example: ActiveDocWindow property  
'This example retrieves the text displayed in the title bar of  
the currently active Word Pro document's window.  
'It assigns that text to a variable and prints the text in the  
Output panel of the Script Editor.  
'You must have a document open for this script to work.  
'Paste this script into Sub Main in the Globals section.  
Dim DocWindowCaption as String  
DocWindowCaption = CurrentApplication.ActiveDocument.Name  
Print DocWindowCaption

'Example: Active property

'This example script has not yet been created.

```
'Example: ActualName property
Sub Main
Print "======"
Forall x In .division.foundry.paragraphstyles
    Print x.font.ActualName & " = " & x.font.size
End Forall
Forall x In .Division.foundry.paragraphstyles
    x.font.FontName = "Arial"
End Forall
Print "-----"
Forall x In .Division.foundry.paragraphstyles
    Print x.font.ActualName & " = " & x.font.size
End Forall
End Sub
```

```
'Example: AddAccelerators method
' This example assigns the 'AcceleratorTest' subroutine to the run every time ' the
Control and 1 keys are simultaneously pressed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
' The AcceleratorTest subroutine must also be included in your script module.
```

```
Const SHIFT = &H1000
Const CTRL = &H4000
Const ALT = &H2000
Dim FunctionName As String
Dim Key As Integer
.ActiveDocument.FullName
FunctionName = .ActiveDocument.FullName & "!AcceleratorTest"
Key = Asc("1")
.ApplicationWindow.Accelerators.AddAccelerators FunctionName, Key + CTRL, 0, True
```

```
Sub AcceleratorTest
MessageBox "Hello There"
End Sub
```

'Example: AddBookmark method

' This example creates a new bookmark named 'NewBookMark' in the active  
' division of the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MarkName as String

MarkName = .Mark(\$LwpMarkerTypeBookmark)

.Division.BookmarkManager.AddBookmark "NewBookMark", MarkName

'Example: AddChildToLayout method

'This example script has not yet been created.

'Example: AddDdeLink method

'This example script has not yet been created.



```
'Example: AddDivisionToPrint method
' This example prints the current division to the default printer.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim CurrentDivName As String
```

```
CurrentDivName = .Division.Name
```

```
.ActiveDocument.GetPageRange $LwpPresentationTypeLayout, 100
.ActiveDocument.PrintSettings.ClearDivisionList
.ActiveDocument.PrintSettings.AddDivisionToPrint CurrentDivName
.ActiveDocument.PrintSettings.SelectedPages = "1-9999"
.ActiveDocument.PrintSettings.PrintRange = $LwpPrintRangeSelectedDivisions
.ActiveDocument.PrintSettings.Copies = 1
.ActiveDocument.PrintSettings.PrintPagesFrom = 1
.ActiveDocument.PrintSettings.PrintPagesTo = 1
.ActiveDocument.PrintSettings.PrintPageType = $LwpPrintPageEvenAndOddPages
.ActiveDocument.PrintSettings.Collate = False
.ActiveDocument.PrintSettings.OutputToFile = False
```

```
'Example: AddDivision method
' This example creates two parent divisions each containing two child divisions.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Dim ParentDivName As String
Dim ChildDivName As String
Dim ParentDiv As String
ParentDivName = "Parent "
ChildDivName = "Child "

For ParentDivCount = 1 To 2
    ParentDiv = .ActiveDocument.AddDivision(ParentDivName & CStr(ParentDivCount))
    For ChildDivCount = 1 To 2
        .ActiveDocument.AddDivision ChildDivName & CStr(ChildDivCount), ParentDiv
    Next
Next
Next
```

```
'Example: AddEditorManager method
' This example adds a new editor with read only rights to the current
' document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim NewEditorName As String
Dim NewEditorInitials As String
```

```
NewEditorName = "Lotus User"
NewEditorInitials = "LU"
.ActiveDocument.EditorManager.AddEditorManager NewEditorName, NewEditorName
.ActiveDocument.EditorManager.Editors(NewEditorName).Abilities =
$LwpEditAbilEditingNotAllowed
```

'Example: AddEnvelopeReturnAddress property  
'This example script has not yet been created.

'Example: AddField method

' This example adds a new field named 'ExampleField' for the current document  
' where it is then inserted.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
.ActiveDocument.DocInfo.FieldManager.AddField "ExampleField", "Some data for  
ExampleField ", 1  
.InsertDocInfo $LwpDocVarField, "ExampleField"
```

```
'Example: AddIcon method
Dim IcnMgr As IconBarManager
Set IcnMgr = .ApplicationWindow.IconBarManager

' Select the icon to add.
' For this example to work, the icon and the script must already be linked
IcnMgr.SelectCustomIcon "c:\lotus\wordpro\icons\mynew.bmp","c:\lotus\wordpro\scripts\
mynew.lss"
' Add the icon to the Internet icon bar
IcnMgr.IconBars("InternetTools").AddIcon 2
' Set this bar to show in its context
IcnMgr.IconBars("InternetTools").ShowInContext = True
' This will force a redraw of IconBars
IcnMgr.ShowIconBars
```

'Example: AddIndexEntry method

'This example script has not yet been created.

'Example: AddItemIndex method

'This example script has not yet been created.



'Example: AddLayoutOverride method

'This example script has not yet been created.

'Example: AddOutlineSequenceItem method

'This example script has not yet been created.

'Example: AddPopupGraphicItem method

'This example script has not yet been created.

'Example: AddPopupPointSizeItem method

'This example script has not yet been created.

'Example: AddPopupTextItem method

'This example script has not yet been created.

'Example: Address1 property

'This example script has not yet been created.

'Example: Address2 property

'This example script has not yet been created.

'Example: AddSectionTabs method

' This example adds a quick division after the current division.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.SectionTabs.AddSectionTabs



```
'Example: AddSmartCorrect method
' This example adds a new entry to the SmartCorrect list
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Language = "English (United States)"
With .Application.SmartCorrects(Language)
    .AddSmartCorrect "lts", "Lotus Development"
End With
```

'Example: AddStringToList method

'This example script has not yet been written.

```
'Example: AddTOCEntry method
' This example adds the current sentence to the table of contents.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Dim TOCName As String
Dim TOCCommand As String

TOCName = .Text.GetText($LwpGetObjectTypeInfoSentence,False)
TOCCommand = "TOC 1" & """" & TOCName & """"
.AddTOCEntry TOCCommand
```

'Example: AddVerbMenu method

'This example script has not yet been created.

```
'Example: Add method
' This example prompts for a word to be added to the user dictionary.  The
' word is inserted into the current document, selected and then added to the
' user dictionary.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim NewWord As String
NewWord = Inputbox ("Enter a word to add to the user dictionary:", "Example Script", "")
If NewWord <> "" Then
    .Type NewWord
    .Text.MoveToStart $LwpLocationTypeWord
    .SelectWord
    .Text.Add $LwpAddTypeSpell
    MessageBox NewWord & " was added to the user dictionary.", MB_OK, "Example Script"
End If
```

```
'Example: AdjustShade method
' This example inserts 20 words into the current document and shades the
' last 5 words. After the message box is closed, 5 characters to the right
' of the insertion point are unshaded.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim WordNumber as Integer
For WordNumber = 1 To 20
    .Text.InsertText "Word" & Format$(WordNumber) & " "
Next
.Text.Shade $LwpLocationTypeWord,$LwpNavigateDirectionLeft,5
MessageBox "Click OK to adjust the shading.",MB_OK,"Example Script"
.Text.AdjustShade $LwpWhichSideLeft,5
```

'Example: Adopt method

'This example script has not yet been created.

'Example: AdviseOnRename method

'This example script has not yet been created.



'Example: AdviseOnSave method

'This example script has not yet been created.

'Example: Afid property

'This example script has not yet been created.

'Example: AlignmentChar property

'This example script has not yet been created.

'Example: AlignmentType property

'This example script has not yet been created.

'Example: Alignment property

'This example script has not yet been created.

'Example: AlignStyleName property

'This example script has not yet been created.

'Example: AllBorders property

'This example script has not yet been created.

'Example: AllowAlternateVerification property  
'This example script has not yet been created.



'Example: All property

'This example script has not yet been created.

'Example: AlternateName property

'This example script has not yet been created.

'Example: Always property

'This example script has not yet been created.

'Example: AmikakeName property

'This example script has not yet been created.

'Example: Amikake property

'This example script has not yet been created.

'Example: AmountOfSpaceAbove property

'This example script has not yet been created.

'Example: AmountOfSpaceBelow property

'This example script has not yet been created.

'Example: Amount property

'This example script has not yet been created.



'Example: AmtTether property

'This example script has not yet been created.

'Example: AmtToRotateContent property

'This example script has not yet been created.

'Example: AmtToTetherFrom property

'This example script has not yet been created.

'Example: Anchor method

' This example creates a frame and then anchors the frame 'In text'.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateFrame False, "Default Frame", 1440, 1440

.Frame.Anchor 0, \$LwpConditionTypeAllpages, \$LwpRelativeTypeLytInline

```
'Example: AnswerMsgBox method
' This example uses the AnswerMsgBox function to close the example
' message box.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.AnswerMsgBox $LwpMsgboxReplyOk
Messagebox "",MB_OK,"Example Script"
```

```
'Example: AnyEdits method
' This example prints the number of edits made by the current editor to the
' active document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim UserName As String
UserName = .Preferences.UserName
Print .ActiveDocument.AnyEdits(UserName)
```

'Example: AnyNumber property

'This example script has not yet been created.

'Example: AnyOleObjects property

'This example script has not yet been created.



'Example: AppendMacro method

'This example script has not yet been created.

'Example: AppFoundry property

'This example script has not yet been created.

```
'Example: ApplicationWindow property
''This example first prints the value of the Height property on
the ApplicationWindow object to the Output Panel.
'Then it sets Height to half of its previous value and prints the
new value to the Output Panel.
'You must have a document open for this script to work.
'Paste this script into Sub Main in the Globals section.
Print .ApplicationWindow.Height
.ApplicationWindow.Height = (.5 * .ApplicationWindow.Height)
Print .ApplicationWindow.Height
```

```
'Example: Application property
''This script gets the name of the currently active document by
going
through the Application property of whatever object has the
focus.
It prints the name to the Output panel in the Script Editor.
You must have a document open to run this script.
Paste this code example in Sub Main and run it.
DIM AppName As String
AppName = .Application.ActiveDocument.Name
Print AppName
```

'Example: ApplyAdjectivePos property

'This example script has not yet been created.

'Example: ApplyAdjectNounPart property

'This example script has not yet been created.

'Example: ApplyAgreementWithHereThere property  
'This example script has not yet been created.

'Example: ApplyAnglicisms property  
'This example script has not yet been created.



'Example: ApplyArchaicExpressions property  
'This example script has not yet been created.

'Example: ApplyArticleAgreement property  
'This example script has not yet been created.

'Example: ApplyBadComparatives property

'This example script has not yet been created.

'Example: ApplyBadInflection property

'This example script has not yet been created.

'Example: ApplyBadNounGender property

'This example script has not yet been created.

'Example: ApplyBadNoun property

'This example script has not yet been created.

'Example: ApplyBadPlural property

'This example script has not yet been created.

'Example: ApplyBadPrepositions property

'This example script has not yet been created.



'Example: ApplyBelgianExpression property  
'This example script has not yet been created.

'Example: ApplyBorrowedForeign property

'This example script has not yet been created.

'Example: ApplyBureuaJargon property

'This example script has not yet been created.

'Example: ApplyCalque property

'This example script has not yet been created.

'Example: ApplyCapitalizationCheck property  
'This example script has not yet been created.

'Example: ApplyClauseErrors property

'This example script has not yet been created.

'Example: ApplyCliches property

'This example script has not yet been created.

'Example: ApplyColloquialExpression property  
'This example script has not yet been created.



'Example: ApplyCommonlyConfusedWords property  
'This example script has not yet been created.

'Example: ApplyCommonMisspell property

'This example script has not yet been created.

'Example: ApplyComplexWords property

'This example script has not yet been created.

'Example: ApplyConfusedEasy property

'This example script has not yet been created.

'Example: ApplyConfusedEnglish property

'This example script has not yet been created.

'Example: ApplyConfusedHard property

'This example script has not yet been created.

'Example: ApplyConfusedMedium property

'This example script has not yet been created.

'Example: ApplyConfusedVerb property

'This example script has not yet been created.



'Example: ApplyConsecutiveNouns property

'This example script has not yet been created.

'Example: ApplyContractions property

'This example script has not yet been created.

'Example: Action property

'This example script has not yet been created.

'Example: Action property

'This example script has not yet been created.

## Word Pro: Abandon method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCO  
NTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ABANDON_METHOD_EXSCRIPT',1)} See example
```

Converts the child container of a non-page container to the child container of the current page container.

## Syntax

```
[objectreference].Abandon()
```

## Parameters

## Return value

Boolean

## Usage

An example of how to use the Abandon method can be seen in the following scenario: On a page is a parent frame that contains a child frame. When you move the parent frame across the page, the child frame anchored to the parent frame moves with it. You now want to detach the child frame from the parent frame and anchor it instead to the current page, so that the child frame moves across the page by itself. To accomplish this, you set the Abandon method to convert the child frame from a child of its parent frame to a child of the current page container.

Equivalent to choosing Frame Properties, clicking the Placement tab, and selecting "On current page" from the "Place frame" box. Note that the Frame menu displays when the insertion point is in a frame.

### **Word Pro: Activate method**

{button ,AL('H\_DOCUMENT\_CLASS;H\_SECTIONTABS\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACTIVATE\_METHOD\_EXSCRIPT',1)} [See example](#)

[SectionTabs]

Causes an OLE object to become active in the Word Pro application.

### **Syntax**

[objectreference].Activate()

### **Parameters**

### **Return value**

Returns an Integer value indicating success (True) or failure (False). The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

[Document ]

Use this method from a TextDocument object to make that object active.

## Word Pro: AddAccelerators method

{button ,AL('H\_ACCELERATORS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDACCELERATORS\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds accelerator key commands that implement Word Pro functions and commands.

### Syntax

[objectreference].AddAccelerators(MacroName, Key, [Id],[IsTemporaryUse])

### Parameters

#### *Macroname*

A String expression representing the name of the macro, including the file name that should be assigned to a keystroke.

#### *Key*

A Numeric expression representing the specific key used as the shortcut key combination. Data type is Integer.

#### *Id*

A Numeric expression representing the identification number of the menu item to which you want to add the accelerator object. Data type is Integer. Default is 0.

#### *IsTemporaryUse*

A Boolean expression indicating whether this accelerator will persist between sessions of Word Pro (False) or will be discarded when the current session of Word Pro is terminated (True). Default is False.

### Return value

Integer

### Usage

## Word Pro: AddBookmark method

{button ,AL('H\_BOOKMARKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDBOOKMARK\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a bookmark object to the document. Equivalent to choosing Create - Bookmark. The bookmark must first be created by creating a new marker of type Bookmark. After the marker is created, the AddBookmark method notifies the BookmarkManager about the newly created bookmark.

### Syntax

[objectreference].AddBookmark(Name,MarkerName)

### Parameters

#### *Name*

A String expression representing the bookmark object you want to add; user-defined. If you create a duplicate name, a number is added to the end of the original name.

#### *MarkerName*

The String name of the bookmark object marker. You must create a marker before you create a bookmark. You can name the bookmark yourself or use the name provided by Word Pro.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage



## Word Pro: AddChildToLayout method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ADDCHILDTOLAYOUT_METHOD_EXSCRIPT',1)} See example
```

Assigns a parent layout object a child layout object.

### Syntax

```
[objectreference].AddChildToLayout(ChildName)
```

### Parameters

*ChildName*

A String expression that represents the name of child object you want to add to the layout.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: AddDdeLink method

{button ,AL('H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDDDELINK\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a Dde link object to the document. Word Pro creates a Dde link if you use Paste Special, but only if OLE fails.

### Syntax

```
[objectreference].AddDdeLink(ConversationHandle,LinkInfo,MarkerName,ServerName,ClipbrdFormatName,UpdateDataOnly)
```

### Parameters

#### *ConversationHandle*

A Numeric expression that was added manually, representing the conversation handle used by this method. If added using LotusScript, this value is always 0. If added internally, the value may be non-0. Data type is Long. Required parameter.

#### *LinkInfo*

A String expression representing the link information about the Dde link object you want to add. Consists of the server name, the topic name, and the item name. Required parameter.

#### *MarkerName*

The String name of the Dde link object. You must create a marker before you create a Ddelink. You can name it yourself or use the name provided by Word Pro.

#### *ServerName*

The String expression representing the executable name of the server to which you want to link.

#### *TopicName*

The String expression representing the name of the drive, directory, and name for the file that contains the data or the object name of the data.

#### *ItemName*

The String expression representing the location or name for the data, such as a range of cells, a named spreadsheet range, or a bookmark name.

#### *ClipbrdFormatName*

The String expression representing the name of the format used by the Clipboard. The format will be used to read/interpret/import the data.

#### *UpdateDataOnly*

A Boolean expression that specifies whether the new DDE link will update. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). This value specifies whether you can use formatting from the server application or from Word Pro. If the value is True, the formatting is Word Pro.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

In Word Pro, it is difficult to create a Dde link through the user interface because OLE is always tried first. You can also create a link through LotusScript, some Ami Pro documents, or other applications such as WordPerfect.

**Word Pro: AddDivisionToPrint method**

{button ,AL(^H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ADDDIVISIONTOPRINT\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to specify the name of a division object that you want to print.

**Syntax**

[objectreference].AddDivisionToPrint(DivisionToPrint)

**Parameters**

*DivisionToPrint*

A String expression that allows you to print a specified division object.

**Return value**

Integer

**Usage**

Use this method to add a division object to a list of divisions. You can locate a list of divisions by choosing File - Print, clicking Select Pages and selecting "Whole divisions" in the Select Pages dialog box.

## Word Pro: AddDivision method

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a division object to a division or text document object. Equivalent to choosing Create - Division.

### Syntax

Division

[objectreference].AddDivision(NewName, [ParentName,] [BeforeNeighbor,][NeighborName])

TextDocument

[objectreference].AddDivision(NewName, [ParentName,] [BeforeNeighbor,][NeighborName])

### Parameters

*NewName*

A String expression that represents the name of the new division object.

*ParentName*

A String expression representing the name of the parent division object.

*BeforeNeighbor*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). This parameter allows you to specify whether you want to locate the new division object before its neighbor division object. Default is False (0).

*NeighborName*

A String expression representing the name of a neighbor division object.

### Return value

### Usage

## **Word Pro: AddEditorManager method**

{button ,AL('H\_EDITORMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDEDITORMANAGER\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a new editor to a document.

### **Syntax**

[objectreference].AddEditorManager(EditorName, EditorInitials)

### **Parameters**

*EditorName*

A String expression representing the name of the assigned editor.

*EditorInitials*

A String expression representing the initials of the assigned editor.

### **Return value**

The return value for this method is always -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

This method returns True if Word Pro adds a new editor to the document. This method returns False if Word Pro does not add a new editor to the document, or if the specific editor name already exists.

### **Usage**

Allows you to add a new editor to a document and assign the editor default editing rights as defined by the special "All others" editor. You can only add a new editor to a document using this method, not an editor manager.

## Word Pro: AddField method

{button ,AL('H\_DOCINFOFIELDMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDFIELD\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a document field object in a document. Equivalent to choosing File - Document Properties, choosing Document, clicking the Fields tab, and clicking New.

### Syntax

[objectreference].AddField(FieldName,Contents, ExportFieldToNotes)

### Parameters

#### *FieldName*

A String expression representing the name of the document field you want to add.

#### *Contents*

A String expression representing the contents that will be contained in the document field you want to add.

#### *ExportFieldToNotes*

A Boolean expression specifying whether or not you want the field exported to Notes. A Boolean expression is either True or False.

### Return value

The return value for this method is always -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Creating a new document field object as part of the document information can be useful when you want to track specific information. For example, you could develop a system to track documents with specific clients. First, you could create a document field called "Client" for the document and assign a client name as its contents. You can then insert the document field into your text stream, so that the client's name would appear in the text of the document.

**Word Pro: AddIcon method**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDICON\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds an icon to an icon bar object.

**Syntax**

[objectreference].AddIcon(Position)

**Parameters***ParameterName*

Data type is Integer. Parameter is the user-defined location on the icon on the bar. Required parameter. The legal values for this parameter will always be -1 or 0 but you can use the LotusScript constants of True (-1) and False (0).

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Use this method to add an icon to an existing icon pallet. The position parameter is a base 0; thus, the first position is position zero, the second position is position one, and so on.

To add an icon, you must first select it using the SelectCustomIcon or SelectStandardIcon method in the IconBarManager class. The Word Pro user interface is in the SmartIcons Setup dialog box where you can drag an icon from the available list and drop it on the icon bar set.

## **Word Pro: AddIndexEntry method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ADDINDEXENTRY\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds an index entry for an entire document, for a section, division, or selected text in a document. Only applies to an existing index.

### **Syntax**

[objectreference].AddIndexEntry(IndexEntry)

### **Parameters**

*IndexEntry*

A String expression representing the name of the entry you want to add to the index.

### **Return value**

### **Usage**

Equivalent to choosing Text - Mark Text As - Index Entry, selecting the desired text, and clicking Mark.



## **Word Pro: AddItemIndex method**

{button ,AL('ERROR:Entrynotfoundinindex',0)} [See list of classes](#)

{button ,AL('H\_ADDITEMINDEX\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds an item in the Order Groups dialog box located in a custom dialog box object. Equivalent to choosing Edit - Order Items within the Word Pro Dialog Editor.

### **Syntax**

[objectreference].AddItemIndex(item,index)

### **Parameters**

#### *Item*

A String expression representing the name of the item you want to add to the combo box object or the list box object.

#### *Index*

A Numeric expression representing the index value for the list item named in the item parameter. You must use an Integer as the numeric expression.

### **Return value**

### **Usage**

## **Word Pro: AddLayoutOverride method**

{button ,AL(`H\_LAYOUTOVERRIDE\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ADDLAYOUTOVERRIDE\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds an overridden layout method to a layout object.

### **Syntax**

[objectreference].AddLayoutOverride(LayoutObjectName)

### **Parameters**

*LayoutObjectName*

A String expression representing the name of the layout object you want to override.

### **Return value**

### **Usage**

## **Word Pro: AddOutlineSequenceItem method**

{button ,AL('H\_OUTLINESTYLESEQUENCE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDOUTLINESEQUENCEITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds an item to an outline style sequence object.

### **Syntax**

[objectreference].AddOutlineSequenceItem(StyleName)

### **Parameters**

*StyleName*

A String expression representing the style name of the outline sequence item that you want to add to an outline style sequence object.

### **Return value**

### **Usage**

## **Word Pro: AddPopupGraphicItem method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDPOPUPGRAPHICITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is called when responding to the StatusBarButtonFillPopupList event. This method adds the graphic into the popup list.

### **Syntax**

[objectreference].AddPopupGraphicItem([BitmapHandle])

### **Parameters**

*BitmapHandle*

The handle to the bitmap that is to be displayed in the popup list.

### **Return value**

True if the item is added; False if the item is not added.

### **Usage**

This method can only be used if the button is of type Graphic. The first item is at the top of the list.

## **Word Pro: AddPopupPointSizeItem method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDPOPUPPOINTSIZETIME\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts an item in the popup list on the point size status bar button object. This method is called when responding to the StatusBarButtonFillPopupList event.

### **Syntax**

[objectreference].AddPopupPointSizeItem(PointSize)

### **Parameters**

*PointSize*

Indicates the value which should appear in the list of point sizes.

### **Return value**

Integer

### **Usage**

You can call this method once for each item you want to add to the list of point sizes. The first item is at the top of the list.

## **Word Pro: AddPopupTextItem method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDPOPUPTEXTITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts an item in a popup list on the text status bar button object. This method is called when responding to the StatusBarButtonFillPopupList event. This method will add the text in the popup list.

### **Syntax**

[objectreference].AddPopupTextItem(Text, [BitmapHandle])

### **Parameters**

#### *Text*

Indicates the text to be displayed in the popup list.

#### *BitmapHandle*

The handle to the bitmap that is to be displayed in the popup list. This parameter is only valid if the button is of type TextAndGraphic. Otherwise, it should be 0. Optional parameter. Data type is Long.

### **Return value**

Integer

### **Usage**

This method may only be used if the button is of type Text, or TextAndGraphic. The first item is at the top of the list.

**Word Pro: AddSectionTabs method**

{button ,AL('H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDSECTIONTABS\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new division after the current division.

**Syntax**

[objectreference].AddSectionTabs()

**Parameters****Return value****Usage**

Equivalent to clicking the right mouse button on an existing division divider tab and choosing Quick Division.

## **Word Pro: AddSmartCorrect method**

{button ,AL('H\_SMARTCORRECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDSMARTCORRECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds the SmartCorrect tool to a document. Equivalent to choosing Edit - SmartCorrect.

### **Syntax**

[objectreference].AddSmartCorrect(Entry,Replacement)

### **Parameters**

#### *Entry*

A String expression representing the entry you want to add to the SmartCorrect tool.

#### *Replacement*

A String expression representing the text you want to use to replace a SmartCorrect entry.

### **Return value**

### **Usage**



## **Word Pro: AddTOCEntry method**

{button ,AL('H\_TOCSUPERTABLELAYOUT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDTOCENTRY\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a table of contents entry for an entire document, for a section, division, or selected text in a document.

### **Syntax**

[objectreference].WPApplication.AddTOCEntry(TOCEntry)

[objectreference].TOCSuperTableLayout.AddTOCEntry()

### **Parameters**

*TOCEntry*

Data type is String.

### **Return value**

### **Usage**

**Word Pro: AddVerbMenu method**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDVERBMENU\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a list of available verbs that a Ole object can support.

**Syntax**

[objectreference].AddVerbMenu(MenuHandle,OleVerbMin,OleVerbMax,OleVerbConve

**Parameters***MenuHandle*

A Numeric expression representing the menu handle used by the AddVerbMenu method. Data type is Long as the numeric expression.

*OleVerbMin*

Data type is Integer.

*OleVerbMax*

Data type is Integer.

*OleVerbConvert*

Data type is Integer.

**Return value**

Integer

**Usage**

## Word Pro: Add method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADD\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds the selected word to the currently active user dictionary.

## Syntax

[objectreference].Add(AddType)

## Parameters

*AddTypeSpell*

Tells Word Pro that you are adding a word to the user dictionary. Data type is Variant which allows the value of this parameter to be either a number or a constant that produces that number. There is no default value. You must include the constant \$LwpAddTypeSpell or its numeric equivalent of 4.

## Return value

Integer

## Usage

If more than one user dictionary is active, Word Pro adds the word to the first dictionary listed in the Spell Check Options dialog box.

If more than one word is selected, only the word at the beginning of the selection is added to the dictionary.

If the word ends with a paragraph or other marker, Word Pro will not add the word to the dictionary.

If no word is selected, the word at the insertion point is added to the dictionary.

If the insertion point is at the end of a word, that word is added to the dictionary.

If the insertion point is at the beginning of a word, that word is added to the dictionary.

If the insertion point is between two spaces, no word is added to the dictionary.

## Word Pro: AdjustShade method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADJUSTSHADE\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the size of a text selection. The selected text could be in a Text, a TextMarker, or a ClickHere object.

### Syntax

[objectreference].AdjustShade(WhichSide, Count, AdjustUnit, MarkerName)

### Parameters

#### *WhichSide*

Specifies which side of the selection you are adjusting. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpWhichSideLeft (2068)

\$LwpWhichSideRight (2069)

\$LwpWhichTypeLeft (1989)

\$LwpWhichTypeRight (1990)

#### *Count*

An Integer expression which specifies how many units (specified in the AdjustUnit parameter) will be added or removed from the selection. To remove units, use positive integers. To add units, use negative integers.

#### *AdjustUnit*

Specifies the type of unit you will use to increment your adjustment. Specify the number of units in the Count parameter. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpAdjustUnitCharacter (2246)

Selects or deselects the specified number of characters.

\$LwpAdjustUnitWord (2247)

Selects or deselects the specified number of words. A word is comprised of a contiguous string of alphanumeric characters. Punctuation and spaces are seen as the end of a word.

\$LwpAdjustUnitChunk (2248)

Selects or deselects the specified number of chunks. A chunk is comprised of a single word (a group of characters with no spaces) and all the contiguous spaces following that word.

\$LwpAdjustUnitSentence (2249)

Selects or deselects the specified number of sentences. A sentence is comprised of a stream of text marked on either side by either a period or a paragraph marker.

\$LwpAdjustUnitObject (2250)

Selects or deselects the specified number of objects.

\$LwpAdjustUnitParagraph (2251)

Selects or deselects the specified number of paragraphs.

\$LwpAdjustUnitMarker (2252)

Moves the specified side of the selection to the marker object named in the MarkerName parameter. If you use this value, you must use 1 for the value of the Count parameter.

#### *MarkerName*

A String expression which specifies the name of the marker object to which you want to move part of your selection. Use this parameter only when you use \$LwpAdjustUnitMarker as the value for the AdjustUnit parameter.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Word Pro sees the sides of a selection as the sides of an expandable blanket. You can stretch or shrink a blanket to fit an area and you can pick up the left side of a blanket and pull it over the right side so the blanket covers an entirely new area. The same is true of a selection in Word Pro. You can change the coverage of your text selection using the left and right sides and marker objects.

For example, if you have a marker named "MarkerOne" and you tell Word Pro to adjust the left side of the selection to match that marker, Word Pro moves the left side of the selection to the MarkerOne position. Word Pro changes the selection in one of three ways:

- If MarkerOne is located before the selection, Word Pro expands the selection to include the text between the original left side and MarkerOne.
- If MarkerOne is located between the original left and right sides of the selection, the selection is reduced to exclude the text between the original left side and MarkerOne.
- If MarkerOne is located after the selection, Word Pro moves the left side to MarkerOne so that the original selection is entirely excluded, and everything between the original right side and MarkerOne becomes selected.

## Word Pro: Adopt method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADOPT\_METHOD\_EXSCRIPT',1)} [See example](#)

Converts a specific child container of a current page container to a child container of a specific parent container within the page.

## Syntax

[objectreference].Adopt()

## Parameters

## Return value

## Usage

An example of how to use the Adopt method can be seen when a page container has a child frame anchored to it. Because the child frame is anchored to the page, it can be moved across the page. If you want to detach the child frame from the page and anchor it to a parent frame container so that the child frame moves within the frame, set the Adopt method.

Equivalent to choosing Frame Properties, clicking the Placement tab, and selecting "In Frame" from the "Place frame" box. Note that the Frame menu displays when the insertion point is in a frame.

## **Word Pro: AdviseOnRename method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADVISEONRENAME\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].AdviseOnRename()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: AdviseOnSave method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADVICEONSAVE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].AdviseOnSave()

### **Parameters**

### **Return value**

### **Usage**



## Word Pro: Anchor method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ANCHOR\_METHOD\_EXSCRIPT',1)} [See example](#)

Attaches an object, such as a table or frame object, to a page or a paragraph in a page.

## Syntax

[objectreference].Anchor([AnchorWhere,] [ConditionType,] [RelativeType,] [AnchorParent])

## Parameters

### *AnchorWhere*

The value of this optional Variant parameter indicates where to attach an object to a page or a paragraph in a page. It must be one of the strings below or its code equivalent. Default is \$LwpAnchorWhereDivisionInfo.

\$LwpAnchorWhereDivisionInfo (12) The default value that attaches an object to a DivisionInfo object.

\$LwpAnchorWhereLayout (13) The value that attaches an object to a layout object.

### *ConditionType*

The value of this optional Variant parameter specifies which pages of a document to attach an object. It must be one of the strings below or its code equivalent. Default is \$LwpConditionTypeAllpages.

\$LwpConditionTypeAllbutspecificpage (156) The value that attaches an object to all pages of a document, except on the page you specify.

\$LwpConditionTypeAllpages (154) The default value that attaches an object to all pages in a document.

\$LwpConditionTypeOnlyevenpages (157) The value that attaches an object to only even pages in a document.

\$LwpConditionTypeOnlyoddpages (158) The value that attaches an object to only odd pages in a document.

\$LwpConditionTypeOnlyspecificpage (155) The value that attaches an object to only a specific page in a document.

\$LwpConditionTypeStartatpage (159) The value that allows you to specify on which page to start attaching an object.

### *RelativeType*

The value of this optional Variant parameter determines where in the page layout the table or frame object is anchored. It must be one of the strings below or its code equivalent. Default is \$LwpRelativeTypeLytInlineNewline.

\$LwpRelativeTypeLytContent (1670) The value that anchors an object so that it is relative to the content box of the parent layout.

\$LwpRelativeTypeLytInline (1668) The value that anchors an object to the text flow as a single character and affects the line height accordingly.

\$LwpRelativeTypeLytInlineNewline (1669) The default value that places an object on a new line by itself in the page layout.

\$LwpRelativeTypeLytInlineVert (1671) The value that anchors an object so that it always moves vertically in the page layout.

\$LwpRelativeTypeLytPara (1667) The value that anchors an object so that the text is relative to a paragraph in the page layout.

\$LwpRelativeTypeLytParent (1666) The value that sets the coordinates of an object so that they are relative to the parent layout and that the anchor position is on the page layout.

### *AnchorParent*

The value of this Variant optional parameter indicates which parent layout to attach an object. It must be one of the strings below or its code equivalent. Default is \$LwpAnchorParentDefault.

\$LwpAnchorParentCell (10) The value that attaches an object to a parent cell object.

\$LwpAnchorParentDefault (11) The default value that attaches a table or frame object.

\$LwpAnchorParentFrame (9) The value that attaches an object to a parent frame object.

### **Return value**

### **Usage**

## Word Pro: AnswerMsgBox method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ANSWERMSGBOX\_METHOD\_EXSCRIPT',1)} [See example](#)

Provides a response to a message box.

### Syntax

[objectreference].AnswerMsgBox(MsgBoxResponse)

### Parameters

#### *MsgBoxResponse*

The response you want to use. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpMsgboxReplyCancel (2076)

\$LwpMsgboxReplyIgnore (2078)

\$LwpMsgboxReplyNo (2080)

\$LwpMsgboxReplyOk (2075)

\$LwpMsgboxReplyRetry (2077)

\$LwpMsgboxReplyYes (2079)

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Use this method to handle any message boxes which Word Pro might display while your script is running. For example, if your script causes a warning message box to appear, you can include this method in your code *before* the statement which elicits the message box. Word Pro answers the first message box it sees with the response you provide in the MsgBoxResponse parameter.

## **Word Pro: AnyEdits method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ANYEDITS\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].AnyEdits(EditorName)

### **Parameters**

*EditorName*

A String expression representing the name of the editor.

### **Return value**

### **Usage**

## **Word Pro: AppendMacro method**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPENDMACRO\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].AppendMacro()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: Backspace method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BACKSPACE\_METHOD\_EXSCRIPT',1)} [See example](#)

Executes a Backspace. This is similar but not identical to pressing the Backspace key.

### **Syntax**

[objectreference].Backspace(Count)

### **Parameters**

#### *Count*

An Integer expression specifying the number of backspaces Word Pro should execute. You must use positive integers for this value. Negative integers will produce unpredictable results.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

If any text is selected when you call this method, the selection is treated the same as the insertion point. The selection itself remains untouched, while the text preceding the selection becomes the subject of the backspace.

## Word Pro: Backward method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CEL  
LAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CL  
ASS;H_DROPCAPCONTAINER_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOO  
TERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_FRAMEGROUPLAYO  
UT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_  
CLASS;H_NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELAYO  
UT_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGROUPLAYOUT_C  
LASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYLAYOUT_CLASS;H_SUBPAGECONTAI  
NER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGR  
OUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEHEADINGLA  
YOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_C  
LASS;H_TOCSUPERTABLELAYOUT_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_BACKWARD_METHOD_EXSCRIPT',1)}
```

[See example](#)

Moves an object or the insertion point backward. A Forward method is also available.

### Syntax

When called from a Layout object:

```
[objectreference.]Backward()
```

When called from a container object:

```
[objectreference.]Backward(Direction)
```

When called from a Text, TextMarker, or ClickHere object:

```
[objectreference.]Backward(Unit, N[, Cursoring][, TextOnly])
```

### Parameters

#### *Direction*

Specifies whether Word Pro should move the insertion point back by page or by window. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

`$LwpDirectionPage (182)`

Moves the insertion point back by one page.

`$LwpDirectionWindow (183)`

Moves the insertion point back by one window.

Only used when calling this method from a container object. A container object is any object created from a container class. A container class is any class derived from the BaseContainer class, including: CellContainer, DropCapContainer, FrameContainer, NoteContainer, PageContainer, ParallelColsContainer, RowContainer, RubyContainer, SubPageContainer, SuperPageContainer, SuperTableContainer, TableContainer, and TableOnlyCont.

#### *Unit*

Specifies the unit of measurement you want to use in moving the insertion point. Use this parameter only when calling this method from a Text, TextMarker, or ClickHere object. You must also use the N parameter to indicate how many of these units to move backward. Data type is Variant, which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

`$LwpNavigateObjectTypeCharacter (1519)`

Moves the insertion point the specified number of characters. Set the Cursoring parameter to True to mimic the use of the arrow keys.

`$LwpNavigateObjectTypeChunk (1522)`

Moves the insertion point the specified number of chunks. A chunk is comprised of a single word (a contiguous group of characters with no spaces) and all the spaces following that word. If the insertion point is at the beginning, the end or anywhere within a word, the chunk is comprised of that word and the spaces that follow it. If the insertion point is between two spaces, the chunk is seen as all the spaces following the insertion point to the beginning of the next word. If there is no word between the spaces and the end of the paragraph, the chunk is comprised of all the spaces up to the end of the paragraph.

`$LwpNavigateObjectTypeObject (1520)`

Any of the objects defined in this list.

`$LwpNavigateObjectTypePage (1518)`

Moves the insertion point the specified number of pages, leaving it at the top of the page.

`$LwpNavigateObjectTypeParagraph` (1524)

Moves the insertion point the specified number of paragraphs. A paragraph is comprised of all the text and tables between two paragraph markers, as well as any frames whose "Place frame" option is set to "With paragraph above."

`$LwpNavigateObjectTypeSentence` (1523)

Moves the insertion point the specified number of sentences. A sentence is comprised of all the text between two periods.

`$LwpNavigateObjectTypeWord` (1521)

Moves the insertion point the specified number of words. A word is comprised of a contiguous string of alphanumeric characters. Punctuation or a space is seen as the end of a word. If the insertion point is between two spaces, the word is comprised of all the spaces on both sides of the insertion point, as well as the word preceding the spaces.

*N*

An Integer expression which specifies the number of units you want to move the insertion point. Use this parameter only when calling this method from a `Text`, `TextMarker`, or `ClickHere` object.

#### *Cursoring*

Use this parameter only when the `Unit` parameter has a value of `$LwpNavigateObjectTypeCharacter`. This parameter takes an Integer expression that indicates whether or not you want Word Pro to move the insertion point as if you were using the arrow keys to move the cursor through a document. When you use the arrow keys, Word Pro skips over hidden markers, such as bookmarks. Default is `False` (0), which causes Word Pro to include any hidden markers when it moves the insertion point by characters. Data type is Integer. The legal values for this parameter are -1 and 0, but you may use the LotusScript constants `True` (-1) and `False` (0) instead of the integer values. Optional parameter. Use this parameter only when calling this method from a `Text`, `TextMarker`, or `ClickHere` object.

#### *TextOnly*

An Integer expression which indicates whether or not you want Word Pro to exclude tables and frames marked as "With paragraph above" when moving the insertion point. Data type is Integer. The legal values for this parameter are -1 and 0, but you may use the LotusScript constants `True` (-1) and `False` (0) instead of the integer values. Optional parameter. Default is `False` (0) which includes tables and certain frames. A value of `True` will cause Word Pro to skip over tables and frames when moving the insertion point. Use this parameter only when calling this method from a `Text`, `TextMarker`, or `ClickHere` object.

#### **Return value**

This method returns a value of -1 (`True`) or 0 (`False`) indicating that the method succeeded or failed respectively.

#### **Usage**

What this method moves backward and how is determined in part or in whole by the object from which you call the method.

When you call this method from a `Layout` object, it moves that `Layout` object backward one level in relation to the other layout objects of the same type.

When you call this method from a container object, Word Pro places the insertion point at the beginning of the previous page.

When you call this method from a `Text`, `TextMarker`, or `ClickHere` object, Word Pro moves the insertion point backward the specified number of units.



## **Word Pro: BeginChange method**

{button ,AL(`H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_BEGINCHANGE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].BeginChange([DontMarkChanges])

### **Parameters**

*DontMarkChanges*

An optional Boolean expression that allows you to mark (True) or not mark (False) any changes you have begun in the Word Pro application, division, or text document object. A boolean expression is either True or False.

### **Return value**

### **Usage**

## **Word Pro: BeginCustomLines method**

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BEGINCUSTOMLINES\_METHOD\_EXSCRIPT',1)} [See example](#)

This method only displays in the Script Editor during a recording to reflect the beginning of a customized table line style selection.

### **Syntax**

[objectreference].BeginCustomLines()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: Bisect method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BISECT\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Bisect(Objectname, Objectname)

### **Parameters**

*Objectname*

A String expression representing the name of the object you want to bisect.

*Objectname*

A String expression representing the name of the object you want to bisect.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: Bold method**

{button ,AL('H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BOLD\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the bold attribute for selected text, or all following text if no text is selected. Acts as a toggle, turning the attribute off if it is on, and on if it is off. Equivalent to choosing Text - Attributes - Bold.

### **Syntax**

[objectreference].Bold()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: BreakLink method**

{button ,AL('H\_GRAPHIC\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BREAKLINK\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].BreakLink(LinkCookie)

[objectreference].BreakLink()

### **Parameters**

*LinkCookie*

Data type is Long.

### **Return value**

Integer

### **Usage**

## **Word Pro: BringFrameToFrontOne method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BRINGFRAMETOFRONTONE\_METHOD\_EXSCRIPT',1)} [See example](#)

Brings the currently active frame one step forward in the frame order. Equivalent to choosing Frame - Priority, then Bring Forward One.

### **Syntax**

[objectreference].BringFrameToFrontOne()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

When you have more than one frame on a page, Word Pro sees the frames as being stacked on top of each other, even if they don't appear to overlap on the page. The first frame you create is at the bottom of the stack. The second frame you create is on top of the first frame, but underneath the third frame, and so on. You can use this method to change the order of a frame in the stack.

## **Word Pro: BringFrameToFront method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_BRINGFRAMETOFRONT\_METHOD\_EXSCRIPT',1)} [See example](#)

Brings the currently active frame to the front of all other frames on the page. Equivalent to choosing Frame - Priority, then Bring to Front.

### **Syntax**

[objectreference].BringFrameToFront()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

When you have more than one frame on a page, Word Pro sees the frames as being stacked on top of each other, even if they don't appear to overlap on the page. The first frame you create is at the bottom of the stack. The second frame you create is on top of the first frame, but underneath the third frame, and so on. You can use this method to change the order of a frame in the stack.

## **Word Pro: CalcSmartLevels method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CALCSMARTLEVELS\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].CalcSmartLevels()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: CalculateSmartLevels method**

```
{button ,AL(^H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL(^H_CALCULATESMARTLEVELS_METHOD_EXSCRIPT',1)} See example
```

Updates the SmartLevels for the currently active division. This update only applies to those paragraphs that are marked to "Use Smart Level" on the Misc panel in the Text Properties InfoBox.

### **Syntax**

```
[objectreference].CalculateSmartLevels()
```

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: Cancel method**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)  
{button ,AL('H\_CANCEL\_METHOD\_EXSCRIPT',1)} [See example](#)

Cancel the printing of a document.

**Syntax**

[objectreference].Cancel()

**Parameters****Return value****Usage**

## Word Pro: CanHaveFootnotes method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCO  
NTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_CANHAVEFOOTNOTES_METHOD_EXSCRIPT',1)} See example
```

[BaseContainer]

Indicates whether or not a table, cell, or row container object can contain footnote objects.

### Syntax

[objectreference].CanHaveFootnotes()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

**Word Pro: CascadeWindow method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CASCADEWINDOW\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the active document window on top of all other open document windows, with the title bar for each document visible. Equivalent to choosing Window - Cascade.

**Syntax**

[objectreference].CascadeWindow()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: Cascade method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CASCADE\_METHOD\_EXSCRIPT',1)} [See example](#)

Cascades the document windows in the application.

**Syntax**

[objectreference].Cascade()

**Parameters**

None

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to choosing Window - Cascade in the Word Pro interface.

## **Word Pro: CellLayout method**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_CELLCONTAINER\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CELLLAYOUT\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the cell layout for a specific cell in a table.

### **Syntax**

[objectreference].CellLayout([Row,] [Column])

### **Parameters**

#### *Row*

An optional Integer parameter that allows you to indicate the specific row from which you want to return its layout.

#### *Column*

An optional Integer parameter that allows you to indicate the specific column from which you want to return its layout.

### **Return value**

Integer

### **Usage**

**Word Pro: CellRevert method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CELLREVERT\_METHOD\_EXSCRIPT',1)} [See example](#)

Reverts the currently active table cell to the attributes of the assigned table cell style.

**Syntax**

[objectreference].CellRevert()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: ChangeAllEditsToEditor method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHANGEALLEEDITSTOEDITOR\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].ChangeAllEditsToEditor(EditorName)

### **Parameters**

*EditorName*

Data type is String.

### **Return value**

### **Usage**



## Word Pro: ChangeSmartMaster method

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CHANGESMARTMASTER\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the SmartMaster for the currently active Word Pro document. Equivalent to choosing File - Choose Another SmartMaster, and specifying a new SmartMaster.

### Syntax

[objectreference].ChangeSmartMaster(PathName,Type, [ApplyTo])

### Parameters

#### *PathName*

A String expression specifying the new SmartMaster to which you are changing.

#### *Type*

A String expression specifying the file type of SmartMaster you want to use. Word Pro automatically recognizes and imports all of the file types listed below. Use this parameter only if the file you are using is not one of these file types:

DCA/RFT	Lotus Manuscript 2.x	MS Word for Windows 1.0
DIF	Lotus Organizer 1.x	MS Word for Windows 2.0
DisplayWrite	Lotus Word Pro	MS Word for Windows 6.0
HTML	Lotus Word Pro SmartMaster	MS Word for Windows95 7.0
Lotus 1-2-3	MS Excel	MS WordPad 1.0
Lotus 1-2-3 for OS/2	MS Excel 3.0	OfficeWriter 4,5,6
Lotus 1-2-3 R3	MS Excel 4.0	Rich Text Format(RTF)
Lotus 1-2-3 R4,5	MS Excel 5.0	SAMNA Word
Lotus 1-2-3 R6	MS Excel 7.0	WordPerfect 5.0
Lotus Ami Pro	MS Windows Write 3.x	WordPerfect 5.1
Lotus Ami Pro 3.x Macro	MS Word for DOS 3,4,5,6	WordPerfect 6.x
Lotus Ami Pro 3.x Styles	MS Word for OS/2	WordStar 2000 R3

#### *ApplyTo*

Allows you to specify the scope of the new SmartMaster. Optional parameter. There are only three legal values for this parameter:

"Entire document" - Applies the new SmartMaster to the entire document.

"All divisions at same level & below" - Applies the new SmartMaster to all the divisions which are at or below the same level as the currently active division. A division's level is indicated by its parent-child relationship to other divisions.

"Current division only" - Applies the new SmartMaster only to the currently active division.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: ChgLineStyle method

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHGLINESTYLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the style of a line in a table.

### Syntax

[objectreference].ChgLineStyle(LineStyle)

### Parameters

#### *LineStyle*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTableLineStyleAll (1878)

\$LwpTableLineStyleCols (1882)

\$LwpTableLineStyleCustom (1886)

\$LwpTableLineStyleInnercols (1887)

\$LwpTableLineStyleInnerRowscols (1888)

\$LwpTableLineStyleMixed (1885)

\$LwpTableLineStyleNone (1877)

\$LwpTableLineStyleOutline (1879)

\$LwpTableLineStyleOutlineall (1880)

\$LwpTableLineStyleOutlinecols (1884)

\$LwpTableLineStyleOutlinerows (1883)

\$LwpTableLineStyleRows (1881)

### Return value

### Usage

**Word Pro: ClearAll method**

{button ,AL('H\_TABRACK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEARALL\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes all tabs from the ruler.

**Syntax**

[objectreference].ClearAll()

**Parameters****Return value****Usage**

**Word Pro: ClearDivisionList method**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEARDIVISIONLIST\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes the list of divisions that can be printed in a document.

**Syntax**

[objectreference].ClearDivisionList()

**Parameters****Return value**

Integer

**Usage**

You can locate a list of divisions you want to remove by choosing File - Print, clicking Select Pages and selecting "Whole divisions" in the Select Pages dialog box.

## **Word Pro: ClearInternalSpellInfo method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CLEARINTERNALSPELLINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].ClearInternalSpellInfo()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: ClearParaRevisionTags method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEARPARAREVISIONTAGS\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes paragraph revision tags from a document. If more than one revision tag exists for a group of revisions, Word Pro prompts the user to leave the paragraphs alone and leave the revision tags intact, or clear the tags and leave both versions of the paragraph.

### **Syntax**

[objectreference].ClearParaRevisionTags()

### **Parameters**

None

### **Return value**

None

### **Usage**

This method affects all tags in all divisions in the currently active document.

**Word Pro: ClearPopupData method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEARPOPUPDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is called when responding to the StatusBarButtonFillPopupList event. This method will clear all items from the popup list.

**Syntax**

[objectreference].ClearPopupData()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) or False (0) instead of the integer values.

**Usage**

Use this method when you want to clear all items from the popup list.

## **Word Pro: ClearSplits method**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEARSPLOTS\_METHOD\_EXSCRIPT',1)} [See example](#)

Clears all split views from the screen. Equivalent to choosing View - Clear All Splits.

### **Syntax**

[objectreference].ClearSplits()

### **Parameters**

### **Return value**

### **Usage**



**Word Pro: ClearTempFoundry method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CLEARTEMPFOUNDRY\_METHOD\_EXSCRIPT',1)} [See example](#)

Clears the contents of the Foundry object located in the TempFoundry property on the WPAApplication object.

**Syntax**

[objectreference].ClearTempFoundry()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

Identical to the Clear method located on the Foundry class, but it only affects the Foundry object located in the TempFoundry property.

## **Word Pro: ClearUpdate method**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEARUPDATE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].ClearUpdate()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: Clear method

```
{button ,AL('H_ATTRIBUTES_CLASS;H_BAG_CLASS;H_CLICKHERE_CLASS;H_DIVISION_CLASS;H_FONT_CLASS;H_FOUNDRY_CLASS;H_MERGEOPTIONS_CLASS;H_OUTLINESTYLESEQUENCE_CLASS;H_PARAGRAPHSTYLE_CLASS;H_TEXT_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_CLEAR_METHOD_EXSCRIPT',1)} See example
```

[Attributes]

[Bag]

[Division]

Clears the content of the Division object.

[Font]

[Foundry]

Clears all objects from a Foundry object. Use this method only on Foundry objects found in the AppFoundry or TempFoundry properties on the WPAApplication object. DO NOT use this method with the Foundry property in WPAApplication, Division, or TextDocument.

[ClickHere]

[TextMarker]

[MergeOptions]

Disconnects the Merge data file from the current document.

[OutlineStyleSequence]

[ParagraphStyle]

[Text]

[TextDocument]

[TOCSuperTableLayout]

## Syntax

[Objectreference].Attributes.Clear

[Objectreference].Bag.Clear

[Objectreference].Division.Clear

[Objectreference].Font.Clear

[Objectreference].Foundry.Clear

[Objectreference].ClickHere.Clear([ClearWhat,] [p2,] [ClassName,] [SubClass])

[Object reference].TextMarker.Clear([ClearWhat,] [ p2,] [ClassName,] [SubClass])

[Object reference].MergeOptions.Clear

[Object reference].OutlineStyleSequence.Clear

[Object reference].ParagraphStyle.Clear

[Object reference].Text.Clear([ClearWhat,] [p2,] [ClassName,] [SubClass])

[Object reference].TextDocument.Clear [Object reference].TOCSuperTableLayout.Clear

## Parameters

[Foundry]

*Reserved*

Not used. Do not use this parameter when using the Clear method on a Foundry object.

[ClickHere, TextMarker, Text]

*ClearWhat*

Data type is Variant. Optional parameter on ClickHere, Text, and TextMarker objects. The value of this parameter must be one of the string values listed below or its Enum code. Default is \$LwpClearWhatDefault.

\$LwpClearWhatClearMisspelledWord (141)

\$LwpClearWhatDefault (145)

\$LwpClearWhatHighlighter (144)

\$LwpClearWhatObject (143)

\$LwpClearWhatTombstone (142)

[ClickHere, TextMarker, Text]

*p2*

Data type is Variant. Optional parameter. Default is 0.

[ClickHere, TextMarker, Text]

*ClassName*

Data type is String. Optional parameter.

[ClickHere, TextMarker, Text]

*SubClass*

Data type is String. Optional parameter.

### **Return value**

ClickHere, TextMarker, Text, Foundry - A Boolean value indicating success (-1) or failure (0). The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

[Foundry]

Used from the Foundry property on Division, TextDocument, or WPApplication, this method clears all styles and everything else.

## Word Pro: CloseAll method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLOSEALL\_METHOD\_EXSCRIPT',1)} [See example](#)

Closes all open documents. This method does not close hidden files, such as glossary files.

### Syntax

[objectreference].CloseAll([CloseFile])

### Parameters

#### *CloseFile*

Allows you an untitled document without the Save Changes dialog box, as long as the document has no contents. Default is \$LwpCloseFileIfLastdocOpenUntitled, which closes empty untitled documents without a prompt. Data type is Variant which allows you to use one of the string values below or its numeric equivalent (in parentheses).

\$LwpCloseFileIfLastdocNoOpen (147) Prompts you to save the untitled document before closing.

\$LwpCloseFileIfLastdocOpenUntitled (146) Closes the untitled document without saving.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

**Word Pro: CloseDocWindow method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLOSEDOCWINDOW\_METHOD\_EXSCRIPT',1)} [See example](#)

Closes the currently active document window.

**Syntax**

[objectreference].CloseDocWindow([AskUserToSave])

**Parameters**

*AskUserToSave*

Allows you to prompt the user to save the document before closing. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True which prompts the user to save the document.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: CloseMergeDataFile method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLOSEMERGEDATAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Closes the data file for the active merge document. Any changes the user makes are lost if this method is called before the user saves the file.

### **Syntax**

[objectreference].CloseMergeDataFile()

### **Parameters**

None

### **Return value**

### **Usage**

## **Word Pro: CloseObject method**

{button ,AL(`H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CLOSEOBJECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Closes the comment note at the insertion point.

### **Syntax**

[objectreference].CloseObject()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

If there is no comment note at the insertion point, Word Pro does nothing.

A comment note is represented in LostuScript by a NoteContainer object which is comprised of several class members and objects, including a NoteLayout, a DivisionInfo, and a Presentation.



## Word Pro: Close method

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCUMENT\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WINDOW\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLOSE\_METHOD\_EXSCRIPT',1)} [See example](#)

Closes the object from which you call this method. For example, when you call this method from WPAApplication, it closes the active document, but when you call it from a StatusBar object, it hides the status bar from which you call the method. See Usage below for details of how this method affects each type of object.

### Syntax

[Objectreference].WPAApplication.Close([SaveChanges,] [DocName,] [Location,] [ DocType,] [CloseFile])

[Objectreference].TextDocument.Close([SaveChanges,] [DocName,] [Location,] [DocType,] [CloseFile])

[Objectreference].ApplicationWindow.Close() Integer

[Objectreference].DocWindow.Close() Integer

[Objectreference].Window.Close() Integer

### Parameters

#### *SaveChanges*

Used only on WPAApplication and TextDocument objects. This parameter lets you choose to save or dismiss all changes before closing a document. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True.

#### *DocName*

Used when the document has never been saved, this parameter takes a string expression which specifies the name of the document you are closing.

#### *Location*

Used when the document has never been saved, this parameter takes a string expression which specifies the directory path for the document.

#### *DocType*

Used when the document has never been saved, this parameter takes a string expression which specifies the file type for the document. Default file type is Word Pro document.

#### *CloseFile*

Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default value is \$LwpCloseFileIfLastdocOpenUntitled.

\$LwpCloseFileIfLastdocNoOpen (147) Leaves an empty application workspace window if you close the last document.

\$LwpCloseFileIfLastdocOpenUntitled (146) Opens an untitled file if you close the last document.

### Return value

The return values for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

This method appears on several objects. It closes whatever type of object you call it from.

WPAApplication - Closes the currently active Word Pro document.

TextDocument - Closes the document from which you call the method.

DocWindow - Closes the document from which you call the method.

StatusBar - Closes/hides the status bar.

ApplicationWindow - Closes the application window. This is NOT the same as choosing File - Exit Word Pro. Although the application window closes, if Word Pro is serving an object to an external client, it remains active but not visible. When all the served objects are released, the application terminates.



## Word Pro: CombineDivisions method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COMBINEDIVISIONS\_METHOD\_EXSCRIPT',1)} [See example](#)

Combines two divisions into a single division. The divisions must be adjacent to each other.

### Syntax

[objectreference].CombineDivisions (StartName, EndName)

### Parameters

#### *StartName*

The internal name for the first division you want to combine. This is not the name Word Pro displays in the division tab. Data type is String.

#### *EndName*

The internal name for the first division you want to combine. This is not the name Word Pro displays in the division tab. Data type is String.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

You can get the internal name for the active division by calling the Name property of the currently active Division object as shown below:

```
DIM StartName As String
StartName = .Division.Name
```

For more information on division names in LotusScript, see [Overview: Division names in LotusScript](#)

## Word Pro: CombineSections method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COMBINESECTIONS\_METHOD\_EXSCRIPT',1)} [See example](#)

Combines the contents of up to ten sections by removing the specified section breaks.

### Syntax

```
[objectreference].CombineSections([Section1Name] [, Section2Name] [, Section3Name]  
[, Section4Name] [, Section5Name] [, Section6Name] [, Section7Name] [, Section8Name]  
[, Section9Name] [, Section10Name])
```

### Parameters

#### *Section1Name*

A String expression which specifies a section break you want to remove. The contents of this section will then be placed at the end of the previous section.

*Section2Name* through *Section10Name* allow you to specify additional section breaks to be removed. The contents of each section are added to the end of the preceding section.

**Note** The names used in these parameters are the internal hexadecimal names found in the Name property on a Section object. You can access a Section object through the SectionCollection found in the Sections property in the Division's Foundry object. A sample script is provided below.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Similar to clicking the right mouse button on a section tab and choosing Combine Sections. However, unlike the menu command, this method allows you to combine more than two sections.

You can use the following script to get both the internal hexadecimal name (found in the Name property of each Section object) and the name which appears in the Section tab (found in the UserName property of each Section object).

```
Print "Section Label = Section Internal Name"  
Forall sec In .Division.Foundry.Sections  
    Print sec.UserName + " = " sec.Name  
End Forall
```

## Word Pro: CompareFiles method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COMPAREFILES\_METHOD\_EXSCRIPT',1)} [See example](#)

Compares the file(s) you specify with the currently active document.

### Syntax

[objectreference].CompareFiles(FilePath, FileType, IsMultiDocs, IndexOfMultiDocToCompare)

### Parameters

#### *FilePath*

A String expression specifying the name and path for the document that you want to compare against the currently active document. If you are comparing more than one document, this should be the name of the first document you want to compare. Data type is String.

#### *FileType*

The file type of document named in FilePath. Use a null string ("") to have Word Pro automatically detect the file type. Word Pro recognizes the following file types:

DCA/RFT	Lotus Manuscript 2.x	MS Word for Windows 1.0
DIF	Lotus Organizer 1.x	MS Word for Windows 2.0
DisplayWrite	Lotus Word Pro	MS Word for Windows 6.0
HTML	Lotus Word Pro SmartMaster	MS Word for Windows95 7.0
Lotus 1-2-3	MS Excel	MS WordPad 1.0
Lotus 1-2-3 for OS/2	MS Excel 3.0	OfficeWriter 4,5,6
Lotus 1-2-3 R3	MS Excel 4.0	Rich Text Format(RTF)
Lotus 1-2-3 R4,5	MS Excel 5.0	SAMNA Word
Lotus 1-2-3 R6	MS Excel 7.0	WordPerfect 5.0
Lotus Ami Pro	MS Windows Write 3.x	WordPerfect 5.1
Lotus Ami Pro 3.x Macro	MS Word for DOS 3,4,5,6	WordPerfect 6.x
Lotus Ami Pro 3.x Styles	MS Word for OS/2	WordStar 2000 R3

#### *IsMultiDocs*

Indicates whether you are comparing one file or multiple files to the active document. Data type is Integer. A value of -1 (True) indicates you are comparing multiple documents. A value of 0 (False) indicates that you are comparing a single document.

#### *IndexOfMultiDocToCompare*

When comparing multiple documents, the value of this parameter specifies which document is being compared in the current iteration of the loop. Data type is Integer.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

To compare multiple files, you must use a loop which will call this method for each file you want to compare, and you must include different values for the *IndexOfMultiDocToCompare* parameter in each iteration of the loop.

## **Word Pro: Configure method**

{button ,AL('H\_ICONBAR\_CLASS;H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONFIGURE\_METHOD\_EXSCRIPT',1)} [See example](#)

Activates (loads and displays) the SmartIcons Setup dialog box.

### **Syntax**

[objectreference].Configure()

### **Parameters**

Data type is Integer. The legal values for this parameter will always be -1 or 0 but you can use the LotusScript constants of True (-1) and False (0).

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

[IconBar]

Allows you to add, change, and edit an existing icon. The bar you are currently working with will display in the SmartIcons Setup dialog box.

[IconBarManager]

Allows you to add, change, and edit icon bar sets by using the list in the IconBarManager.

**Word Pro: ConnectCells method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONNECTCELLS\_METHOD\_EXSCRIPT',1)} [See example](#)

Connects selected table cells. Equivalent to choosing Table - Connect Cell or Table - Disconnect Cell.

**Syntax**

[objectreference].ConnectCells()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

If you disconnect a cell, the contents of the cell remain in a single cell rather than being returned to their original separate cells. Use the DisconnectCells method to disconnect rows.

## Word Pro: ConnectContainer method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONNECTCONTAINER\_METHOD\_EXSCRIPT',1)} [See example](#)

Groups the selected container objects.

### Syntax

[objectreference].ConnectContainer()

### Parameters

### Return value

### Usage

You can use the ConnectContainer method to group two or more containers. When the containers are grouped, handles display on the sides. These handles can be used to move the grouped containers. This is usually used when selecting frames, cells, and so on.

For more information on container objects, see [BaseContainer class](#).



**Word Pro: ConnectRows method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CONNECTROWS\_METHOD\_EXSCRIPT',1)} [See example](#)

Connects all the cells in the same row as the active or selected cell(s). Equivalent to choosing Table - Connect Row or Table - Disconnect Cell.

**Syntax**

[objectreference].ConnectRows()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

If you disconnect a row, the contents of the row remain in a single cell rather than being returned to their original separate cells and rows. Use the DisconnectCells method to disconnect rows.

**Word Pro: ConnectSectionTabs method**

{button ,AL('H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONNECTSECTIONTABS\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new division and makes the currently selected division the child of the new division.

**Syntax**

[objectreference].ConnectSectionTabs()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to clicking the right mouse button on an existing division divider tab and choosing Group Tabs. Word Pro creates a new parent division and places the division on which you originally clicked in the new parent.

## Word Pro: Connect method

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CONNECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Connects two or more rows, columns, and cells to create one large row, column or cell in a table object.

### Syntax

[objectreference].Connect([StartRow],[StartCol],[EndRow],[EndCol],[MergeCon])

### Parameters

#### *StartRow*

Allows you to indicate where you want to start connecting rows in a table object.

Data type is Integer. Optional parameter.

#### *StartCol*

Allows you to indicate where you want to start connecting columns in a table object.

Data type is Integer. Optional parameter.

#### *EndRow*

Allows you to indicate where you want to end connecting rows in a table object.

Data type is Integer. Optional parameter.

#### *EndCol*

Allows you to indicate where you want to end connecting columns in a table object.

Data type is Integer. Optional parameter.

#### *MergeContents*

Merges all the contents in the rows or columns that have been connected.

Data type is Boolean. Optional parameter. Default is True.

### Return value

This method returns an Integer value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: ContractOutlineLevel method

```
{button ,AL('H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_CONTRACTOUTLINELEVEL_METHOD_EXSCRIPT',1)} See example
```

Contracts the lowest level heading(s) that are subordinate to the paragraph for which you are calling the method. For example, when you call this method for a Level 1 heading, it will contract the lowest level heading(s) which are subordinate to that Level 1 heading.

### Syntax

*For WPAplication objects:*

```
[objectreference.]ContractOutlineLevel([ContractAll])
```

*For Text, TextMarker, and ClickHere objects:*

```
[objectreference.]ContractOutlineLevel(ContractAll)
```

### Parameters

ContractAll

Allows you to contract all the subordinate headings under the heading from which you call this method. Data type is Integer but the legal values for this parameter are -1 and 0. You may use the LotusScript constants True (-1) and False (0). A value of True will cause all subordinate headings to be contracted regardless of their level. When called from WPAplication, this parameter is optional and has a default of True.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

This method is defined in four different classes:

[WPAplication]

Call the method from this object when you want to contract the outline level(s) for the heading which currently has the focus.

[ClickHere]

Call the method from this object when you want to contract the outline level(s) for a heading in a ClickHere object.

[TextMarker]

Call the method from this object when you want to contract the outline level(s) for a heading in a TextMarker object.

[Text]

Call the method from this object when you want to contract the outline level(s) for a specific Text object.

**Word Pro: Contract method**

{button ,AL('H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTRACT\_METHOD\_EXSCRIPT',1)} [See example](#)

Hides divider tabs that are the children of a parent division tab in a document.

**Syntax**

[objectreference].Contract()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to clicking the minus sign on the parent division tab to hide all the children divider tabs and displaying just the parent division tab.

**Word Pro: ConvertToClass method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONVERTTOCLASS\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].ConvertToClass(AFIDClassName)

**Parameters**

*AFIDClassName*

Data type is String.

**Return value**

Integer

**Usage**

## Word Pro: CopyItem method

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COPYITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Copies a specified menu item object from one menu item's parent object to another.

### Syntax

[objectreference].CopyItem(FromItem,[After,] [TargetText,] [Caption])

### Parameters

#### *FromItem*

Specifies the menu item you want to copy.

#### *After*

Default of True places the copied item after last item in the parent menu item object. Setting the value of After to False places the copied item before the first item in the parent menu item object. Optional Boolean expression. A Boolean expression is either True or False.

#### *TargetText*

An optional String expression that allows you to specify any menu item object and place the copied item before or after it.

#### *Caption*

The name of the copied menu item that displays on the menu. You can use this optional String parameter to change the caption of a copied menu item.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this method to copy existing menu items from one location to another.

## **Word Pro: CopyMeaning method**

{button ,AL('H\_GLOSSARY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COPYMEANING\_METHOD\_EXSCRIPT',1)} [See example](#)

Copies the meaning of a glossary term to the temporary Foundry.

### **Syntax**

[objectreference].CopyMeaning()

### **Parameters**

### **Return value**

String

### **Usage**

This method is used when a user inserts the meaning of a glossary term into the active document. Word Pro copies the meaning from the glossary file to the temporary Foundry and then pastes the meaning from the temporary Foundry into the active document.



## **Word Pro: CopySelection method**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COPYSELECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Copies the current selection to the Clipboard and to the Foundry object, located in the AppFoundry property on WPAApplication. Equivalent to choosing Edit - Copy.

### **Syntax**

[objectreference].CopySelection()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: Copy method

{button ,AL('H\_BASSETABLE\_CLASS;H\_DIVISION\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FOUNDRY\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TEXDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COPY\_METHOD\_EXSCRIPT',1)} [See example](#)

[Foundry class]

Copies objects from one Foundry object into another Foundry object.

[TextDocument class]

Performs a copy operation on a selected object. To copy selected text to the Clipboard, use .CopySelection

[BaseTable]

Copies a range of selected cells in a table object.

## Syntax

[Objectreference].Division.Copy(Name,[ParentName,][BeforeNeighbor,][NeighborName])

[Objectreference].Foundry.Copy([ObjectType,][ObjectName,][Foundry Type,] [p4,][NewName])

[Objectreference].BaseTable.Copy([Temporary])

[Objectreference].TextDocument.Copy(Name,[ParentName,][BeforeNeighbor,][NeighborName])

## Parameters

[Division and TextDocument classes]

*Name*

Data type is String.

*ParentName*

Data type is String. Optional parameter.

*BeforeNeighbor*

Data type is Bool. Optional parameter. Default is False.

*NeighborName*

Data type is String. Parameter is optional.

[Foundry class]

*ObjectType*

Specifies what type of object you are copying from this Foundry object. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent. Default is \$LwpCopyObjectTypeLayout.

\$LwpCopyObjectTypeContents (170) Allows you to copy any Content object. You must specify the Content object's name in the ObjectName parameter.

\$LwpCopyObjectTypeDivision (171) Allows you to copy any Division object. You must specify the Division object's name in the ObjectName parameter.

\$LwpCopyObjectTypeLayout (168) Allows you to copy any Layout object. You must specify the Layout object's name in the ObjectName parameter.

\$LwpCopyObjectTypeStyle (169) Allows you to copy objects which were created from any of the following classes: CellLayout, CharacterStyle, FrameLayout, PageLayout, ParagraphStyle, TableLayout. You can specify a single object using the ObjectName and P4 parameters, or you can copy all objects created from a single class by leaving ObjectName blank and specifying the class in P4.

*ObjectName*

The name of the object you are copying. Required parameter for copying Content, Layout, and Division objects.

When copying SmartMaster-derived objects, you can specify one object by name or leave this parameter empty to copy all SmartMaster-derived objects. Data type is String.

*FoundryType*

Indicates the Foundry object to which you are copying the object. Data type is Variant. The value of this optional parameter must be one of the strings below or its code equivalent. Default is \$LwpFoundryTypeDocument.

\$LwpFoundryTypeApplication (346) Copies to AppFoundry property.  
\$LwpFoundryTypeDocument (345) Copies to Division.Foundry property.  
\$LwpFoundryTypeTemporary (347) Copies to TempFoundry property.

#### *p4*

When you use \$LwpCopyObjectTypeStyle as the value for ObjectType, you can specify further which class of object you are copying. You can choose objects created from one of the following classes: CellLayout, CharacterStyle, FrameLayout, PageLayout, ParagraphStyle, TableLayout. If you leave this blank and use a null string ("") as the value of ObjectName, all objects created from any of these classes will be copied. Data type is Variant. The value of this parameter must be one of the strings below or its numeric equivalent (shown in parentheses).

\$LwpStyleTypeCell (1834)  
\$LwpStyleTypeCharacter (1830)  
\$LwpStyleTypeDefault (1828)  
\$LwpStyleTypeFrame (1832)  
\$LwpStyleTypePage (1831)  
\$LwpStyleTypeParagraph (1829)  
\$LwpStyleTypeTable (1833)

#### *NewName*

Allows you to specify a new name for the copied object. Data type is String. Optional parameter.

[BaseTable class]

#### *Temporary*

An Integer value of -1 or 0 indicating whether the selection will be copied to the temporary foundry (-1) or to the clipboard (0). You can use the LotusScript constants of True (-1) and False (0) as the value for this parameter.

#### **Return value**

[Division, TextDocument classes]

[Foundry class]

A String value indicating the name of the object created. No return value is given when ObjectType is \$LwpCopyObjectTypeStyle.

[BaseTable class]

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

#### **Usage**

## Word Pro: CreateDataFile method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEDATAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new Merge data file including the records and fields you specify. The currently active document becomes the Merge document for the new data file. Equivalent to clicking the Create New button in the Mail Merge Assistant dialog box to display the Create Data File dialog box.

### Syntax

[objectreference].CreateDataFile(Delimiters,FieldNames,[IsAscii])

### Parameters

#### *Delimiters*

A String expression specifying the characters or symbols which you want Word Pro to use to delineate between fields and records. The value of this parameter can be any two alphanumeric characters. Default is "~|". Tilde is the separator for fields and the bar is the separator for records. If *IsAscii* is True, the value of this parameter must be "Fixed length ASCII."

#### *FieldNames*

A String expression representing the names of the fields in the new data file. Each field name is separated by the first delimiter character specified in the *Delimiters* parameter.

#### *IsAscii*

Indicates whether or not the data file is fixed length ASCII. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: CreateDivision method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new division in a document using a SmartMaster or another file. Equivalent to choosing Create - Division.

### Syntax

[objectreference].CreateDivision (MasterFileName [, FileType] [, DivisionLocation] [,Parent] [,NeighborName])

### Parameters

#### *MasterFileName*

A String expression which specifies the name of a SmartMaster file or an external file from which you want to create the new division.

#### *FileType*

An optional String expression which specifies the file type of the file used in creating the division. Word Pro automatically recognizes and imports all of the file types listed below. Use this parameter only if the file you are using is not one of these file types:

DCA/RFT	Lotus Manuscript 2.x	MS Word for Windows 1.0
DIF	Lotus Organizer 1.x	MS Word for Windows 2.0
DisplayWrite	Lotus Word Pro	MS Word for Windows 6.0
HTML	Lotus Word Pro SmartMaster	MS Word for Windows95 7.0
Lotus 1-2-3	MS Excel	MS WordPad 1.0
Lotus 1-2-3 for OS/2	MS Excel 3.0	OfficeWriter 4,5,6
Lotus 1-2-3 R3	MS Excel 4.0	Rich Text Format(RTF)
Lotus 1-2-3 R4,5	MS Excel 5.0	SAMNA Word
Lotus 1-2-3 R6	MS Excel 7.0	WordPerfect 5.0
Lotus Ami Pro	MS Windows Write 3.x	WordPerfect 5.1
Lotus Ami Pro 3.x Macro	MS Word for DOS 3,4,5,6	WordPerfect 6.x
Lotus Ami Pro 3.x Styles	MS Word for OS/2	WordStar 2000 R3

#### *DivisionLocation*

Indicates where you want the new division inserted. Data type is Variant which allows the value of this parameter to be one of the three division locations listed below or its numeric equivalent (in parentheses). Default is \$LwpDivLocInsertAtInsertionPt.

\$LwpDivLocInsertBeforeCurrentdiv (184) Inserts the new division before the currently active division.

\$LwpDivLocInsertAfterCurrentdiv (185) Inserts the new division after the currently active division.

\$LwpDivLocInsertAtInsertionPt (186) Inserts the new division at the insertion point. All items that fall before the insertion point remain part of the active division. All items after the insertion point become part of the new division.

**Note** If the insertion point is in a table cell or a frame, Word Pro splits the contents of the cell or frame, leaving the items before the insertion point intact and moving the items after the insertion point into the new division. Items outside the cell or frame are not affected and remain in the original division.

#### *Parent*

An optional String expression representing the internal name of the division which you want to become the parent of the new division.

#### *NeighborName*

An optional String expression representing the name of the division which you want to become the neighbor of the new division.

### Return value

A String expression which represents the internal name of the new division.

For more information on division names in LotusScript, see [Overview: Division names in LotusScript](#)

### Usage



**Word Pro: CreateDocument method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEDOCUMENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new document. It is equivalent to File - New Document.

**Syntax**

[objectreference].CreateDocument()

**Parameters****Return value****Usage**

## Word Pro: CreateDropCap method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEDDROPCAP\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a DropCap at the insertion point. Each DropCap is comprised of several objects and is accessible through the DropCaps property on a Foundry object.

### Syntax

[objectreference].CreateDropCap(NumLines, Position)

### Parameters

#### *NumLines*

Specifies the height of the DropCap in lines of text. Data type is Integer. If you specify 3 lines, the DropCap will be as high as three lines of text in the current paragraph style.

#### *Position*

An Integer which allows you to specify the position of the DropCap. There are three legal values for this parameter:

1 = Below

Aligns the top edge of the DropCap with the top edge of the first line of text and places the DropCap inside the page margin so that the remaining lines of text flow around the DropCap.

2 = Above

Aligns the bottom edge of the DropCap with the bottom edge of the first line of text and places the DropCap inside the page margin so that the preceding lines of text flow above the DropCap.

3 = Beside

This is the same as choosing Below, except the DropCap is placed in the margin beside the text so there is no text flowing around the DropCap.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage



## Word Pro: CreateExternalDivision method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEEXTERNALDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new division which is linked to the contents of an external file. If you edit the contents of the division, you can save your changes to the external file. Similarly, if you edit the external file, the external division reflects those changes the next time you open the Word Pro document.

### Syntax

[objectreference].CreateExternalDivision(Path [, FileType][, DivisionLocation][, Parent][, NeighborName])

### Parameters

#### *Path*

A String expression representing the drive and directory location of the external file you want to use as the source for the external division.

#### *FileType*

An optional String expression which specifies the file type of the file specified in the Path parameter. Word Pro automatically recognizes and imports many file types. Use this parameter only if the file specified in the Path parameter is not one of these file types:

DCA/RFT	Lotus Manuscript 2.x	MS Word for Windows 1.0
DIF	Lotus Organizer 1.x	MS Word for Windows 2.0
DisplayWrite	Lotus Word Pro	MS Word for Windows 6.0
HTML	Lotus Word Pro SmartMaster	MS Word for Windows95 7.0
Lotus 1-2-3	MS Excel	MS WordPad 1.0
Lotus 1-2-3 for OS/2	MS Excel 3.0	OfficeWriter 4,5,6
Lotus 1-2-3 R3	MS Excel 4.0	Rich Text Format(RTF)
Lotus 1-2-3 R4,5	MS Excel 5.0	SAMNA Word
Lotus 1-2-3 R6	MS Excel 7.0	WordPerfect 5.0
Lotus Ami Pro	MS Windows Write 3.x	WordPerfect 5.1
Lotus Ami Pro 3.x Macro	MS Word for DOS 3,4,5,6	WordPerfect 6.x
Lotus Ami Pro 3.x Styles	MS Word for OS/2	WordStar 2000 R3

#### *DivisionLocation*

A String or Integer value which indicates where you want the new division inserted. Data type is Variant which allows the value of this parameter to be one of the three division locations listed below or its numeric equivalent (in parentheses). Default is \$LwpDivLocInsertAtInsertionPt.

\$LwpDivLocInsertBeforeCurrentdiv (184) Inserts the new division before the currently active division.

\$LwpDivLocInsertAfterCurrentdiv (185) Inserts the new division after the currently active division.

\$LwpDivLocInsertAtInsertionPt (186) Inserts the new division at the insertion point. All items which fall before the insertion point remain part of the active division. All items after the insertion point become part of the new division.

**Note** If the insertion point is in a table cell or a frame, Word Pro splits the contents of the cell or frame, leaving the items before the insertion point intact and moving the items after the insertion point into the new division. Items outside the cell or frame are not affected and remain in the original division.

#### *Parent*

An optional String expression representing the internal name of the division which you want to become the parent of the external division.

#### *NeighborName*

An optional String expression representing the name of the division which you want to become the neighbor of the external division.

### Return value

A String expression which represents the internal name of the external division.

For more information on division names in LotusScript, see [Overview: Division names in LotusScript](#)

### **Usage**

Word Pro displays the contents of the external source file within the external division in the Word Pro document. The contents are displayed in a format which approximates the way in which they would be displayed in the source application.

You can specify where you want Word Pro to place the new external division by using the DivisionLocation, Parent, or NeighborName parameters.

## Word Pro: CreateFrame method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEFRAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a frame object in a document. Equivalent to choosing Create - Frame.

### Syntax

[objectreference].CreateFrame([UseDefault],[FrameStyle],[Width],[Height]

### Parameters

#### *UseDefault*

Allows you to use the default frame style (True). Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True.

#### *FrameStyle*

A String expression specifying the name of the frame style for the new frame.

#### *Width*

An Integer that specifies the width of the new frame in Twips. This parameter is only needed if you do not use the default frame style.

#### *Height*

An Integer that specifies the height of the new frame in Twips. This parameter is only needed if you do not use the default frame style.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

**Word Pro: CreateFromBitmap method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEFROMBITMAP\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a graphic object from a bitmap.

**Syntax**

[objectreference].CreateFromBitmap(BitMmapHandle,IsDeviceIndependent)

**Parameters**

*BitmapHandle*

Data type is Long.

*IsDeviceIndependent*

Data type is Integer.

**Return value**

Integer

**Usage**

## **Word Pro: CreateFromClipBrd method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEFROMCLIPBRD\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a graphic object from the clipboard.

### **Syntax**

[objectreference].CreateFromClipBrd([ClipBrdFormat])

### **Parameters**

*ClipBrdFormat*

Data type is String. Optional parameter.

### **Return value**

Integer

### **Usage**

## **Word Pro: CreateFromDataObject method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEFROMDATAOBJECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a graphic from a data object.

### **Syntax**

[objectreference].CreateFromDataObject(DataObjPtr,[ClipBrdFormat])

### **Parameters**

*DataObjPtr*

Data type is Long.

*ClipBrdFormat*

An optional String expression representing the format of the Clipboard.

### **Return value**

Integer

### **Usage**

## **Word Pro: CreateFromMetafile method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEFROMMETAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a graphic from a metafile.

### **Syntax**

[objectreference].CreateFromMetafile(MetaFileHandle,EnhancedMetafile)

### **Parameters**

*MetaFileHandle*

Data type is Long.

*EnhancedMetafile*

Data type is Integer.

### **Return value**

Integer

### **Usage**

## **Word Pro: CreateGlossaryEntry method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEGLOSSARYENTRY\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a glossary entry for the current selection in the specified Glossary file. Equivalent to choosing Edit - Glossary and inserting a new glossary entry for the current selection.

### **Syntax**

[objectreference].CreateGlossaryEntry(GlossFilePath, KeyName)

### **Parameters**

*GlossFilePath*

A String expression which specifies the path and name of the Glossary file (.GLS) to which you are adding this entry.

*Keyname*

A String expression you want to use as the abbreviation for the new glossary entry. Equivalent to the string value you provide as the "Glossary entry name" in the Glossary dialog box.

### **Return value**

None

### **Usage**



## **Word Pro: CreateGlossary method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEGLOSSARY\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a Glossary object at the insertion point in the active document. Word Pro displays the Glossary object in parallel columns with "Name" at the top of the first column and "Contents" at the top of the second column.

### **Syntax**

[objectreference].CreateGlossary()

### **Parameters**

None

### **Return value**

None

### **Usage**

## Word Pro: CreateGraphic method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEGRAPHIC\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a graphic object in a Word Pro document. You can use the parameters to specify the type of graphic object and whether you want the user or Word Pro to draw the graphic's frame.

### Syntax

[objectreference].CreateGraphic(AFIDClassName, ScratchOutFrame)

### Parameters

#### *AFIDClassName*

A String expression which specifies the type of graphic you are creating. There are three types of graphics which are native to Word Pro: equations, drawings, and charts.

#### WordProEqn

Use this value if you want to create an Equation graphic. When you use this value, Word Pro switches to equation mode and places the insertion point in the equation frame.

#### WordProDraw

Use this value to create a drawing using the Word Pro drawing tools. When you use this value, Word Pro switches to drawing mode and displays the drawing tools.

#### LotusChart (or WordProChart)

If you are using the 32-bit or OS/2 version of Word Pro, you can use "LotusChart" to create a chart graphic. If you are using the 16-bit version of Word Pro, you must use "WordProChart." The effect is much the same. Either value launches the charting tool for Word Pro, which allows you to create a chart for the new chart graphic frame.

#### *ScratchOutFrame*

An Integer value which indicates whether you want to draw the new graphic frame by hand or let Word Pro draw the frame based on a frame style. If you want to draw the frame yourself, use the value of True (-1) for this parameter. If you want Word Pro to draw the frame based on an existing style, use a value of False (0) for this parameter.

### Return value

None

### Usage

When ScratchOutFrame is set to False, Word Pro checks the type of graphic you are creating and uses the default style for that type of graphic frame. If no default style exists for the type of graphic you are creating, Word Pro uses a predefined style to create that frame.

The drawing and chart graphics share the same default frame style. Equation graphics have their own default frame style.

## Word Pro: CreateNewButton method

{button ,AL('H\_STATUSBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATENEWBUTTON\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new button in the status bar. After the button is created, the InvalidateWholeBar method should be called to repaint the status bar.

### Syntax

[objectreference].CreateNewButton(ParentButtonId, InsertAfterButtonId, ButtonWidth, ButtonType)

### Parameters

#### *ParentButtonId*

Data type is Long. Required parameter. Value should be 0 unless you are creating a child button.

#### *InsertAfterButtonId*

Data type is Long. Required parameter. Value of 0 causes the button to be added to the beginning of the bar.

#### *ButtonWidth*

Data type is Integer. Required parameter.

#### *ButtonType*

Data type is Variant which allows the value of this parameter to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these constants when you want Word Pro to combine the features listed below. Use the OR operator to combine constants.

LwpButtonBehaviorClickable (&H8) A value that allows the button to be left-clicked.

LwpButtonBehaviorCollapsible (&H10) A value that allows the button to shrink or grow so that the status bar can fill up the window. Only one is allowed per status bar. Word Pro's collapsible button is the date/time button.

LwpButtonBehaviorContainer (&H20) A value that allows the button to contain child buttons.

LwpButtonBehaviorLeftclick (&H8) A value that allows the button to be left-clicked.

LwpButtonBehaviorPopup (&H4) A value that allows the button to pop up a list of alternatives.

LwpButtonBehaviorThermometer (&H80000) A value that allows the button to display a thermometer graphic.

LwpButtonCanBeDepressed (&H40000) A value that allows the button to stay depressed.

LwpButtonContentsCenterAligned (&H80) A value that allows the button contents to be center-aligned.

LwpButtonContentsGray (&H200) A value that allows the button contents to be grayed.

LwpButtonContentsHilited (&H400) A value that allows the button contents to be highlighted (red in Word Pro).

LwpButtonContentsLeftAligned (&H40) A value that allows the button contents to be left-aligned.

LwpButtonContentsRightAligned (&H100) A value that allows the button contents to be right-aligned.

LwpButtonHasAutorepeat (&H4000) A value that allows the button to repeat a command.

LwpButtonHasUpdownCtrl (&H20000) A value that allows the button to have up/down control.

LwpButtonNoTextFromHost (&H800) A value that allows the button to keep its user-defined text without changing; in other words, the text on this button is never going to require text from a host.

LwpButtonReserved (&H8000)

LwpButtonSpacer (&H10000) A spacer status bar button.

LwpButtonSupportDbIclick (&H2000) A value that allows the button to respond to a double-click .

LwpButtonSupportRightClick (&H1000) A value that allows the button to support a right mouse click.

LwpButtonTypeGraphics (&H2) A value that allows the button to display a graphic.

LwpButtonTypeText (&H1) A value that allows the button to display text.

### Return value

String

### Usage

Use this method to add a new button to the status bar object. Some of the ButtonType parameters can be combined together. For example, you can combine the left-click, text, and left-aligned values to allow the button to respond to a

left-mouse click and display left-aligned text. You cannot combine certain parameters that are in obvious conflict with each other, such as left-align and right-align.

**Word Pro: CreateNew method**

{button ,AL(^H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CREATENEW\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new graphic in a document.

**Syntax**

[objectreference].CreateNew(NameOfGraphicType)

**Parameters**

*NameOfGraphicType*

Data type is String.

**Return value**

Integer

**Usage**

## Word Pro: Abilities property

{button ,AL('H\_EDITOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ABILITIES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Controls which version of a document that a specific editor can edit.

### Data Type

Variant (Enumerated)

EditAbil

### Syntax

abilities = [objectreference].Abilities

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpEditAbilEditCurrentOrNewVer (223)	An assigned editor can work in the current document or any new versions of the document.
\$LwpEditAbilEditCurrentVersionOnly (221)	An assigned editor can only work in the current version of the document and cannot edit previous versions of the document.
\$LwpEditAbilEditingNotAllowed (220)	A specific editor cannot edit the document.
\$LwpEditAbilEditNewVersionsOnly (222)	An assigned editor can only work in a new version of a document that is automatically created when the document opens. The editor can review previous versions of the document.

### Usage

Use any one of the above values to determine what version of a document an editor can edit.

## Word Pro: AbsoluteOn property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ABSOLUTEON_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) A Boolean value which indicates whether the top left corner of the layout (the origin) is positioned relative to the PageLayout origin or relative to the parent layout's origin. Default value is False. If this property is set, the coordinates are specified in the AbsoluteXPos and AbsoluteYPos properties.

## Data Type

Integer

## Syntax

[objectreference].AbsoluteOn = absoluteonvalue

absoluteonvalue = [objectreference].AbsoluteOn

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This property allows import filters to process relative page coordinates.

## Word Pro: AbsoluteXPos property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ABSOLUTEXPOS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The top left position X coordinate for a layout.

## Data Type

Long

## Syntax

[objectreference].AbsoluteXPos = absolutexposvalue

absolutexposvalue = [objectreference].AbsoluteXPos

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

The AbsoluteOn property uses this property to set the layout object's top left position. This property allows import filters to process relative page coordinates.



## Word Pro: AbsoluteYPos property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ABSOLUTEYPOS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The top left position Y coordinate for a layout.

### Data Type

Long

### Syntax

[objectreference].AbsoluteYPos = absoluteyposvalue

absoluteyposvalue = [objectreference].AbsoluteYPos

### Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

The AbsoluteOn property uses this property to set the layout object's top left position. This property allows import filters to process relative page coordinates.

## Word Pro: AccessRights property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ACCESSRIGHTS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Restricts access to a layout by excluding the layout from the list of layouts in a collection. This property is not implemented in Word Pro '97.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

accessrightsvalue = [objectreference].AccessRights

[objectreference].AccessRights = accessrightsvalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LtsAccessRightsPublic (1056964814)	Setting this value assigns public access rights to a layout object.
\$LtsAccessRightsPrivate (1056964815)	Setting this value assigns private access rights to a layout object. If you set a layout's access rights to private, you cannot access a layout object by enumerating its collection. Instead, you must access the layout by indexing into its collection with the layout object's name.
\$LtsAccessRightsProtected (1056964816)	Setting this value assigns protected access rights to a layout object. If you set a layout's access rights to protected, you cannot access a layout object by enumerating its collection. Instead, you must access the layout by indexing into its collection with the layout object's name.

## Usage

## **Word Pro: Active property**

{button ,AL(^H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ACTIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

[StatusBar]

A flag that indicates if the status bar is usable.

[ApplicationWindow]

A flag that indicates if the application window is active.

## **Data Type**

[Integer](#)

## **Syntax**

activevalue = [objectreference].Active

## **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

[Application Window]

Default value is None.

## **Usage**

IconBarManager - This property is not valid for IconBarManager.

ApplicationWindow - This property is not implemented for ApplicationWindow in Word Pro '97.

## **Word Pro: ActualName property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACTUALNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

actualnamevalue = [objectreference].ActualName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: AddEnvelopeReturnAddress property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDENVELOPERETURNADDRESS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if the envelope return address is enabled.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

addenvelopereturnaddressvalue = [objectreference].AddEnvelopeReturnAddress

[objectreference].AddEnvelopeReturnAddress = addenvelopereturnaddressvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to the "Disable" field on the General panel of the Word Pro Preferences dialog box. If the legal value for this property is False, Word Pro does not create a return address field when it creates an envelope. If the legal value for this property is True, Word Pro creates a return address field and uses the information on the Personal panel of the Word Pro Preferences dialog box to fill it in.

## **Word Pro: Address1 property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDRESS1\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

address1value = [objectreference].Address1

[objectreference].Address1 = address1value

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: Address2 property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDRESS2\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

address2value = [objectreference].Address2

[objectreference].Address2 = address2value

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: AlignmentChar property**

{button ,AL('H\_ALIGNMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALIGNMENTCHAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

alignmentcharvalue = [objectreference].AlignmentChar

[objectreference].AlignmentChar = alignmentcharvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## Word Pro: AlignmentType property

{button ,AL('H\_ALIGNMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALIGNMENTTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Variant (Enumerated)

AlignmentType

### Syntax

alignmenttypevalue = [objectreference].AlignmentType

[objectreference].AlignmentType = alignmenttypevalue

### Legal values

\$LtsAlignmentHorizCenter (1056964611)

\$LtsAlignmentJustify (1056964613)

\$LtsAlignmentLeft (1056964609)

\$LtsAlignmentRight (1056964610)

\$LtsAlignmentSmart (1056964612)

\$LwpAlignmentTypeAlignRevert (8)

\$LwpAlignmentTypeJustifyall (5)

\$LwpAlignmentTypeNumericleft (6)

\$LwpAlignmentTypeNumericright (7)

### Usage

## **Word Pro: AlignStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALIGNSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

alignstylevalue = [objectreference].AlignStyleName

[objectreference].AlignStyleName = alignstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: AllowAlternateVerification property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALLOWALTERNATEVERIFICATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Provides alternate ways for Word Pro to verify authorized users of a document.

### **Data Type**

[Integer](#)

### **Syntax**

allowalternateverificationvalue = [objectreference].AllowAlternateVerification

[objectreference].AllowAlternateVerification = allowalternateverificationvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

The AllowAlternateVerification property enables you to structure a hierarchy of verification types to verify an assigned user of a document. The verification types are e-mail login, operating system login, and the Word Pro user name. For example, suppose you set up an e-mail login for assigned users of a document. However, a user with an operating system login wants to gain access to that document. When the user attempts to login, Word Pro denies access.

If you set AllowAlternateVerification to True, Word Pro displays the other two verification types: operating system login and the Word Pro user name. The user can now gain access when the operating system login displays. If you set AllowAlternateVerification to False, Word Pro does not display the other verification types, thereby denying access to the user.

Setting the AllowAlternateVerification property is equivalent to choosing File - TeamSecurity and selecting "Allow alternate verification" on the Access panel.

**Word Pro: All property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

allvalue = [objectreference].All

[objectreference].All = allvalue

**Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

**Word Pro: AlternateName property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALTERNATENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

String

**Syntax**

alternatenamevalue = [objectreference].AlternateName

[objectreference].AlternateName = alternatenamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: Always property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALWAYS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

alwaysvalue = [objectreference].Always

[objectreference].Always = alwaysvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: AmikakeName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AMIKAKENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the text background object for the Asian language versions of Word Pro. If you are using an English language version of Word Pro, this property is not available.

**Data Type**

[String](#)

**Syntax**

amikakenamevalue = [objectreference].AmikakeName

[objectreference].AmikakeName = amikakenamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: AmountOfSpaceAbove property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AMOUNTOFSPACEABOVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

amountofspaceabovevalue = [objectreference].AmountOfSpaceAbove

[objectreference].AmountOfSpaceAbove = amountofspaceabovevalue

**Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**



**Word Pro: AmountOfSpaceBelow property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AMOUNTOFSPACEBELOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

amountofspacebelowvalue = [objectreference].AmountOfSpaceBelow

[objectreference].AmountOfSpaceBelow = amountofspacebelowvalue

**Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

## **Word Pro: Amount property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AMOUNT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

amountvalue = [objectreference].Amount

[objectreference].Amount = amountvalue

### **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## Word Pro: AmtTether property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERGROUPLAYOUT_CLASS;H_SUPERLAYOUT_CLASS;H_TABLEGROUPLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERGROUPLAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_AMTTETHER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates the location of a layout object's knot. The knot is the position on the frame from which it is anchored.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

[objectreference].AmtTether = amttethervalue

amttethervalue = [objectreference].AmtTether

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpWhereTypeUpperLeft (1976)	Setting this value attaches the knot in the upper left side of a layout object.
\$LwpWhereTypeMiddleTop (1977)	Setting this value attaches the knot in the middle of the top area of a layout object.
\$LwpWhereTypeLayoutUpperRight (1978)	Setting this value attaches the knot in the upper right side of a layout object.
\$LwpWhereTypeMiddleLeft (1979)	Setting this value attaches the knot in the middle of the left side of a layout object.
\$LwpWhereTypeMiddleRight (1980)	Setting this value attaches the knot in the middle of the right side of a layout object.
\$LwpWhereTypeLowerLeft (1981)	Setting this value attaches the knot in the lower left side of a layout object.
\$LwpWhereTypeMiddleBottom (1982)	Setting this value attaches the knot in the middle of the bottom area of a layout object.
\$LwpWhereTypeLowerRight (1983)	Setting this value attaches the knot in the lower right side of a layout object.
\$LwpWhereTypeMiddle (1984)	Setting this value attaches the knot in the middle of a layout object.

## Usage

The AmtTether property applies only to frame and table layout objects. In Word Pro, this property is represented by the "Tie anchor to frame setting" in the Placement and Anchoring Options dialog box. Use this property in conjunction with the AmtToTetherFrom property in order to properly place the layout object's knot.

## Word Pro: AmtToRotateContent property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAY  
OUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;  
H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PA  
GELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_  
SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLAS  
S;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_AMTTOROTATECONTENT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to rotate the content of a layout in tenths of degrees.

## Data Type

[Integer](#)

## Syntax

amtrotatecontentvalue = [objectreference].AmtToRotateContent

[objectreference].AmtToRotateContent = amtrotatecontentvalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

Use this property when you need to rotate graphics in a layout.

## Word Pro: AmtToTetherFrom property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_AMTTOTETHERFROM_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether a layout object's knot is tied to the interior, border, or exterior of the object. The knot is the position on the frame from which it is anchored.

## Data Type

The data type for this property is Variant which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

amttotetherfromvalue = [objectreference].AmtToTetherFrom

[objectreference].AmtToTetherFrom = amttotetherfromvalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpWhereTypeUpperLeft (1976)	Setting this value attaches the knot to the interior of a layout object.
\$LwpWhereTypeMiddleTop (1977)	Setting this value attaches the knot to the exterior of a layout object.
\$LwpWhereTypeLayoutUpperRight (1978)	Setting this value attaches the knot to the border of a layout object.

## Usage

The AmtToTetherFrom property applies only to frame and table layout objects. In Word Pro, this property is represented by the "Tie anchor to frame setting" in the Placement and Anchoring Options dialog box. Use this property in conjunction with the AmtTether property in order to properly place the layout object's knot.

## Word Pro: AnyOleObjects property

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ANYOLEOBJECTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether there are any OLE objects present in the Foundry of a particular Division object. Only available in early releases of Word Pro 96 for Windows 3.1 and Windows 95. To avoid incompatibilities with more recent releases of Word Pro, use the Count method on the OLEObjects property instead of this property. If the count is 0, there are no OLE objects in the Foundry.

### Data Type

Integer

### Syntax

anyoleobjectsvalue = [objectreference].AnyOleObjects

### Legal values

### Usage

This property is only available in early versions of Word Pro 96, so you should use the Count method on the OleObjects property instead. However, if you use this property, its value will reflect the presence or absence of OLE objects in the specified Foundry object. There are four different Foundry objects: AppFoundry and TempFoundry (both on the WPAApplication object), Foundry (on each Division object), and Foundry (on each TextDocument object). WPAApplication also has a Foundry property which you can use to access the currently active Division object's Foundry.

WPAApplication.AppFoundry.AnyOleObjects indicates the presence (or absence) of OLE objects in the Word Pro Clipboard.

WPAApplication.TempFoundry.AnyOleObjects indicates the presence (or absence) of OLE objects in the TempFoundry Foundry object.

Division.Foundry.AnyOleObjects indicates the presence (or absence) of OLE objects in the specified division of a Word Pro document.

WPAApplication.Foundry.AnyOleObjects indicates the presence (or absence) of OLE objects in the currently active division of the currently active Word Pro document. This Foundry object is the same as seen under Division.Foundry.

TextDocument.Foundry is never used, so the value of the AnyOleObjects property is always False.

## **Word Pro: ApplyAdjectivePos property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYADJECTIVEPOS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking adjective positions.

### **Data Type**

[Integer](#)

### **Syntax**

applyadjectiveposvalue = [objectreference].ApplyAdjectivePos

[objectreference].ApplyAdjectivePos = applyadjectiveposvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyAdjectNounPart property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYADJECTNOUNPART\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking adjective/noun parts.

### **Data Type**

[Integer](#)

### **Syntax**

applyadjectnounpartvalue = [objectreference].ApplyAdjectNounPart

[objectreference].ApplyAdjectNounPart = applyadjectnounpartvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyAgreementWithHereThere property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYAGREEMENTWITHHERETHERE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for agreement between usage of the words "here" and "there."

### Data Type

[Integer](#)

### Syntax

applyagreementwithheretherevalue = [objectreference].ApplyAgreementWithHereThere  
[objectreference].ApplyAgreementWithHereThere = applyagreementwithheretherevalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Agreement with 'here'/'there' " in the "Rule type" field on the Rules panel.

This rule flags errors of agreement between verbs and their predicate nouns when the sentence has "here" or "there" as its apparent subject. A predicate noun identifies or restates the subject of the sentence. When the apparent subject of the sentence is "here" or "there," the verb must agree with the predicate noun. For example, one rule will flag the sentence, "Here comes the two men who can help us," because the singular verb ("comes") does not agree with the plural predicate noun ("men").

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyAnglicisms property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYANGLICISMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyanglicismsvalue = [objectreference].ApplyAnglicisms

[objectreference].ApplyAnglicisms = applyanglicismsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro.

## Word Pro: ApplyArchaicExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYARCHAICEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking archaic expressions.

### Data Type

[Integer](#)

### Syntax

applyarchaicexpressionsvalue = [objectreference].ApplyArchaicExpressions

[objectreference].ApplyArchaicExpressions = applyarchaicexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Archaic expressions" in the "Rule type" field on the Rules panel.

This rule flags words and expressions that are no longer current in standard usage. These words or expressions may be appropriate in certain contexts, but might seem stilted or awkward in everyday writing. They should be replaced with contemporary equivalents, whenever possible. For example, the sentence, "Would you perchance be free for lunch on Tuesday?" can be revised using the more contemporary term, "possibly," in place of "perchance."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyArticleAgreement property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYARTICLEAGREEMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking "A" and "An" article agreements.

### **Data Type**

[Integer](#)

### **Syntax**

applyarticleagreementvalue = [objectreference].ApplyArticleAgreement [objectreference].ApplyArticleAgreement = applyarticleagreementvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "A vs. An " in the "Rule type" field on the Rules panel.

This rule flags incorrect indefinite articles (a/an) in noun phrases. Words that begin with a vowel usually take "an" as the indefinite article (an army), and words that begin with a consonant usually take "a" as the indefinite article (a carrot).

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBadComparatives property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBADCOMPARATIVES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybadcomparativesvalue =[objectreference].ApplyBadComparatives [objectreference].ApplyBadComparatives = applybadcomparativesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBadInflection property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBADINFLECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybadinflectionvalue = [objectreference].ApplyBadInflection

[objectreference].ApplyBadInflection = applybadinflectionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBadNounGender property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBADNOUNGENDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybadnougendervalue = [objectreference].ApplyBadNounGender

[objectreference].ApplyBadNounGender = applybadnougendervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBadNoun property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBADNOUN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyBadNounvalue = [objectreference].ApplyBadNoun

[objectreference].ApplyBadNoun = applyBadNounvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyBadPlural property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBADPLURAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybadpluralvalue = [objectreference]. ApplyBadPlural

[objectreference]. ApplyBadPlural= applybadpluralvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBadPrepositions property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBADPREPOSITIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybadprepositionsvalue = [objectreference].ApplyBadPrepositions

[objectreference].ApplyBadPrepositions = applybadprepositionsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBelgianExpression property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBELGIANEXPRESSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybelgianexpressionvalue = [objectreference].ApplyBelgianExpression

[objectreference].ApplyBelgianExpression = applybelgianexpressionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBorrowedForeign property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBORROWEDFOREIGN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyborrowedforeignvalue = [objectreference].ApplyBorrowedForeign

[objectreference].ApplyBorrowedForeign = applyborrowedforeignvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyBureuaJargon property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYBUREUAJARGON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applybureujargonvalue = [objectreference].ApplyBureuaJargon

[objectreference].ApplyBureuaJargon = applybureujargonvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyCalque property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCALQUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applycalquevalue = [objectreference].ApplyCalque

[objectreference].ApplyCalque = applycalquevalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Spanish.

## **Word Pro: ApplyCapitalizationCheck property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCAPITALIZATIONCHECK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking capitalization.

### **Data Type**

[Integer](#)

### **Syntax**

applycapitalizationcheckvalue = [objectreference].ApplyCapitalizationCheck

[objectreference].ApplyCapitalizationCheck = applycapitalizationcheckvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Capitalization errors" in the "Rule type" field on the Rules panel.

This rule flags the most common capitalization errors, including uncapitalized proper names, uncapitalized salutations, and incorrectly capitalized or uncapitalized names of days, months, seasons, holidays and abbreviations. It also flags sentences that begin with a lowercase letter. For example, this rule flags "chicago" as "Chicago," "best wishes" as "Best Wishes," and "Memorial day" as "Memorial Day."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyClauseErrors property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCLAUSEERRORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking clause errors.

### Data Type

[Integer](#)

### Syntax

applyclauseerrorsvalue = [objectreference].ApplyClauseErrors

[objectreference].ApplyClauseErrors = applyclauseerrorsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Clause errors" in the "Rule type" field on the Rules panel.

This rule flags general errors of sentence structure, such as run-on sentences and sentence fragments. It checks to see that conjunctions are used correctly and that correct punctuation appears between clauses. For example, this rule will flag the sentence, "We chopped up fruit, and we diced the potatoes, and we made a pie crust," since only one "and" is necessary when three clauses appear in sequence.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyCliches property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCLICHES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking expressions that are cliches.

### Data Type

[Integer](#)

### Syntax

applyclichesvalue = [objectreference].ApplyCliches

[objectreference].ApplyCliches = applyclichesvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Cliches" in the "Rule type" field on the Rules panel.

This rule flags clichés, colorful expressions used so often that they have lost their original force. Although clichés may occasionally be appropriate, you should avoid using them casually or excessively. Sometimes a cliché can be replaced by a more direct term. In other cases, the sentence must be rephrased to avoid the cliché. For example, the phrase, "make a mountain out of a molehill," might become "exaggerate" or "overreact."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyColloquialExpression property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCOLLOQUIALEXPRESSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking colloquial expressions.

### **Data Type**

[Integer](#)

### **Syntax**

applycolloquialexpressionvalue = [objectreference].ApplyColloquialExpression

[objectreference].ApplyColloquialExpression = applycolloquialexpressionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyCommonlyConfusedWords property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCOMMONLYCONFUSEDWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking words that are commonly confused.

### Data Type

[Integer](#)

### Syntax

applycommonlyconfusedwordsvalue = [objectreference].ApplyCommonlyConfusedWords

[objectreference].ApplyCommonlyConfusedWords = applycommonlyconfusedwordsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Commonly confused words" in the "Rule type" field on the Rules panel.

This rule flags commonly confused words that have similar, though not identical, pronunciations. The confused pairs include words that involve confusion between a noun and a verb. For example, the rule will flag the sentence, "You would be wise to seek legal advise before signing a contract," because the verb, "advise," is mistakenly used instead of the noun, "advice."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyCommonMisspell property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCOMMONMISPELL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking common misspellings.

### **Data Type**

[Integer](#)

### **Syntax**

applycommonmisspellvalue = [objectreference].ApplyCommonMisspell [objectreference].ApplyCommonMisspell = applycommonmisspellvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyComplexWords property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCOMPLEXWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applycomplexwordvalue = [objectreference].ApplyComplexWords

[objectreference].ApplyComplexWords = applycomplexwordvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyConfusedEasy property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONFUSEDEASY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applyconfusedeasyvalue = [objectreference].ApplyConfusedEasy

[objectreference].ApplyConfusedEasy = applyconfusedeasyvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, French.

## Word Pro: ApplyConfusedEnglish property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONFUSEDENGLISH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applyconfusedenglishvalue = [objectreference].ApplyConfusedEnglish [objectreference].ApplyConfusedEnglish = applyconfusedenglishvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, French.

## **Word Pro: ApplyConfusedHard property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONFUSEDHARD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyconfusedhardvalue = [objectreference].ApplyConfusedHard

[objectreference].ApplyConfusedHard = applyconfusedhardvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, French.



## **Word Pro: ApplyConfusedMedium property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONFUSEDMEDIUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyconfusedmediumvalue = [objectreference].ApplyConfusedMedium [objectreference].ApplyConfusedMedium = applyconfusedmediumvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, French.

## Word Pro: ApplyConfusedVerb property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONFUSEDVERB\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applyconfusedverbvalue = [objectreference].ApplyConfusedVerb

[objectreference].ApplyConfusedVerb = applyconfusedverbvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Dutch.

## **Word Pro: ApplyConsecutiveNouns property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONSECUTIVENOUNS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking consecutive nouns.

### **Data Type**

[Integer](#)

### **Syntax**

applyconsecutivenounsvalue = [objectreference].ApplyConsecutiveNouns

[objectreference].ApplyConsecutiveNouns = applyconsecutivenounsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyContractions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYCONTRACTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking contractions.

### Data Type

[Integer](#)

### Syntax

applycontractionsvalue = [objectreference].ApplyContractions

[objectreference].ApplyContractions = applycontractionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Contractions" in the "Rule type" field on the Rules panel.

This rule flags contractions and recommends the appropriate expanded forms. For example, one rule will flag the sentence, "I've completed the course," and suggest replacing "I've" with "I have." Contractions are acceptable in many written contexts, especially if you are striving for a conversational tone. They may be inappropriate, however, in some formal documents. This rule helps you identify and revise contractions when you are working in formal documents.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyDerogatory property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYDEROGATORY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyderogatoryvalue = [objectreference].ApplyDerogatory

[objectreference].ApplyDerogatory = applyderogatoryvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyDifferentPrep property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYDIFFERENTPREP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking incorrect prepositions.

### **Data Type**

[Integer](#)

### **Syntax**

applydifferentprepvalue = [objectreference].ApplyDifferentPrep

[objectreference].ApplyDifferentPrep = applydifferentprepvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Inappropriate prepositions" in the "Rule type" field on the Rules panel.

This rule flags expressions that include an incorrect preposition and offers the appropriate preposition as a correction. For example, the rule will flag "adhere by" as "adhere to," "center around" as "center on," and so on.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyDoubleNegative property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYDOUBLENEGATIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking double negatives.

### **Data Type**

[Integer](#)

### **Syntax**

applydoublenegativevalue = [objectreference].ApplyDoubleNegative [objectreference].ApplyDoubleNegative = applydoublenegativevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Double negatives" in the "Rule type" field on the Rules panel.

This rule flags confusing or awkward phrases that may contain more than one negative word. For example, a sentence may be flagged that uses "can't hardly" instead of "cannot" or "can hardly," and "in no uncertain terms" instead of "clearly" or "specifically."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyDoublePlural property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYDOUBLEPLURAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applydoublepluralvalue = [objectreference].ApplyDoublePlural

[objectreference].ApplyDoublePlural = applydoublepluralvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyDoubleWordCheck property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYDOUBLEWORDCHECK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking double words.

### Data Type

[Integer](#)

### Syntax

applydoublewordcheckvalue = [objectreference].ApplyDoubleWordCheck

[objectreference].ApplyDoubleWordCheck = applydoublewordcheckvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Doubled words" in the "Rule type" field on the Rules panel.

This rule flags sequences of two identical words. Language-specific exceptions are made for legitimately doubled words (for example, "had had" as in "We had had the same discussion before," and "that that" as in "She thought that that problem had been solved."). This rule also flags a succession of articles ("the" and "a"), possessive pronouns ("my" and "his"), and similar words that must not be followed by a word of the same type.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyElision property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYELISION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking elisions (omitting something in a word, such as a final or initial pronunciation).

### Data Type

[Integer](#)

### Syntax

applyelisionvalue = [objectreference].ApplyElision

[objectreference].ApplyElision = applyelisionvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyEnglishDerived property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYENGLISHDERIVED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyenglishderivedvalue = [objectreference].ApplyEnglishDerived

[objectreference].ApplyEnglishDerived = applyenglishderivedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyEnglishWords property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYENGLISHWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyenglishwordsvvalue = [objectreference].ApplyEnglishWords

[objectreference].ApplyEnglishWords = applyenglishwordsvvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyExotic property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYEXOTIC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyexoticvalue = [objectreference].ApplyExotic

[objectreference].ApplyExotic = applyexoticvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyExtraPrepositionCheck property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYEXTRAPREPOSITIONCHECK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applyextraprepositioncheckvalue = [objectreference].ApplyExtraPrepositionCheck

[objectreference].ApplyExtraPrepositionCheck = applyextraprepositioncheckvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyFalseFriend property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYFALSEFRIEND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applyfalsefriendvalue = [objectreference].ApplyFalseFriend

[objectreference].ApplyFalseFriend = applyfalsefriendvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Spanish.

## **Word Pro: ApplyFemaleOccupation property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYFEMALEOCCUPATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyfemaleoccupationvalue = [objectreference].ApplyFemaleOccupation [objectreference].ApplyFemaleOccupation  
= applyfemaleoccupationvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyFixedExpression property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYFIXEDEXPRESSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyfixedexpressionvalue = [objectreference].ApplyFixedExpression

[objectreference].ApplyFixedExpression = applyfixedexpressionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyForeignWord property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYFOREIGNWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

Integer

### **Syntax**

applyforeignwordvalue = [objectreference].ApplyForeignWord

[objectreference].ApplyForeignWord = applyforeignwordvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyFormalTerms property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYFORMALTERMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyformaltermsvalue = [objectreference].ApplyFormalTerms

[objectreference].ApplyFormalTerms = applyformaltermsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyFormatErrors property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYFORMATERRORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking formatting errors.

### Data Type

[Integer](#)

### Syntax

applyformaterrorsvalue = [objectreference].ApplyFormatErrors

[objectreference].ApplyFormatErrors = applyformaterrorsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Formatting errors" in the "Rule type" field on the Rules panel.

This rule checks the format of numbers (placement of periods/commas, endings of ordinal numbers, spelling of fractions/other numbers), dates (use of cardinal/ordinal numbers), times (use of abbreviations and punctuation marks), currency/other symbols, and addresses.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyGallicisms property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYGALLICISMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applygallicismsvalue = [objectreference].ApplyGallicisms

[objectreference].ApplyGallicisms = applygallicismsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyGenderExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYGENDEREXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking gender expressions.

### Data Type

[Integer](#)

### Syntax

applygenderexpressionsvalue = [objectreference].ApplyGenderExpressions

[objectreference].ApplyGenderExpressions = applygenderexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Gender-specific expressions" in the "Rule type" field on the Rules panel.

This rule flags gender-specific terms, such as names of occupations or professions that may unnecessarily indicate a person's gender. For professions that were formerly dominated by women but now include men, gender-neutral designations are preferred. For example, one rule will flag the sentence, "The guest speaker was a popular local poetess," and suggest the word, "poet" as a substitute.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyGermanisms property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYGERMANISMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applygermanismsvalue = [objectreference].ApplyGermanisms

[objectreference].ApplyGermanisms = applygermanismsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyHomoGraphs property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMOGRAPHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking homographic expressions (one of two or more words that have the same spelling but differ in origin, meaning, and sometimes pronunciation).

### **Data Type**

[Integer](#)

### **Syntax**

applyhomographsvalue = [objectreference].ApplyHomoGraphs

[objectreference].ApplyHomoGraphs = applyhomographsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Italian.



## **Word Pro: ApplyHomonymsEasy property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMONYMSEASY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyhomonymseasyvalue = [objectreference].ApplyHomonymsEasy [objectreference].ApplyHomonymsEasy = applyhomonymseasyvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyHomonymsHard property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMONYMSHARD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyhomonymshardvalue = [objectreference].ApplyHomonymsHard

[objectreference].ApplyHomonymsHard = applyhomonymshardvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyHomonyms property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMONYMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking homonyms.

### Data Type

[Integer](#)

### Syntax

applyhomonymsvalue = [objectreference].ApplyHomonyms

[objectreference].ApplyHomonyms = applyhomonymsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Homonyms" in the "Rule type" field on the Rules panel.

This rule flags homonyms or near-homonyms, words that may be confused because they sound alike (for example, "principle" and "principal," "complacent" and "complaisant"). Note that all occurrences of these words will be flagged because there are no contextual clues to distinguish their usage. You must consult the explanations given in the error message to determine whether your usage is correct.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyHomoPhone1 property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMOPHONE1\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking homophonic words (one or two or more words, such as "night" and "knight," that are pronounced the same but differ in meaning, origin, and sometimes spelling).

### **Data Type**

[Integer](#)

### **Syntax**

applyhomophone1value = [objectreference].ApplyHomoPhone1

[objectreference].ApplyHomoPhone1 = applyhomophone1value

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Spanish.

## **Word Pro: ApplyHomoPhone2 property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMOPHONE2\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking homophonic words (one or two or more words, such as "night" and "knight," that are pronounced the same but differ in meaning, origin, and sometimes spelling).

### **Data Type**

[Integer](#)

### **Syntax**

applyhomophone2value = [objectreference].ApplyHomoPhone2

[objectreference].ApplyHomoPhone2 = applyhomophone2value

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Spanish.

## **Word Pro: ApplyHomoPhones property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYHOMOPHONES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking homophonic words (one or two or more words, such as "night" and "knight," that are pronounced the same but differ in meaning, origin, and sometimes spelling).

### **Data Type**

[Integer](#)

### **Syntax**

applyhomophonesvalue = [objectreference].ApplyHomoPhones

[objectreference].ApplyHomoPhones = applyhomophonesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Swedish.

## **Word Pro: ApplyIncorrectPlural property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYINCORRECTPLURAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking incorrect plurals.

### **Data Type**

[Integer](#)

### **Syntax**

applyincorrectpluralvalue = [objectreference].ApplyIncorrectPlural

[objectreference].ApplyIncorrectPlural = applyincorrectpluralvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyInformalExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYINFORMALEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking informal expressions.

### Data Type

[Integer](#)

### Syntax

applyinformalexpressionsvalue = [objectreference].ApplyInformalExpressions

[objectreference].ApplyInformalExpressions = applyinformalexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Informal expressions" in the "Rule type" field on the Rules panel.

This rule flags words and expressions that are more appropriate in speech than in writing. This rule offers a less casual alternative or suggests rephrasing the sentence to eliminate the expression. For example, the phrase "a shot at" can be replaced by "a chance to."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyJargonWords property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYJARGONWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking jargon words and expressions.

### **Data Type**

[Integer](#)

### **Syntax**

applyjargonwordsvalue = [objectreference].ApplyJargonWords

[objectreference].ApplyJargonWords = applyjargonwordsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Jargon" in the "Rule type" field on the Rules panel.

This rule flags words and expressions that belong to a specific technical vocabulary (medicine, science, music, and so on), but are inappropriate when used in general writing (for example, "input").

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

### **Word Pro: ApplyLowercaseAdjective property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYLOWERCASEADJECTIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

H\_WP\_SCALAR\_DATA\_TYPES\_OVER

Integer

### **Syntax**

applylowercaseadjectivevalue = [objectreference].ApplyLowercaseAdjective

[objectreference].ApplyLowercaseAdjective = applylowercaseadjectivevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, German.

## Word Pro: ApplyLowercaseColor property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYLOWERCASECOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applylowercasecolorvalue = [objectreference].ApplyLowercaseColor

[objectreference].ApplyLowercaseColor = applylowercasecolorvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, German.

## **Word Pro: ApplyLowercaseNumbers property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYLOWERCASENUMBERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applylowercasenumbersvalue = [objectreference].ApplyLowercaseNumbers

[objectreference].ApplyLowercaseNumbers = applylowercasenumbersvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, German.

## **Word Pro: ApplyLowercasePhrases property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYLOWERCASEPHRASES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applylowercasephrasesvalue = [objectreference].ApplyLowercasePhrases [objectreference].ApplyLowercasePhrases  
= applylowercasephrasesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, German.

## Word Pro: ApplyLowercasePronouns property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYLOWERCASEPRONOUNS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applylowercasepronounsvalue = [objectreference].ApplyLowercasePronouns

[objectreference].ApplyLowercasePronouns = applylowercasepronounsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, German.

## Word Pro: ApplyMassVsCount property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYMASSVSCOUNT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking errors of mass/count agreement.

### Data Type

[Integer](#)

### Syntax

applymassvscountvalue = [objectreference].ApplyMassVsCount

[objectreference].ApplyMassVsCount = applymassvscountvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Misspelled expressions" in the "Rule type" field on the Rules panel.

This rule flags errors of mass/count agreement which conflict with the number the noun represents (singular or plural) and the modifying adjectives. For example, one rule flags the sentence, "There are less mistakes in this document," because the adjective, "fewer," not "less" is the correct one to use with the plural noun, "mistakes."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyMisspelledExpressions property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYMISSPELLEDEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking misspelled expressions.

### **Data Type**

[Integer](#)

### **Syntax**

applymisspelledexpressionsvalue = [objectreference].ApplyMisspelledExpressions

[objectreference].ApplyMisspelledExpressions = applymisspelledexpressionsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Misspelled expressions" in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyMisspelledForeignExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYMISPELLEDFOREIGNEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking misspelled foreign expressions.

### Data Type

[Integer](#)

### Syntax

applymisspelledforeignexpressionsvalue = [objectreference].ApplyMisspelledForeignExpressions

[objectreference].ApplyMisspelledForeignExpressions = applymisspelledforeignexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Misspelled foreign expressions" in the "Rule type" field on the Rules panel.

This rule flags misspelled foreign expressions. It may also flag typing errors that make a word look like a foreign expression (for example, "esprit di corps" to "esprit de corps.")

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

### **Word Pro: ApplyMisspelledItalian property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYMISPELLEDITALIAN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applymisspelleditalianvalue = [objectreference].ApplyMisspelledItalian

[objectreference].ApplyMisspelledItalian = applymisspelleditalianvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyMisspelledWords property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYMISPELLEDWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking misspelled words.

### **Data Type**

[Integer](#)

### **Syntax**

applymisspelledwordsvvalue = [objectreference].ApplyMisspelledWords

[objectreference].ApplyMisspelledWords = applymisspelledwordsvvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyMisusedWords property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYMISUSEDWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking misused words.

### Data Type

[Integer](#)

### Syntax

applymisusedwordsvvalue = [objectreference].ApplyMisusedWords

[objectreference].ApplyMisusedWords = applymisusedwordsvvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Misused words" in the "Rule type" field on the Rules panel.

This rule flags words or phrases that are often confused with similar words or phrases (for example, "elude to" instead of "allude to," "sit the books on the chair" instead of "set the books on the chair," and so on). The confused expressions should be used in different constructions.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNonStandardExpression property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNONSTANDARDEXPRESSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking non-standard expressions.

### **Data Type**

[Integer](#)

### **Syntax**

applynonstandardexpressionvalue = [objectreference].ApplyNonStandardExpression

[objectreference].ApplyNonStandardExpression = applynonstandardexpressionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Nonstandard terms" in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNonStandardModifiers property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNONSTANDARDMODIFIERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking non-standard modifiers.

### **Data Type**

[Integer](#)

### **Syntax**

applynonstandardmodifiersvalue = [objectreference].ApplyNonStandardModifiers

[objectreference].ApplyNonStandardModifiers = applynonstandardmodifiersvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Nonstandard modifiers" in the "Rule type" field on the Rules panel.

This rule flags errors of modification, such as using adjectives rather than adverbs to modify verbs. For example, the rule will flag the sentence, "His new car really drives good," since "good" is an adjective mistakenly used in place of the adverb, "well." This rule also checks to determine whether the hyphenation of modifiers conforms to standard usage.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyNoudModifierOrderCheck property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNOUDMODIFIERORDERCHECK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking errors in word order.

### Data Type

[Integer](#)

### Syntax

applynoudmodifierordercheckvalue = [objectreference].ApplyNoudModifierOrderCheck

[objectreference].ApplyNoudModifierOrderCheck = applynoudmodifierordercheckvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Word order errors" in the "Rule type" field on the Rules panel.

This rule flags the incorrect order of certain words that modify nouns, for example, "my both children" instead of "both my children."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyNounConsistency property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNOUNCONSISTENCY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking noun consistency errors.

### Data Type

[Integer](#)

### Syntax

applynounconsistencyvalue = [objectreference].ApplyNounConsistency [objectreference].ApplyNounConsistency = applynounconsistencyvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Noun phrase consistency errors" in the "Rule type" field on the Rules panel.

This rule flags errors of number agreement within noun phrases. A noun phrase consists of a noun and the words that modify it, for example, "this old man," "that red bicycle," "a tall building." Certain modifiers, such as "this," "that," and "a" are singular and must be used with singular nouns. Other modifiers, such as "these," "those," "both," and "many" must be used with plural nouns. This rule will flag the sentence, "These five machine are still under warranty," because "these" is plural and "machine" is singular.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyNounPhraseAgree property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNOUNPHRASEAGREE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynounphraseagreevalue = [objectreference].ApplyNounPhraseAgree

[objectreference].ApplyNounPhraseAgree = applynounphraseagreevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSAdjective property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSADJECTIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynsadjectivevalue = [objectreference].ApplyNSAdjective

[objectreference].ApplyNSAdjective = applynsadjectivevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSClause property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSCLAUSE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynsclausevalue = [objectreference].ApplyNSClause

[objectreference].ApplyNSClause = applynsclausevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSCompare property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSCOMPARE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynscomparevalue = [objectreference].ApplyNSCompare

[objectreference].ApplyNSCompare = applynscomparevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplySplitInfinitives property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSPLITINFINITIVES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applysplitinfinitivesvalue = [objectreference].ApplySplitInfinitives

[objectreference].ApplySplitInfinitives = applysplitinfinitivesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

Example: AbsoluteTextOrientation property

'This example script has not yet been created.'

Example: ActualEnumName property

'This example script has not yet been created.'

Example: AfidClassName property

'This example script has not yet been created.'



Example: Align property

'This example script has not yet been created.'

Example: AllowListEdit property

'This example script has not yet been created.'

Example: AllowListMultiValues property

'This example script has not yet been created.'

Example: AmikakeType property

'This example script has not yet been created.'

Example: AmountOfSpaceAboveLine property

'This example script has not yet been created.'

Example: AnyOLEDBLinks property

'This example script has not yet been created.'

Example: ApplykSplitInfinitives property

'This example script has not yet been created.'

Example: AsciiCodePage property

'This example script has not yet been created.'



Example: BackColorIndex property

'This example script has not yet been created.'

Example: BackgroundSpellingOn property

'This example script has not yet been created.'

Example: BaseTables property

'This example script has not yet been created.'

Example: BaseURL property

'This example script has not yet been created.'

Example: BinNumber property

'This example script has not yet been created.'

Example: BookletPrinting property

'This example script has not yet been created.'

Example: CanCreatePreviewBitmap property

'This example script has not yet been created.'

Example: Center property

'This example script has not yet been created.'



Example: ChangeKeyboardToLanguage property  
'This example script has not yet been created.'

Example: ChangeTextToMatchkeyboard property  
'This example script has not yet been created.'

Example: ClassId property

'This example script has not yet been created.'

Example: CreatePreviewBitmap property

'This example script has not yet been created.'

Example: DataFormat property

'This example script has not yet been created.'

Example: DdeEnabled property

'This example script has not yet been created.'

Example: DefaultDropCapStyleDescription property  
'This example script has not yet been created.'

Example: DefaultNewCategory property

'This example script has not yet been created.'



Example: DiagonalLines property

'This example script has not yet been created.'

Example: DisplayAsIcon property

'This example script has not yet been created.'

Example: DriverName property

'This example script has not yet been created.'

Example: DropCapPosition property

'This example script has not yet been created.'

Example: DropCapStyleName property

'This example script has not yet been created.'

Example: DropCapStyles property

'This example script has not yet been created.'

Example: DropCaps property

'This example script has not yet been created.'

Example: ExternallyControlledUndo property  
'This example script has not yet been created.'



Example: FaceName property

'This example script has not yet been created.'

Example: FileFormat property

'This example script has not yet been created.'

Example: FinishedSpellChecking property

'This example script has not yet been created.'

Example: FirstDivision property

'This example script has not yet been created.'

Example: FirstSpellString property

'This example script has not yet been created.'

Example: FontMatching property

'This example script has not yet been created.'

Example: FooterStyleName property

'This example script has not yet been created.'

Example: FooterStyles property

'This example script has not yet been created.'



Example: ForeColorIndex property

'This example script has not yet been created.'

Example: GotoPageLoadInBackground property  
'This example script has not yet been created.'

Example: GraphicPaths property

'This example script has not yet been created.'

Example: GraphicPath property

'This example script has not yet been created.'

Example: HasIndex property

'This example script has not yet been created.'

Example: HasTOC property

'This example script has not yet been created.'

Example: HeaderStyleName property

'This example script has not yet been created.'

Example: HeaderStyles property

'This example script has not yet been created.'



Example: IDispatch property

'This example script has not yet been created.'

Example: IsAmikake property

'This example script has not yet been created.'

Example: IsChangedOtherThanLinkTo property  
'This example script has not yet been created.'

Example: IsChangedToLinkTo property

'This example script has not yet been created.'

Example: IsChartLink property

'This example script has not yet been created.'

Example: IsEndnoteDivision property

'This example script has not yet been created.'

Example: IsLocked property

'This example script has not yet been created.'

Example: IsPrompting property

'This example script has not yet been created.'



Example: IsSmartEditEnabled property

'This example script has not yet been created.'

Example: IsStyle property

'This example script has not yet been created.'

Example: IsSymbolic property

'This example script has not yet been created.'

Example: IsViewRubyMarks property

'This example script has not yet been created.'

Example: IsViewStatusSpell property

'This example script has not yet been created.'

Example: LastDivision property

'This example script has not yet been created.'

Example: LayerName property

'This example script has not yet been created.'

Example: Layer property

'This example script has not yet been created.'



Example: LinkFrame property

'This example script has not yet been created.'

Example: LockResult property

'This example script has not yet been created.'

Example: MergeFileType property

'This example script has not yet been created.'

Example: MergeToFile property

'This example script has not yet been created.'

Example: MetafilePict property

'This example script has not yet been created.'

Example: NextSpellString property

'This example script has not yet been created.'

Example: NonUserDocument property

'This example script has not yet been created.'

Example: NumberOfCharacters property

'This example script has not yet been created.'



Example: NumberOfLines property

'This example script has not yet been created.'

Example: NumberOfMergeConditions property

'This example script has not yet been created.'

Example: NumLinesOfSpaceAboveLine property  
'This example script has not yet been created.'

Example: OCXDesignMode property

'This example script has not yet been created.'

Example: Ole1Object property

'This example script has not yet been created.'

Example: OLEAutomation property

'This example script has not yet been created.'

Example: OLEEnabled property

'This example script has not yet been created.'

Example: OleMinHeight property

'This example script has not yet been created.'



Example: OleObjectSize property

'This example script has not yet been created.'

Example: ParagraphHasDropCap property

'This example script has not yet been created.'

Example: Placement property

'This example script has not yet been created.'

Example: Plain property

'This example script has not yet been created.'

Example: ProgID property

'This example script has not yet been created.'

Example: ReadCompressed property

'This example script has not yet been created.'

Example: RetainNameOfImportedFile property  
'This example script has not yet been created.'

Example: Shape property

'This example script has not yet been created.'



Example: ShowBubbleHelp property

'This example script has not yet been created.'

Example: ShowFileNew property

'This example script has not yet been created.'

Example: SingleCellSelected property

'This example script has not yet been created.'

Example: Span property

'This example script has not yet been created.'

Example: SpellString property

'This example script has not yet been created.'

Example: ~~StrikeThrough~~ property

'This example script has not yet been created.'

Example: Tile property

'This example script has not yet been created.'

Example: TypeAboveLine property

'This example script has not yet been created.'



Example: UpdateOnLoad property

'This example script has not yet been created.'

Example: UserDefinedFilter property

'This example script has not yet been created.'

Example: WaterMarkName property

'This example script has not yet been created.'

Example: WidthInLongtwips property

'This example script has not yet been created.'

## Word Pro: AbsoluteTextOrientation property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCO  
LSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ABSOLUTETEXTORIENTATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the orientation of text in a container.

## Data Type

The data type for this property is Integer. It will always contain one of the values listed below, under Legal Values.

## Syntax

absolutetextorientationvalue = [objectreference].AbsoluteTextOrientation

## Legal values

The legal values for this property are listed below:

<u>Value</u>	<u>Effect</u>
0	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientLefttorightToptobottom
1	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientToptobottomRighttoleft
2	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientRighttoleftBottomtotop
3	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientBottomtotopLefttoright

## Usage

**Word Pro: ActualEnumName property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACTUALENUMNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: AfidClassName property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AFIDCLASSNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: Align property**

{button ,AL(`H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ALIGN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: AllowListEdit property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALLOWLISTEDIT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: AllowListMultiValues property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ALLOWLISTMULTIVALUES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: AmikakeType property**

{button ,AL('H\_AMIKAKE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AMIKAKETYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: AmountOfSpaceAboveLine property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AMOUNTOFSPACEABOVELINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: AnyOLEDELinks property**

{button ,AL('H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ANYOLEDELINKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: ApplykSplitInfinitives property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYKSPPLITINFINITIVES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: AsciiCodePage property**

{button ,AL(`H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ASCII\_CODEPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

### **Word Pro: BackColorIndex property**

{button ,AL('H\_BACKGROUND\_CLASS;H\_BORDER\_CLASS;H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BACKCOLORINDEX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

#### **Data Type**

Unknown

#### **Syntax**

Unknown

#### **Legal values**

Unknown

#### **Usage**



## **Word Pro: BackgroundSpellingOn property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BACKGROUNDSPELLINGON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: BaseTables property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BASETABLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: BaseURL property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BASEURL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## Word Pro: BinNumber property

{button ,AL(^H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS';0)} [See list of classes](#)

{button ,AL(^H\_BINNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

This language element is not yet defined.

### Data Type

Unknown

### Syntax

Unknown

### Legal values

Unknown

### Usage

**Word Pro: BookletPrinting property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BOOKLETPRINTING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: CanCreatePreviewBitmap property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CANCREATEPREVIEWBITMAP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## Word Pro: Center property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_CENTER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a graphic object is centered horizontally and vertically within a layout object.

## Data Type

[Integer](#)

## Syntax

[objectreference].Center = centervalue

centervalue = [objectreference].Center

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Equivalent to the Placement setting, which is located in the Watermark properties panel of the InfoBox.

## **Word Pro: ChangeKeyboardToLanguage property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHANGEKEYBOARDTOLANGUAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**



## **Word Pro: ChangeTextToMatchkeyboard property**

{button ,AL(`H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CHANGETEXTTOMATCHKEYBOARD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: ClassId property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLASSID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: CreatePreviewBitmap property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEPREVIEWBITMAP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: DataFormat property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATAFORMAT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: DdeEnabled property**

{button ,AL(`H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DDEENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: DefaultDropCapStyleDescription property**

{button ,AL(`H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DEFAULTDROPCAPSTYLEDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: DefaultNewCategory property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTNEWCATEGORY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: DiagonalLines property**

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIAGONALLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**



**Word Pro: DisplayAsIcon property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DISPLAYASICON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: DriverName property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DRIVERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## Word Pro: DropCapPosition property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_DROPCAPPOSITION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Controls where a DropCap layout object will be placed in relation to the first line of text.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

dropcappositionvalue = [objectreference].DropCapPosition

[objectreference].DropCapPosition = dropcappositionvalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpDropcapAbove (2235)	Equivalent to choosing Above first line in the Drop Cap dialog box.
\$LwpDropcapBelow (2234)	Equivalent to choosing Below first line in the Drop Cap dialog box.
\$LwpDropcapBeside (2236)	Equivalent to choosing Beside paragraph in the Drop Cap dialog box.

## Usage

## **Word Pro: DropCapStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DROPCAPSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: DropCapStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DROPCAPSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: DropCaps property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DROPCAPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: ExternallyControlledUndo property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXTERNALLYCONTROLLEDUNDO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: FaceName property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FACENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: FileFormat property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILEFORMAT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: FinishedSpellChecking property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINISHEDSPELLCHECKING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: FirstDivision property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIRSTDIVISION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: FirstSpellString property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIRSTSPELLSTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: FontMatching property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTMATCHING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: FooterStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTERSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: FooterStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTERSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: ForeColorIndex property**

{button ,AL('H\_BACKGROUND\_CLASS;H\_BORDER\_CLASS;H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORECOLORINDEX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: GotoPageLoadInBackground property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOPAGELOADINBACKGROUND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: GraphicPaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHICPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: GraphicPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHICPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: HasIndex property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HASINDEX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: HasTOC property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HASTOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: HeaderStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HEADERSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## **Word Pro: HeaderStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HEADERSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: IDispatch property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IDISPATCH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: IsAmikake property**

{button ,AL('H\_AMIKAKE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISAMIKAKE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: IsChangedOtherThanLinkTo property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCHANGEDOTHERTHANLINKTO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: IsChangedToLinkTo property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCHANGEDTOLINKTO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: IsChartLink property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCHARTLINK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: IsEndnoteDivision property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISENDNOTEDIVISION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## Word Pro: IsLocked property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISLOCKED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to lock a CellLayout object from being removed.

## Data Type

[Integer](#)

## Syntax

[objectreference].IsLocked = islockedvalue

islockedvalue = [objectreference].IsLocked

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

When you enter a table cell in Word Pro, a cell layout object is created. If you leave the cell layout without modifying the layout or its content, Word Pro removes the cell layout object. Setting the IsLocked property to True prevents WordPro from removing a particular cell layout object.

The IsLocked property applies only to cell layout objects.

### **Word Pro: IsPrompting property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPROMPTING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: IsSmartEditEnabled property**

{button ,AL(`H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ISSMARTEDITENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



## Word Pro: IsStyle property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISSTYLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether a particular layout object is a style.

## Data Type

[Integer](#)

## Syntax

[objectreference].IsStyle = isstylevalue

isstylevalue = [objectreference].IsStyle

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

You can reference this property to determine whether a particular layout object is a style layout object.

This property is useful when accessing layout object collections, since style layout objects are stored in collections along with user-created layout objects. Modifying style layout objects can sometimes cause unpredictable results.

**Word Pro: IsSymbolic property**

{button ,AL('H\_FONTMETRICS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSYMBOLIC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: IsViewRubyMarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWRUBYMarks\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: IsViewStatusSpell property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWSTATUSSPELL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: LastDivision property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTDIVISION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## Word Pro: LayerName property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_LAYERNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates the name of the Layer object for a specified layout object.

## Data Type

String

## Syntax

layernamevalue = [objectreference].LayerName

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

The LayerName property provides you with the name of the Layer, or Watermark, object for a specified layout object. If a layer object does not exist in the specified layout object, its LayerName property will contain an empty string.

## Word Pro: Layer property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_LAYER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Provides access to the Layer object for a specific layout object.

## Data Type

[FrameLayout](#)

## Syntax

layervalue = [objectreference].Layer

## Legal values

Always contains an instance of the FrameLayout class.

## Usage

The Layer property provides access to a layout object's Layer, or Watermark, object.

## Word Pro: LinkFrame property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_LINKFRAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines whether or not a frame layout object will link its contents with another frame layout object.

## Data Type

String

## Syntax

linkframevalue = [objectreference].LinkFrame

[objectreference].LinkFrame = linkframevalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

Equivalent to the "Link frame contents to" setting, located on the Misc panel of the InfoBox for frame layout objects. The LinkFrame property is used only with frame layout objects.

In order to remove the link from a frame layout object, set the LinkFrame property to an empty string value. Setting a layout object's LinkFrame property to a value that does not correspond with an appropriate frame layout object also results in an empty string value being assigned to the LinkFrame property.



**Word Pro: LockResult property**

{button ,AL('H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOCKRESULT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: MergeFileType property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGEFILETYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: MergeToFile property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGETOFILE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: MetafilePict property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_METAFILEPICT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: NextSpellString property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTSPELLSTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: NonUserDocument property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NONUSERDOCUMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: NumberOfCharacters property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBEROFCHARACTERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## Word Pro: NumberOfLines property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_NUMBEROFLINES_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates the number of lines used to determine the height of a drop cap.

## Data Type

[Integer](#)

## Syntax

numberoflinesvalue = [objectreference].NumberOfLines

[objectreference].NumberOfLines = numberoflinesvalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

Equivalent to the "Height of Drop Cap" setting in the Drop Cap dialog box.



**Word Pro: NumberOfMergeConditions property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBEROFMERGECONDITIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Sets the number of conditions for a Merge.

**Data Type**

[Integer](#)

**Syntax**

numberofmergeconditionsvalue = [objectreference].NumberOfMergeConditions

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: NumLinesOfSpaceAboveLine property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMLINESOFSPACEABOVELINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: OCXDesignMode property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OCXDESIGNMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: Ole1Object property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLE1OBJECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: OLEAutomation property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLEAUTOMATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: OLEEnabled property**

{button ,AL(`H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_OLEENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

### **Word Pro: OleMinHeight property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLEMINHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: OleObjectSize property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLEOBJECTSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: Placement property**

{button ,AL(`H\_RUBYLAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_PLACEMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: Plain property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PLAIN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: ProgID property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PROGID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: ReadCompressed property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_READCOMPRESSED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: RetainNameOfImportedFile property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RETAINNAMEOFIMPORTEDFILE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: Shape property**

{button ,AL('H\_BACKGROUND\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHAPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: ShowBubbleHelp property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWBUBBLEHELP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: ShowFileNew property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWFILENEW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: SingleCellSelected property**

{button ,AL('H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUM  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SINGLECELLSELECTED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## Word Pro: Span property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_SPAN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a row layout object will span across multiple pages.

## Data Type

[Integer](#)

## Syntax

spanvalue = [objectreference].Span

[objectreference].Span = spanvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Equivalent to the "Row spans pages" setting, located in the Table Cell Size & Margins panel of the InfoBox for cell layout objects. This property is used only with cell layout objects.

**Word Pro: SpellString property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLSTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: StrikeThrough property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STRIKETHROUGH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

## Word Pro: Tile property

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_CELLGROUPLAYOUT\_CLASS;H\_CELLAYOUT\_CLASS;H\_COLUMNNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTE LAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS ;H\_NOTELAYOUT\_CLASS;H\_PAGE\_LAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_T ABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TILE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines whether or not a layout object's graphic content will repeat horizontally and vertically.

## Data Type

[Integer](#)

## Syntax

tilevalue = [objectreference].Tile

[objectreference].Tile = tilevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: TypeAboveLine property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TYPEABOVELINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: UpdateOnLoad property**

{button ,AL('H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATEONLOAD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

### **Data Type**

Unknown

### **Syntax**

Unknown

### **Legal values**

Unknown

### **Usage**

**Word Pro: UserDefinedFilter property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERDEFINEDFILTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**



**Word Pro: WaterMarkName property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WATERMARKNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

**Word Pro: WidthInLongtwips property**

{button ,AL('H\_BORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDTHINLONGTWIPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only/Read-write)

This language element is not yet defined.

**Data Type**

Unknown

**Syntax**

Unknown

**Legal values**

Unknown

**Usage**

## **Word Pro: Accelerators class**

A short-cut key assignment for any Word Pro script or Ami Pro macro. With the methods in this class, you can add or remove any accelerator key assignment.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the Accelerators Property

### **Usage**

Note that any accelerators you create and assign will expire when you end the session of Word Pro in which you created those accelerators. To keep an accelerator in memory between Word Pro sessions, you must set the accelerator's IsTemporaryUse parameter to "0."

## **Word Pro: Alignment class**

The alignment settings for an object.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [Alignment](#) Property

[Formula](#) in the [Alignment](#) Property

[Graphic](#) in the [Alignment](#) Property

[ParagraphStyle](#) in the [Alignment](#) Property

[Text](#) in the [Alignment](#) Property

[TextMarker](#) in the [Alignment](#) Property

### **Usage**

Alignment objects are stored in the Alignment property of other objects. Use the syntax shown in the Alignment property description to make use of this class' properties and methods.

## **Word Pro: Amikake class**

The text background object for the Asian language versions of Word Pro. If you are using an English language version of Word Pro, this class is not available.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

CharacterStyle in the Amikake Property

ClickHere in the Amikake Property

ParagraphStyle in the Amikake Property

Text in the Amikake Property

TextMarker in the Amikake Property

### **Usage**

## **Word Pro: ApplicationWindow class**

The Word Pro window which acts as the container for all of your document windows. Also known as the application workspace, this is the window which remains after you close all of your documents and leave Word Pro running.

### **Base Classes**

BaseObjectWindow

### **Derived Classes**

None

### **Contained by**

Application in the ApplicationWindow Property

WPApplication in the ApplicationWindow Property

### **Usage**

This class is shared by all Lotus applications. Each Lotus application creates a single ApplicationWindow object when you launch the application. Word Pro's ApplicationWindow object is contained by the WPApplication class in the ApplicationWindow property. WPApplication inherits the ApplicationWindow property from the Application class.

Only one ApplicationWindow object is instantiated at any given time. The ApplicationWindow object allows you to access all of the Word Pro features which are available independently of the documents you create. For example, your preference settings, accelerator keys, SmartIcons bars, menus, the status bar, and other features are available, regardless of which document you have open. Many of these global features are accessed through the ApplicationWindow property in WPApplication.

## **Word Pro: Application class**

An abstract class that acts as a template for the WPAApplication class. To access and manipulate the Word Pro application, use the Word Pro subclass named WPAApplication.

### **Base Classes**

BaseObject

### **Derived Classes**

WPAApplication

### **Contained by**

### **Usage**

Each Lotus application is represented in LotusScript by its own subclass of the Application class. By sharing a common parent class, each Lotus application inherits a common set of properties, methods, and events. This makes it possible for you to interact with each Lotus application in much the same way. You should also note that, as in all LotusScript classes, the Application class is itself a subclass of the BaseObject class.

The Application class lays the groundwork for a number of abilities and attributes which are common to all Lotus applications. These are inherited by each application's subclass and enhanced for that application's unique needs. For example, Application maintains application-wide settings and user information. Application also manages and creates documents. There is a single window associated with each application. WPAApplication inherits each of these traits and enhances them to meet the needs of the Word Pro application.

There is only one application object per running application instance. You cannot instantiate an object from the Application class.

## **Word Pro: AppViewPrefs class**

The view preferences for a session of Word Pro.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

WPApplication in the AppViewPrefs Property

### **Usage**

This class defines the color of margins, window panes, selection borders, spelling errors, and the currently selected spelling error. To access the object instantiated from this class, use the AppViewPrefs property on WPApplication.



## **Word Pro: Attributes class**

The attributes of an object.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[CharacterStyle](#) in the [TextAttributes](#) Property

[ClickHere](#) in the [Attributes](#) Property

[Editor](#) in the [TextAttributes](#) Property

[FindAndReplace](#) in the [ReplaceAttributes](#) Property

[FindAndReplace](#) in the [SearchAttributes](#) Property

[FormatPreferences](#) in the [Attributes](#) Property

[ParagraphStyle](#) in the [TextAttributes](#) Property

[RevisionDisplay](#) in the [TextAttributes](#) Property

[Text](#) in the [Attributes](#) Property

[TextMarker](#) in the [Attributes](#) Property

### **Usage**

**Word Pro: AutoRunMacro class**

Contains the names of macros which will run automatically each time a document is created, opened, or closed.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

TextDocument in the AutoRunMacro Property

**Usage**

## **Word Pro: Background class**

The background of an object. Three properties correspond to the three elements of an object's background: the pattern, the pattern color, and the color of the null space behind the pattern.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Amikake in the Background Property

Layout in the Background Property

TableFill in the Background Property

### **Usage**

Setting the properties of a Background object is the same as setting the InfoBox options labeled Pattern, Pattern color, and Background color for a Layout or TableFill object.

Use the Pattern property to choose a pattern for the Background object. Use the Color property to choose a color for that pattern. Use the BackColor property to choose a color for the null space behind the pattern.

**Word Pro: BagCollection class**

A collection of Bag objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Bags Property

**Usage**

**Word Pro: Bag class**

Stores custom data for a division.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by****Usage**

## **Word Pro: BaseCollection class**

A virtual class which provides the basic members for all collection classes.

### **Base Classes**

BaseObject

### **Derived Classes**

[BagCollection](#)

[BookmarkCollection](#)

[CellCollection](#)

[CellLayoutCollection](#)

[CharacterStyleCollection](#)

[ClickHereCollection](#)

[ConnectedLayoutCollection](#)

[ContentCollection](#)

[DdeLinkCollection](#)

[DivisionCollection](#)

[DocInfoFieldCollection](#)

[Documents](#)

[DocWindowCollection](#)

[EditorCollection](#)

[EndnoteLayoutCollection](#)

[FooterLayoutCollection](#)

[FootnoteCollection](#)

[FootnoteLayoutCollection](#)

[FrameLayoutCollection](#)

[GlossaryCollection](#)

[GraphicCollection](#)

[GraphicOleObjectCollection](#)

[GroupLayoutCollection](#)

[HeaderLayoutCollection](#)

[IconBarCollection](#)

[LayoutCollection](#)

[LongCollection](#)

[MarkerCollection](#)

[MenuItemCollection](#)

[NoteLayoutCollection](#)

[OleObjectCollection](#)

[OutlineSeqCollection](#)

[OutlineSeqItemCollection](#)

[PageLayoutCollection](#)

[ParagraphStyleCollection](#)

[ParallelColsCollection](#)

[PowerFieldCollection](#)

[RowLayoutCollection](#)

[RubyLayoutCollection](#)

[ScriptDataSetCollection](#)

[SectionCollection](#)  
[SilverBulletCollection](#)  
[StatusBarButtonCollection](#)  
[StringCollection](#)  
[SuperTableCollection](#)  
[SuperTableLayoutCollection](#)  
[TableCollection](#)  
[TableHeadingCollection](#)  
[TableHeadingLayoutCollection](#)  
[TableLayoutCollection](#)  
[TableMarkerCollection](#)  
[TableOnlyCollection](#)  
[TextCollection](#)  
[TextMarkerCollection](#)  
[TextStyleCollection](#)  
[UnitCollection](#)  
[VersionCollection](#)  
[WPDataSetCollection](#)

**Contained by**

**Usage**

## **Word Pro: BaseContainer class**

An abstract class which defines properties and methods that are common to all Word Pro container objects. An explanation of container objects is provided below under Usage. This information applies to all container objects in Word Pro, but each container object may exhibit minor differences. These differences are noted in the documentation of each specific container class.

### **Base Classes**

BaseObject

### **Derived Classes**

[CellContainer](#)

[FrameContainer](#)

[NoteContainer](#)

[PageContainer](#)

[ParallelColsContainer](#)

[RowContainer](#)

[RubyContainer](#)

[SubPageContainer](#)

[SuperPageContainer](#)

[SuperTableContainer](#)

[TableContainer](#)

[TableOnlyCont](#)

### **Contained by**

[WPApplication](#) in the [Container](#) Property

### **Usage**

Word Pro creates container objects as a means of giving you quick and easy access to a group of related objects. However, that access is provided only when those related objects have the focus. For example, a table cell is comprised of a group of related objects, including a CellLayout object, a Background object, a Borderlines object, and all the attributes of those objects. When you move the insertion point into that cell, we say that cell has the focus. When a cell gets the focus, Word Pro creates a container object from the CellContainer class to hold all the objects related to that cell.

Container objects are temporary and exist only as long as a group of related objects has the focus. If you move the insertion point to another object or group of objects, Word Pro destroys the container object, leaving the group of related objects intact. Only a handful of objects have related objects that can be pulled together in a container. They include pages, tables, parallel columns, super tables, cells, and frames. Each of these objects has related objects that get pulled into a container when you give that object the focus (for example, when you place the insertion point in that object). A container object is always stored in the same property on the WPApplication object. You can never have more than one kind of container object at any given time.

The container object properties on the WPApplication object include:

Cell

Frame

Page

SuperTableContainer

TableContainer

TableOnlyContainer

You may also notice the Container property, which always contains the topmost container object in the focus. Container uses the abstract class BaseContainer as its data type. This allows the Container property to store any kind of container object.

Container properties remain empty unless the focus includes a cell, a frame, a page, a table, or some combination of these. As you move the focus around in a document and different groups of related objects come into the focus, Word Pro creates temporary container objects and stores them in these container properties to give you easy access to



each group of related objects. While Word Pro never creates more than one of each type of container object at any given time, it is not unusual to have one of each kind of container object stored in the WPAApplication properties.

### **An Example of Container Objects**

Here's an example of how Word Pro would manage container objects in a document with a table on the first page, a table and a frame on the second page, and a set of parallel columns on the third page:

When you place your insertion point on the first page, Word Pro creates a PageContainer object and stores it in the Page property. When you move your insertion point to a table cell, Word Pro creates CellContainer, TableOnlyCont, and SuperTableContainer objects, and places them in their respective container properties. Word Pro leaves the PageContainer object for page one intact because you never moved your insertion point (and thus the focus) off page one.

When you move the insertion point to page two, Word Pro destroys all the container objects from the first page and creates a new PageContainer object for the second page. When you move the insertion point to a cell in the table on the second page, Word Pro creates container objects for that table, its super table, and the cell in which you placed the insertion point. When you move the insertion point to the frame on the second page, Word Pro destroys the table, super table, and cell container objects, and creates a FrameContainer object. Word Pro leaves the PageContainer object for page two intact because your insertion point never left page two.

When you move the insertion point to page three, Word Pro destroys all the container objects from the second page. Page three has a set of parallel columns. Word Pro sees parallel columns as a special kind of table. In fact, the Table class and ParallelColumns class are derived from the same parent class (BaseTable), and the container classes (TableOnlyCont and ParallelColsContainer) for these objects are derived from the same parent class (TableContainer). You may notice that there is a property for the TableOnlyCont object, but no property set aside for the ParallelColsContainer object. Word Pro provides a place for the ParallelColsContainer object in the TableContainer property by specifying the data type of the TableContainer property as TableContainer (the parent class for ParallelColsContainer). Using the parent class as the data type allows Word Pro to store either a ParallelColsContainer or a TableOnlyCont object in the TableContainer property. Thus, when you move the insertion point onto page three, Word Pro creates a PageContainer object for the Page property and a ParallelColsContainer object for the TableContainer property.

### **Using Containers To Access The Appropriate Layout**

In most cases, you will use container objects as a means of selecting a single Layout object when the focus encompasses several layout objects. In the example above, if your focus was on a table cell and you wanted to access the PageLayout object, you couldn't use the statement:

```
.Layout.layoutpropertyname
```

This would return the layout of the cell. Instead, you need a way of specifying which layout object you want. The container objects make this easy because you can always use a statement such as:

```
Page.Layout
```

or

```
TableOnlyContainer.Layout
```

### **Variables and Container Objects**

Remember that a container object is temporary and changes with the focus. So if the insertion point was in cell (1,1) of a table, and you assign the container object to a variable (myCellContainer) like this:

```
set myCellContainer = .Cell
```

You can call the variable and get cell (1,1), as long as the focus remains on cell (1,1). However, if you move the focus to cell (2,2) and try to use that same variable as in the statement below:

```
myCellContainer.Layout.Content.InsertText "Hello"
```

Word Pro will insert the text "Hello" into cell (2,2) because the variable myCellContainer contains the CellContainer object and that object changed from cell (1,1) to cell (2,2) when you moved the focus. Container objects always refer to the current context and the variables that contain container objects also change with the focus.

Finally, if you move the focus out of the table entirely so that the insertion point is not within a cell at all, then this statement will result in an error:

```
myCellContainer.Layout.Content.InsertText "Hello"
```

This is because the myCellContainer variable must contain a container object for the statement to work, but the object is destroyed when you move the focus out of the table.

## **Word Pro: BaseObject class**

The BaseObject class is shared among all Lotus applications and is a true virtual class: no instances of the BaseObject class are ever created. BaseObject exists to provide a basic set of properties to all Word Pro classes. It is important to remember, that while the properties in the BaseObject class are inherited by every Word Pro class, not every Word Pro class actually implements each property. Check the descriptions for the BaseObject properties to see which are implemented by Word Pro classes.

### **Base Classes**

[BaseObject](#)

### **Derived Classes**

[Accelerators](#)

[Alignment](#)

[Amikake](#)

[Application](#)

[ApplicationWindow](#)

[AppViewPrefs](#)

[Attributes](#)

[AutoRunMacro](#)

[Background](#)

[BagCollection](#)

[Bag](#)

[BaseCollection](#)

[BaseContainer](#)

[BaseObject](#)

[BaseTable](#)

[BookmarkCollection](#)

[Bookmark](#)

[BookmarkManager](#)

[Border](#)

[BorderLines](#)

[Breaks](#)

[Bullet](#)

[CellCollection](#)

[CellContainer](#)

[CellEngine](#)

[CellGroupLayout](#)

[CellLayoutCollection](#)

[CellLayout](#)

[CharacterBorder](#)

[CharacterSet](#)

[CharacterStyleCollection](#)

[CharacterStyle](#)

[ClickHereCollection](#)

[ClickHere](#)

[Color](#)

[Column](#)

[ConnectedLayoutCollection](#)

[ConnectedLayout](#)  
[ContentCollection](#)  
[Content](#)  
[ContextMenuOptions](#)  
[DdeLinkCollection](#)  
[DdeLink](#)  
[DdeLinkManager](#)  
[DivisionCollection](#)  
[Division](#)  
[DivisionInfo](#)  
[DivisionOptions](#)  
[DocControl](#)  
[DocInfoFieldCollection](#)  
[DocInfoField](#)  
[DocInfoFieldManager](#)  
[DocInfo](#)  
[Document](#)  
[Documents](#)  
[DocWindowCollection](#)  
[DocWindow](#)  
[EditorCollection](#)  
[Editor](#)  
[EditorManager](#)  
[EndnoteDivisionGroupNum](#)  
[EndnoteDivisionNum](#)  
[EndnoteDocNum](#)  
[EndnoteLayoutCollection](#)  
[EndnoteLayout](#)  
[FileProtection](#)  
[Filter](#)  
[FilterHelper](#)  
[FindAndReplace](#)  
[Font](#)  
[FontMetrics](#)  
[FooterLayoutCollection](#)  
[FooterLayout](#)  
[FootnoteCollection](#)  
[FootnoteContSep](#)  
[Footnote](#)  
[FootnoteLayoutCollection](#)  
[FootnoteLayout](#)  
[FootnoteNumbering](#)  
[FootnoteNumOpt](#)  
[FootnoteOptions](#)  
[FootnoteSeparator](#)  
[FootnoteSepOpt](#)  
[FootnoteTable](#)

[FormatCheckPref](#)  
[FormatPreferences](#)  
[Formula](#)  
[Foundry](#)  
[FrameContainer](#)  
[FrameGroupLayout](#)  
[FrameLayoutCollection](#)  
[FrameLayout](#)  
[GlossaryCollection](#)  
[Glossary](#)  
[Grammar](#)  
[GraphicCollection](#)  
[Graphic](#)  
[GraphicOleObjectCollection](#)  
[GraphicOleObject](#)  
[GroupLayoutCollection](#)  
[GroupLayout](#)  
[Gutter](#)  
[HeaderLayoutCollection](#)  
[HeaderLayout](#)  
[HyphenationOptions](#)  
[IconBarCollection](#)  
[IconBar](#)  
[IconBarManager](#)  
[Indent](#)  
[Index](#)  
[IndexSection](#)  
[Join](#)  
[Kinsoku](#)  
[Language](#)  
[LayoutCollection](#)  
[Layout](#)  
[LayoutOverride](#)  
[LineNumberOptions](#)  
[LongCollection](#)  
[LWPTimer](#)  
[Macro](#)  
[MailRouting](#)  
[MarkerCollection](#)  
[Marker](#)  
[MenuItemCollection](#)  
[MenuItem](#)  
[MergeOptions](#)  
[NoteContainer](#)  
[NoteLayoutCollection](#)  
[NoteLayout](#)  
[Numbering](#)

[NumericFormat](#)  
[NumericFormatSubset](#)  
[OleObjectCollection](#)  
[OleObject](#)  
[Options](#)  
[OutlineSeqCollection](#)  
[OutlineSeqItemCollection](#)  
[OutlineStyleSequence](#)  
[OutSeqItem](#)  
[PageContainer](#)  
[PageLayoutCollection](#)  
[PageLayout](#)  
[ParagraphBorder](#)  
[ParagraphStyleCollection](#)  
[ParagraphStyle](#)  
[ParallelColsCollection](#)  
[ParallelColsContainer](#)  
[ParallelColumns](#)  
[PowerFieldCollection](#)  
[PowerField](#)  
[Preferences](#)  
[Presentation](#)  
[PrintManager](#)  
[PrintSettings](#)  
[RelativeIndent](#)  
[ReviewVersions](#)  
[RevisionDisplay](#)  
[Revision](#)  
[RowContainer](#)  
[RowLayoutCollection](#)  
[RowLayout](#)  
[RubyContainer](#)  
[RubyLayoutCollection](#)  
[RubyLayout](#)  
[RubyMarker](#)  
[Ruler](#)  
[ScriptDataSetCollection](#)  
[ScriptDataSet](#)  
[Script](#)  
[SectionCollection](#)  
[Section](#)  
[SectionTabs](#)  
[SetTabsDialog](#)  
[Shadow](#)  
[SilverBulletCollection](#)  
[SilverBullet](#)  
[SmartCorrect](#)

[SortKey](#)  
[SortOptions](#)  
[Spacing](#)  
[StatusBarButtonCollection](#)  
[StatusBarButton](#)  
[StatusBar](#)  
[StringCollection](#)  
[SubPageContainer](#)  
[SuperPageContainer](#)  
[SuperTableCollection](#)  
[SuperTableContainer](#)  
[SuperTableGroupLayout](#)  
[SuperTable](#)  
[SuperTableLayoutCollection](#)  
[SuperTableLayout](#)  
[TableCollection](#)  
[TableContainer](#)  
[TableFill](#)  
[Table](#)  
[TableHeadingCollection](#)  
[TableHeading](#)  
[TableHeadingLayoutCollection](#)  
[TableHeadingLayout](#)  
[TableLayoutCollection](#)  
[TableLayout](#)  
[TableLine](#)  
[TableMarkerCollection](#)  
[TableMarker](#)  
[TableOnlyCollection](#)  
[TableOnlyCont](#)  
[TabRack](#)  
[TextCollection](#)  
[TextDocument](#)  
[Text](#)  
[TextMarkerCollection](#)  
[TextMarker](#)  
[TextStyleCollection](#)  
[TOCSuperTableLayout](#)  
[UnitCollection](#)  
[UserInterfacePrefs](#)  
[UseWhen](#)  
[VersionCollection](#)  
[Version](#)  
[VersionManager](#)  
[Window](#)  
[WinViewPrefs](#)  
[WPAApplication](#)

[WPDataSetCollection](#)

[WPDataSet](#)

### **Contained by**

[BaseObject](#) in the [Parent](#) Property

### **Usage**

BaseObject is one of the shared Lotus Object Interface (LOI) classes. The BaseObject class exists solely as a means of providing a common set of class members to all other classes in Word Pro and other Lotus applications. Word Pro never instantiates an object from the BaseObject class alone, and you cannot instantiate an object from the BaseObject class. The properties defined in the BaseObject class are inherited, directly or indirectly, by every class in the Word Pro object model. However, not every class implements these inherited properties. For example, every Word Pro class inherits the Description property, but only the DocInfo class implements the property for you to use in your scripts.

Another feature of deriving Word Pro classes from the BaseObject class is the ability to store any Word Pro object in a variable of type BaseObject. When you store an object in this manner, you can access only those class members which the object inherited from the BaseObject class and implemented for your use. For Word Pro, the BaseObject class defines only six class members, all of them properties. With this basic set of properties, you can get basic information about any object in the Word Pro object model, including the object's base application, parent, version, and whether or not the object is available for your use in a specific context. For more information on which classes implement the BaseObject properties, see the individual property descriptions.

## **Word Pro: BaseTable class**

An abstract class acting as the parent class for several types of tables, glossaries, and parallel columns. This class is used as the data type for the BaseTable property so that property can contain any object created from one of BaseTable's derived classes.

### **Base Classes**

BaseObject\Content

### **Derived Classes**

FootnoteTable

Glossary

ParallelColumns

Table

TableHeading

### **Contained by**

WPAApplication in the BaseTable Property

### **Usage**

This abstract class describes the behavior that is common to FootnoteTables, Tables, TableHeadings, Glossaries, and ParallelColumns, all of which are derived from BaseTable. Note that the BaseTable class is derived from the Content class. This allows the objects created from classes derived from BaseTable to provide the content for other objects. These objects typically contain RowLayouts, ColumnLayouts, and CellLayouts. Each CellLayout has its own content which may be any of the following Content types: Text, Graphic, OleObject.



**Word Pro: BookmarkCollection class**

A collection of bookmark objects in the BookmarkManager class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

BookmarkManager in the Bookmarks Property

BookmarkManager in the BookmarksByMarkerName Property

**Usage**

Use this collection to access any of the bookmark objects in the BookmarkManager class.

Word Pro automatically creates and maintains one BookmarkCollection object for each division of a document. The BookmarkCollection object and its contents are stored with the document.

## **Word Pro: BookmarkManager class**

A tool for managing bookmarks in a document. Keeps and manages the list of bookmarks in the document. You must go through the BookmarkManager before using the Bookmark property.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the BookmarkManager Property

TextDocument in the BookmarkManager Property

### **Usage**

Used in conjunction with Bookmark and BookmarkCollection objects. You can use the BookmarkManager to select, find, add, or remove a bookmark. Word Pro keeps a list of each document's bookmarks in the BookmarkCollection object.

## **Word Pro: Bookmark class**

A bookmark in a Word Pro document. Word Pro instantiates (creates an instance of) a Bookmark object each time you create a bookmark in a document. Once created, the bookmark name will display in the Bookmarks dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**

You can move easily through a set of bookmarks in a document to find the one you want and use the properties of the Bookmark class to set and retrieve a specific bookmark's attributes. Word Pro keeps a list of each document's bookmarks in the BookmarkCollection object.

## Word Pro: BorderLines class

The lines which comprise the border of an object.

### Base Classes

BaseObject

### Derived Classes

Gutter

### Contained by

CharacterBorder in the BorderLines Property

FootnoteSepOpt in the BorderLines Property

Layout in the BorderLines Property

ParagraphBorder in the BorderLines Property

TableLine in the BorderLines Property

TableLine in the OutlineBorderLines Property

### Usage

Most of the properties in BorderLines contain instances of the Border class. You can use the contents of these properties to effect changes to the lines around an object. The complete syntax will depend on the containing object but will always follow this standard:

```
.objectname.BorderLines.AllBorders.property/method
```

The same syntax applies to BottomBorder, LeftBorder, RightBorder, and TopBorder. Each of these properties has a data type of Border, and therefore contains a Border object which has its own set of properties that you must use to achieve your desired results.

## Word Pro: Border class

The attributes of a border including the border's width, color, and pattern.

### Base Classes

BaseObject

### Derived Classes

None

### Contained by

BorderLines in the AllBorders Property

BorderLines in the BottomBorder Property

BorderLines in the RightBorder Property

BorderLines in the TopBorder Property

BorderLines in the LeftBorder Property

### Usage

Nearly every object in Word Pro has a border of some sort. You can use this object in any object which contains an instance of the BorderLines class. The complete syntax will depend on the containing object but will always follow this standard:

```
.objectname.BorderLines.AllBorders.Color.Red=0
```

```
.objectname.BorderLines.AllBorders.Color.Green=0
```

```
.objectname.BorderLines.AllBorders.Color.Blue=255
```

```
.objectname.BorderLines.AllBorders.Pattern=1
```

```
.objectname.BorderLines.AllBorders.Width=200
```

The same syntax applies to BottomBorder, LeftBorder, RightBorder, and TopBorder. Each of these properties has a data type of Border and contains a Border object which comprises the top, bottom, left, right, and all border properties of the larger BorderLines object.

## **Word Pro: Breaks class**

Break options as seen in the Advanced panel of the Text Properties dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [Breaks](#) Property

[ParagraphStyle](#) in the [Breaks](#) Property

[Text](#) in the [Breaks](#) Property

[TextMarker](#) in the [Breaks](#) Property

### **Usage**

Use the properties of this class to set the Break options for an object. Use the RevertBreaksToStyle method to return all the break options to the options selected in the object's style.

## **Word Pro: Bullet class**

Bullet options as seen in the Bullet panel of the Text Properties dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [Bullet](#) Property

[ParagraphStyle](#) in the [Bullet](#) Property

[Text](#) in the [Bullet](#) Property

[TextMarker](#) in the [Bullet](#) Property

### **Usage**

Use the properties of this class to set the Bullet options for an object. Use the RevertBulletToStyle method to return all the bullet options to the options selected in the object's style.

**Word Pro: CellCollection class**

A collection of cell objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the CellEngines Property

**Usage**

Use this collection to access any of the cell objects in the foundry of a specific division.



## **Word Pro: CellContainer class**

The container object for table cells. This object only exists for one table cell at a time and only when there is a table cell within the focus. When a CellContainer object is present, it is stored in the Cell property on the WPAApplication object.

### **Base Classes**

BaseObject\BaseContainer

### **Derived Classes**

None

### **Contained by**

WPAApplication in the Cell Property

### **Usage**

The primary use for a CellContainer object is to provide quick and easy access to the CellLayout object for the currently active cell. A CellContainer object always represents the cell that currently has the focus. Thus, if you assign a CellContainer object to a variable, you can use that variable to access the currently active cell. However, you must remember that the cell referenced by the variable will change as the focus moves from one cell to another. This is because the variable references the CellContainer object, and the CellContainer object always represents the cell that has the focus. If there is no cell within the focus, there is no CellContainer object. Thus, a variable that stores a CellContainer object will have a null value whenever the focus does not contain a cell. There is never more than one CellContainer object at any given time.

For more information about container objects, see BaseContainer.

## **Word Pro: CellEngine class**

This class allows you to access formulas within a table.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Table in the CellEngine Property

### **Usage**

## **Word Pro: CellGroupLayout class**

The cell group layout for a cell group.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

WPApplication in the CurrentCell property

### **Usage**

## **Word Pro: CellLayoutCollection class**

A collection of cell layout objects in the foundry of a specific division.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

Foundry in the CellLayouts Property

Foundry in the CellLayoutStyles Property

### **Usage**

Use this collection to access any of the cell layout objects in the foundry of a specific division.

## **Word Pro: CellLayout class**

The cell layout for a CellContainer object. This class inherits most of its members from the Layout class.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

ConnectedLayout

### **Contained by**

BaseTable in the CurrentCell Property

CellContainer in the CellLayout Property

WPApplication in the CurrentCell Property

### **Usage**

## **Word Pro: CharacterBorder class**

The border around a character in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[CharacterStyle](#) in the [CharacterBorder](#) Property

[ClickHere](#) in the [CharacterBorder](#) Property

[ParagraphStyle](#) in the [CharacterBorder](#) Property

[Text](#) in the [CharacterBorder](#) Property

[TextMarker](#) in the [CharacterBorder](#) Property

### **Usage**

## **Word Pro: CharacterSet class**

A set of characters used in Find and Replace.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[FindAndReplace](#) in the [CharacterSet](#) Property

[Preferences](#) in the [CharacterSet](#) Property

### **Usage**

**Word Pro: CharacterStyleCollection class**

A collection of character style objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the CharacterStyles Property

**Usage**

Use this collection to access any of the character style objects in the foundry of a specific division.



## **Word Pro: CharacterStyle class**

Contains the style used to create a character in a division.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [CharacterStyle](#) Property

[Text](#) in the [CharacterStyle](#) Property

[TextMarker](#) in the [CharacterStyle](#) Property

### **Usage**

**Word Pro: ClickHereCollection class**

A collection of ClickHere objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the ClickHeres Property

**Usage**

Use this collection to access any of the ClickHere objects in the foundry of a specific division application.

**Word Pro: ClickHere class**

A ClickHere block in a document.

**Base Classes**

BaseObject\Marker

**Derived Classes**

None

**Contained by****Usage**

ClickHere objects have many of the same properties and methods as Text objects. Note, however, that the events available on a ClickHere object are limited to EnterClickHere and ExitClickHere. Use these events to trigger scripts written for a ClickHere object.

## Word Pro: Color class

Defines the specific color that is applied to a given object. Equivalent to accessing the color palette for text and lines in the InfoBox.

### Base Classes

BaseObject

### Derived Classes

None

### Contained by

[AppViewPrefs](#) in the [MarginColor](#) Property

[AppViewPrefs](#) in the [PaneColor](#) Property

[AppViewPrefs](#) in the [SelectionBorderColor1](#) Property

[AppViewPrefs](#) in the [SelectionBorderColor2](#) Property

[AppViewPrefs](#) in the [SelectionBorderColor3](#) Property

[AppViewPrefs](#) in the [SpellColor](#) Property

[AppViewPrefs](#) in the [SpellFocusedColor](#) Property

[Background](#) in the [BackColor](#) Property

[Background](#) in the [Color](#) Property

[Border](#) in the [Color](#) Property

[DivisionInfo](#) in the [Color](#) Property

[Editor](#) in the [HiLiteColor](#) Property

[Font](#) in the [FontColor](#) Property

[Font](#) in the [BackColor](#) Property

[NoteLayout](#) in the [Color](#) Property

[NumericFormatSubset](#) in the [Color](#) Property

[Preferences](#) in the [HiLiteColor](#) Property

[Script](#) in the [ErrorColor](#) Property

[Script](#) in the [IdentifierColor](#) Property

[Script](#) in the [DirectiveColor](#) Property

[Script](#) in the [KeywordColor](#) Property

[Script](#) in the [CommentColor](#) Property

[Section](#) in the [Color](#) Property

[Shadow](#) in the [Color](#) Property

[UserInterfacePrefs](#) in the [NoteColor](#) Property

### Usage

You can set a color for an object that has a color property. For example, objects such as text, shadow, frame, and line have color contexts associated with them. You can select a color context associated with an object and use that context to change the object's color. You can set a color for an object in three ways:

- Set the RGB value to pass the color triplet red, green, and blue to an object.
- Set the RGB value independently. Each one of the RGB values is a property of Color. Therefore, you can set the red, green or blue property.
- Set the Override property to use predefined Word Pro colors (red, green, blue, black, white, light gray, dark gray, and transparent). When you use the Override property value, Word Pro does not recognize previously set RGB values.

## **Word Pro: ColumnGroupLayout class**

The layout for a column group in a table object.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: Column class**

Newspaper columns in a page, frame, header, or footer container object.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

Layout

**Usage**

You use this property to change the width of a newspaper column or the distance between newspaper columns. Before you can change a newspaper column, you must first select the layout of the column and then specify the change you want to make to it.

**Word Pro: ConnectedLayoutCollection class**

A collection of connected layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the ConnectedLayouts Property

**Usage**

Use this collection to access any of the connected layout objects in the foundry of a specific division.

**Word Pro: ConnectedLayout class**

The connected layout for a connected cell object.

**Base Classes**

BaseObject\Layout\CellLayout

**Derived Classes**

None

**Contained by****Usage**

The connecting of two or more cells to create one larger cell.



**Word Pro: ContentCollection class**

A collection of content objects in the foundry of a specific division. Content can include FootnoteTable objects, Formula objects, Graphic objects, GraphicOleObjects, OleObjects, ParallelColumns objects, SuperTable objects, BaseTable objects, TableHeading objects, Table objects, and Text objects.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Contents Property

**Usage**

Use this collection to access any of the content objects in the foundry of a specific division.

## Word Pro: Content class

Content is an abstract class which provides the basic functionality that is common to all content-related objects. Each content object represents the contents of a particular type of object. For example, a Formula object represents the content of a Cell object; a Text object might represent the contents of a page or the prompt in a ClickHere block; a Graphic object might represent the contents of a frame.

In essence, while each content object is tailored to represent the contents of a particular type of object, all content objects share a few common traits. The Content class defines those common traits and each class derived from the Content class inherits those traits as properties.

### Base Classes

[BaseObject](#)

### Derived Classes

[BaseTable](#)

[FootnoteTable](#)

[Formula](#)

[Glossary](#)

[Graphic](#)

[GraphicOleObject](#)

[OleObject](#)

[ParallelColumns](#)

[SuperTable](#)

[Table](#)

[TableHeading](#)

[Text](#)

### Contained by

[Footnote](#) in the [Content](#) Property

[WPApplication](#) in the [Content](#) Property

### Usage

While no object is ever instantiated from this class, two properties use this class as their data type. By using the abstract Content class as the data type for a property, Word Pro can store any content object in that property. Note that WPApplication provides a current context property called Content, which uses the abstract class as its data type. This allows Word Pro to give you access to the content object for whatever object has the focus. By using the same technique in the Footnote class, Word Pro ensures that you can place any type of content object you like in your footnotes.

## **Word Pro: ContextMenuOptions class**

Controls the display of context sensitive menus in a cell, frame, graphic, parallel columns or text.

### **Base Classes**

None

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the ContextMenuOptions Property

### **Usage**

**Word Pro: DdeLinkCollection class**

A collection of DdeLink objects in the DdeLinkManager class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

DdeLinkManager in the DdeLinks Property

DdeLinkManager in the DdeLinksFromMarker Property

**Usage**

Use this collection to access any of the DdeLink objects in the DdeLinkManager class.

## **Word Pro: DdeLinkManager class**

A tool that creates and maintains DDE links in a document. DDE links can be created in script via the DDELinkManager. There are also DDE functions (outside the WordPro object model) for maintaining DDE links. The DDELinkManager maintains a collection of DDE link objects.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the DdeLinkManager Property

TextDocument in the DdeLinkManager Property

### **Usage**

Used in conjunction with DdeLink and DdeLinkCollection objects. DdeLinkManager allows you to perform a variety of tasks such as finding, adding, and removing Dde links between parts or all of a document. You can use the Paste Special dialog box to paste data that uses different formats. DDE link is used only if OLE fails.

## **Word Pro: DdeLink class**

Represents a DDE link in a document. DDE links can be created in script via the DDELinkManager. There are also DDE functions (outside the WordPro object model) for maintaining DDE links. The DDELinkManager maintains a collection of DDE link objects.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**

Places a DDE marker around a section of text or a range in a document in order to link it to another application. Sets up the appropriate name, format, range, and other properties.

You can link objects to a document in two ways: by DDE link or OLE link. DDE link is used only if OLE fails. You can determine if a link is an OLE link or DDE link by double-clicking on it. If the linked object remains in the background, it is a DDE link; if the linked object comes to the foreground, it is an OLE link.

You can add and remove links between parts or all of a document and use the Paste Special dialog box to paste data in different formats.

## **Word Pro: DivisionCollection class**

A collection of division objects in a document or division.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

Division in the Divisions Property

TextDocument in the Divisions Property

WPApplication in the Divisions Property

### **Usage**

Use this collection to access any of the division objects in the TextDocument or Division class.

## **Word Pro: DivisionInfo class**

Information about a division in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

BaseContainer in the DivisionInfo Property

Division in the DivisionInfo Property

Marker in the DivisionInfo Property

TextDocument in the DivisionInfo Property

### **Usage**



## **Word Pro: DivisionOptions class**

Division options displayed in the Division Properties dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the DivisionOptions Property

TextDocument in the DivisionOptions Property

### **Usage**

**Word Pro: Division class**

A division in a document. A division can contain text, frames, text marked as sections, other divisions with different properties from each other, external files linked to a document, or OLE objects.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

WPApplication in the Division Property

**Usage**

## **Word Pro: DocControl class**

The DocControl class allows you to access a document, assign editing rights, enable password protection, select or change colors that show editor markups, make insertions and deletions, and enable document protection in a division.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the DocControl Property

TextDocument in the DocControl Property

### **Usage**

DocControl is the same as TeamSecurity in the Word Pro user interface. Therefore, setting properties for DocControl is the same as choosing File - TeamSecurity and doing one of the following: opening a specific file, assigning editing rights, verifying editing rights, creating a password, disabling version review, disabling Notes/FX fields, editing ClickHere Blocks, initiating startup scripts, revealing hidden text, editing protected text, protecting frames and table cells, or displaying all division tabs.

**Word Pro: DocInfoFieldCollection class**

A collection of DocInfoField objects in the DocInfoFieldManager class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

DocInfoFieldManager in the Fields Property

**Usage**

Use this collection to access any of the DocInfoField objects in the DocInfoFieldManager class.

## **Word Pro: DocInfoFieldManager class**

A tool for managing DocInfo fields in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

DocInfo in the FieldManager Property

### **Usage**

Use the DocInfoFieldManager class to access, add, or delete DocInfo fields in a document.

**Word Pro: DocInfoField class**

Represents the information in a specific DocInfo field.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by****Usage**

You can use this class to access information about a DocInfo field, such as its name or its contents.

## **Word Pro: DocInfo class**

The class that holds all the statistics associated with a document, such as document/version editing information or field name descriptions.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the DocInfo Property

TextDocument in the DocInfo Property

### **Usage**

The DocInfo class is the container for all DocInfo fields. As a result, you can obtain any general information about a document by using this class.

Setting the properties and methods of this class is equivalent to choosing File - Document Properties, choosing Document, and clicking the Fields panel.

## **Word Pro: Documents class**

A collection of text document objects in the Word Pro application.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

Application in the Documents Property

### **Usage**

Use this collection to access any open documents in a WordPro session.



## **Word Pro: Document class**

An abstract class that describes the basic top-level container for data in a Lotus application. Each Lotus application defines its own subclass for Document. In Word Pro, the subclass is TextDocument. You should use TextDocument and its class members when working with Word Pro documents.

### **Base Classes**

BaseObject

### **Derived Classes**

TextDocument

### **Contained by**

### **Usage**

In Word Pro you can open documents and create new documents using the OpenDocument and CreateDocument methods found in the WPAApplication class.

**Word Pro: DocWindowCollection class**

A collection of DocWindow objects in the ApplicationWindow class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

ApplicationWindow in the DocWindows Property

**Usage**

Use this collection to access any of the DocWindow objects in the ApplicationWindow class.

**Word Pro: DocWindow class**

DocWindow is the class of the document window.

**Base Classes**

BaseObjectWindow

**Derived Classes**

None

**Contained by**

Application in the ActiveDocWindow Property

**Usage**

**Word Pro: EditorCollection class**

A collection of editor objects in the EditorManager class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

EditorManager in the Editors Property

**Usage**

Use this collection to access any of the editor objects in the EditorManager class.

## **Word Pro: EditorManager class**

A tool for managing editor objects in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the EditorManager Property

TextDocument in the EditorManager Property

### **Usage**

The EditorManager class allows you to add, remove, or access editors in a document.

**Word Pro: Editor class**

Represents the information associated with a specific editor of a document.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

EditorManager in the CurrentEditor Property

**Usage**

While writing a script, be careful how you change editing options for yourself as the current editor. Carelessly changing access options for the current editor could cause you to accidentally lock yourself out of the document.

## **Word Pro: EndnoteDivisionGroupNum class**

Controls the way endnote numbers display at the end of a division group.

### **Base Classes**

BaseObject\FooterNumOpt

### **Derived Classes**

None

### **Contained by**

FootnoteOptions in the EndnoteDivisionGroupNum Property

### **Usage**

You can set endnote numbers at the end of a division group by assigning the Reset OptionEachDivisiongroup value to the ResetWhen property. This value increases endnote numbers through a division group and resets with the first endnote in the next division group. For information, see ResetWhen property.

**Word Pro: EndnoteDivisionNum class**

Controls the way endnote numbers display at the end of a current division.

**Base Classes**

BaseObject\FooterNumOpt

**Derived Classes**

None

**Contained by**

FootnoteOptions in the EndnoteDivisionNum Property

**Usage**

You can set endnote numbers at the end of a current division by assigning the ResetWhenOptionEachDivision value to the ResetWhen property. This value increases endnote numbers throughout the division and resets with the first endnote in the next division. For information, see ResetWhen property.



**Word Pro: EndnoteDocNum class**

Controls the way endnote numbers display at the end of a document.

**Base Classes**

BaseObject\FooterNumOpt

**Derived Classes**

None

**Contained by**

FootnoteOptions in the EndnoteDocNum Property

**Usage**

You can set endnote numbers at the end of a document by assigning the ResetWhenOptionEachDoc value to the ResetWhen property. This value increases endnote numbers each time you add a new endnote and continues increasing throughout the document. For information, see ResetWhen property.

**Word Pro: EndnoteLayoutCollection class**

A collection of endnote layouts in the foundry of a specific division, document, or application.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Endnotes Property

**Usage**

Use this collection to access any of the endnote objects in th foundry of a specific division, document, or application.

**Word Pro: EndnoteLayout class**

The layout for an endnote object.

**Base Classes**

BaseObject\Layout\TableLayout

**Derived Classes**

[FootnoteLayout](#)

**Contained by****Usage**

The default property settings should remain as they are. Changing any endnote layout properties will cause a script not to work properly.

## **Word Pro: FileProtection class**

The file protection options available in the TeamSecurity dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

WPApplication in the FileProtection Property

### **Usage**

FileProtection is used to set all of the individual file protection options in TeamSecurity. For example, a user can protect a file from being edited, mailed, or saved. A user can use this class to indicate that he wants to use TeamSecurity. WPApplication is the only class that contains the FileProtection class.

## **Word Pro: FilterHelper class**

Helps a filter convert non-Word Pro file formats to Word Pro file formats.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Filter in the FilterHelper Property

### **Usage**

**Word Pro: Filter class**

Converts non-Word Pro file formats to a Word Pro file format.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

ApplicationWindow in the Filter Property

**Usage**

## **Word Pro: FindAndReplace class**

The Find & Replace tool in the Word Pro application.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

WPAApplication in the FindAndReplace Property

WPAApplication in the TempFindAndReplace Property

### **Usage**

You can use the Find & Replace feature to find and replace text, paragraph styles and special characters in a document. Always runs in a default state in the Word Pro application.

## **Word Pro: FontMetrics class**

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Font in the FontMetrics Property

### **Usage**



## Word Pro: Font class

All of the font and text style properties associated with an object that has text.

### Base Classes

BaseObject

### Derived Classes

None

### Contained by

[CharacterStyle](#) in the [Font](#) Property

[ClickHere](#) in the [Font](#) Property

[Editor](#) in the [InsertFont](#) Property

[Editor](#) in the [DeleteFont](#) Property

[FindAndReplace](#) in the [FindFont](#) Property

[FindAndReplace](#) in the [ReplaceFont](#) Property

[FormatPreferences](#) in the [Font](#) Property

[Formula](#) in the [Font](#) Property

[Graphic](#) in the [Font](#) Property

[ParagraphStyle](#) in the [Font](#) Property

[RevisionDisplay](#) in the [InsertFont](#) Property

[RevisionDisplay](#) in the [DeleteFont](#) Property

[Text](#) in the [Font](#) Property

[TextMarker](#) in the [Font](#) Property

### Usage

**Word Pro: FooterLayoutCollection class**

A collection of footer layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Footers Property

**Usage**

Use this collection to access any of the footer layout objects in the foundry of a specific division.

## **Word Pro: FooterLayout class**

The layout for a footer object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: FootnoteCollection class**

A collection of footnote objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Footnotes Property

**Usage**

Use this collection to access any of the footnote objects in the foundry of a specific division.

**Word Pro: Footnote class**

A footnote object in a document.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by****Usage**

## **Word Pro: Accelerators class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddAccelerators

DeleteMacroAccelerator

RemovePersistentAccelerators

### **Events**

None

## **Word Pro: Alignment class members**

### **Properties**

AlignmentChar AS Integer

AlignmentType AS AlignmentType

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Position AS Long (measured in Twips)

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: Amikake class members**

### **Properties**

AmikakeType AS Integer

Application AS WPApplication

Background AS Background

Description AS String

IsAmikake AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None



## **Word Pro: ApplicationWindow class members**

### **Properties**

Accelerators AS Accelerators  
Active AS Integer (Boolean)  
ActiveDocument AS TextDocument  
Application AS WPApplication  
Caption AS String  
ContextMenuOptions AS ContextMenuOptions  
Description AS String  
DocWindows AS DocWindowCollection  
Filter AS Filter  
FreeMenus AS MenuItem  
Height AS Long (measured in Twips)  
HorzRuler AS Ruler  
Hwnd AS Long  
IconBarManager AS IconBarManager  
IsValid AS Integer (Boolean)  
Left AS Long (measured in Twips)  
LwpMenuBar AS MenuItem  
Macro AS Macro  
Name AS String  
Parent AS BaseObject  
ReviewVersions AS ReviewVersions  
RightMouseMenus AS MenuItem  
Script AS Script  
SectionTabs AS SectionTabs  
SetTabsDialog AS SetTabsDialog  
StatusBar AS StatusBar  
StatusBarVisible AS Integer (Boolean)  
TitleBarVisible AS Integer (Boolean)  
Top AS Long (measured in Twips)  
UserInterfacePrefs AS UserInterfacePrefs  
VersionID AS Long  
VertRuler AS Ruler  
Visible AS Integer (Boolean)  
Width AS Long (measured in Twips)

### **Methods**

Cascade  
Close  
Close  
DarkMode  
MacroRecord  
Maximize  
Minimize  
Move  
Open

Play

Resize

Restore

SaveUserDefaults

Tile

Update

**Events**

Moved

## **Word Pro: Application class members**

### **Properties**

ActiveDocument AS TextDocument

ActiveDocWindow AS DocWindow

Application AS WPApplication

ApplicationWindow AS ApplicationWindow

DefaultFilePath AS String

Description AS String

Documents AS Documents

FullName AS String

Interactive AS Integer (Boolean)

IsValid AS Integer (Boolean)

Language AS String

Location AS String

Name AS String

Parent AS BaseObject

Path AS String

VersionID AS Long

Visible AS Integer (Boolean)

### **Methods**

GetEnum

NewDocument

OpenDocument

Quit

### **Events**

DocumentCreated

DocumentOpen

DocumentOpened

Quit

## **Word Pro: AppViewPrefs class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

MarginColor AS Color

Name AS String

PaneColor AS Color

Parent AS BaseObject

SelectionBorderColor1 AS Color

SelectionBorderColor2 AS Color

SelectionBorderColor3 AS Color

SpellColor AS Color

SpellFocusedColor AS Color

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Attributes class members**

### **Properties**

Application AS WPApplication

BaselineOffset AS Long (measured in Twips)

Description AS String

HiddenMode AS Integer (Boolean)

HideOutlineLevels AS Integer

HighLightMode AS Integer (Boolean)

IsDoubleWordError AS Integer (Boolean)

IsGrammarError AS Integer (Boolean)

IsHiddenMark AS Integer (Boolean)

IsMisspelled AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

NoHyphenate AS Integer (Boolean)

Parent AS BaseObject

ProtectedMode AS Integer (Boolean)

SkipWordMode AS Integer (Boolean)

VersionID AS Long

### **Methods**

Clear

RevertToStyle

### **Events**

None

## **Word Pro: AutoRunMacro class members**

### **Properties**

Application AS WPAApplication

CloseDocMacroName AS String

Description AS String

IsValid AS Integer (Boolean)

Name AS String

NewDocMacroName AS String

OpenDocMacroName AS String

Parent AS BaseObject

RunOnCloseDoc AS Integer (Boolean)

RunOnNewDoc AS Integer (Boolean)

RunOnOpenDoc AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Background class members**

### **Properties**

Application AS WPAApplication

BackColor AS Color

BackColorIndex AS Integer

Color AS Color

Description AS String

ForeColorIndex AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Pattern AS Fill

Shape AS Integer

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: BagCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: Bag class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Length AS Long

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

Clear

DeleteBag

Read

Reset

Write

### **Events**

None

## **Word Pro: BaseCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

### **Events**

None

## **Word Pro: BaseContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: BaseObject class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## Word Pro: BaseTable class members

### Properties

Application AS WPAApplication  
CanEmbed AS Integer (Boolean)  
CellLayouts AS StringCollection  
ClassName AS String  
ColumnLayouts AS StringCollection  
ContentLayouts AS StringCollection  
ContentType AS ContentType  
CurrentCell AS CellLayout  
CurrentColumn AS Layout  
CurrentRow AS RowLayout  
DefCellStyleName AS String  
DefColWidth AS Long (measured in Twips)  
DefRowHeight AS Long (measured in Twips)  
Description AS String  
EndingColOfSelection AS Integer  
EndingRowOfSelection AS Integer  
IsAutoGrow AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsParagraphNumberingDown AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsResetParagraphNumber AS Integer (Boolean)  
IsSizingViaMouse AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxBottomBorder AS Long (measured in Twips)  
MaxBottomGutter AS Long (measured in Twips)  
MaxLeftBorder AS Long (measured in Twips)  
MaxLeftGutter AS Long (measured in Twips)  
MaxNumColsAllowed AS Integer  
MaxNumRowsAllowed AS Integer  
MaxRightBorder AS Long (measured in Twips)  
MaxRightGutter AS Long (measured in Twips)  
MaxSplitCols AS Integer  
MaxSplitRows AS Integer  
MaxTopBorder AS Long (measured in Twips)  
MaxTopGutter AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
Parent AS BaseObject  
RowLayouts AS StringCollection  
SelectionType AS SelectionType  
SingleCellSelected AS Integer (Boolean)  
StartingColOfSelection AS Integer

StartingColStringOfSelection AS String

StartingRowOfSelection AS Integer

TableFill AS TableFill

TableLine AS TableLine

VersionID AS Long

### **Methods**

CellLayout

Connect

Copy

DeleteTable

DisconnectCells

DoesMarkerNameMatch

FindCellLayout

GetMarkerName

GoToTableCell

InsertRowOrColumn

Mark

NextItem

PreviousItem

SelectTableItem

Split

### **Events**

None

## **Word Pro: BookmarkCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: BookmarkManager class members**

### **Properties**

Application AS WPAApplication

Bookmarks AS BookmarkCollection

BookmarksByMarkerName AS BookmarkCollection

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddBookmark

Find

GetUniqueName

RemoveBookmark

### **Events**

None

## **Word Pro: Bookmark class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsExportedToNotesFX AS Integer (Boolean)

IsExportedToOldNotesFX AS Integer (Boolean)

IsLinked AS Integer (Boolean)

IsOnClipboard AS Integer (Boolean)

IsValid AS Integer (Boolean)

MarkerName AS String

Name AS String

Parent AS BaseObject

VersionID AS Long

WasPasted AS Integer (Boolean)

### **Methods**

None

### **Events**

None

## **Word Pro: BorderLines class members**

### **Properties**

AllBorders AS Border

Application AS WPApplication

BottomBorder AS Border

Description AS String

IsValid AS Integer (Boolean)

LeftBorder AS Border

LinePlacement AS LinePlacement

LineValid AS LinePlacement

Name AS String

Parent AS BaseObject

RightBorder AS Border

TopBorder AS Border

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Border class members**

### **Properties**

Application AS WPAApplication

BackColorIndex AS Integer

Color AS Color

Description AS String

ForeColorIndex AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Pattern AS BorderPattern

Style AS Integer

VersionID AS Long

Width AS Integer

WidthInLongtwips AS Long (measured in Twips)

### **Methods**

None

### **Events**

None

## **Word Pro: Breaks class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsColumnBreakAfter AS Integer (Boolean)

IsColumnBreakBefore AS Integer (Boolean)

IsPageBreakAfter AS Integer (Boolean)

IsPageBreakBefore AS Integer (Boolean)

IsPageBreakWithin AS Integer (Boolean)

IsValid AS Integer (Boolean)

KeepWithNext AS Integer (Boolean)

KeepWithPrev AS Integer (Boolean)

Name AS String

NextStyleName AS String

Parent AS BaseObject

UseNextStyle AS Integer (Boolean)

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: Bullet class members**

### **Properties**

Application AS WPAApplication

Description AS String

Editable AS CommandState

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

RightAlign AS CommandState

SilverBullet AS SilverBullet

Skipped AS CommandState

Text AS Text

Valid AS CommandState

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: CellCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: CellContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
CellLayout AS CellLayout  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer



Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: CellEngine class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

GetFormula

SetFormula

### **Events**

None

## Word Pro: CellGroupLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: CellLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## Word Pro: CellLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
Formula AS String  
GetValue AS String  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)

IsBadReference AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)  
IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS Layout  
Layer AS String  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)

LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer  
LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)

TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer  
UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

#### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet

Update

ValidateValue

**Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: CharacterBorder class members**

### **Properties**

Application AS WPApplication

BorderLines AS BorderLines

Description AS String

IsBorder AS Integer (Boolean)

IsValid AS Integer (Boolean)

MarginBottom AS Long (measured in Twips)

MarginLeft AS Long (measured in Twips)

MarginRight AS Long (measured in Twips)

MarginTop AS Long (measured in Twips)

Name AS String

Parent AS BaseObject

VersionID AS Long

WidthAbove AS Long (measured in Twips)

WidthBelow AS Long (measured in Twips)

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: CharacterSet class members**

### **Properties**

Application AS WPAApplication

CharSet AS CharSet

Description AS String

IsValid AS Integer (Boolean)

LeaderDotDashChar AS Integer

LeaderDotDotChar AS Integer

LeaderDotUnderscoreChar AS Integer

Name AS String

ParagraphSymbolChar AS Integer

Parent AS BaseObject

TabSymbolChar AS Integer

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: CharacterStyleCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: CharacterStyle class members**

### **Properties**

Amikake AS Amikake  
Application AS WPApplication  
AttrStyleName AS String  
CharacterBorder AS CharacterBorder  
Definition AS Long  
Description AS String  
FaceStyleName AS String  
Font AS Font  
FontStyleName AS String  
InUseCount AS Long  
IsLocal AS Integer (Boolean)  
IsPrivate AS Integer (Boolean)  
IsTemp AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Language AS Language  
Name AS String  
Parent AS BaseObject  
SizeStyleName AS String  
StyleName AS String  
TextAttributes AS Attributes  
Type AS CharStyleType  
VersionID AS Long  
WPDataSets AS WPDataSetCollection

### **Methods**

GetNamedProperty  
HasProperty  
IsTemporary  
RegisterWPDataSet  
RemoveProperty  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

None

**Word Pro: ClickHereCollection class members**

**Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

**Methods**

IsEmpty

Item

**Events**

None

## Word Pro: ClickHere class members

### Properties

Action AS Integer  
Alignment AS Alignment  
AllowListEdit AS Integer (Boolean)  
AllowListMultiValues AS Integer (Boolean)  
Amikake AS Amikake  
Application AS WPAApplication  
AtBeginning AS Integer (Boolean)  
AtBeginningOfLine AS Integer (Boolean)  
AtBeginningOfObject AS Integer (Boolean)  
AtBeginningOfParagraph AS Integer (Boolean)  
AtBeginningOfWord AS Integer (Boolean)  
AtEnd AS Integer (Boolean)  
AtEndOfLine AS Integer (Boolean)  
AtEndOfObject AS Integer (Boolean)  
AtEndOfParagraph AS Integer (Boolean)  
AtEndOfWord AS Integer (Boolean)  
Attributes AS Attributes  
Breaks AS Breaks  
Bullet AS Bullet  
CharacterBorder AS CharacterBorder  
CharacterStyle AS CharacterStyle  
CharacterStyleName AS String  
CodePage AS Integer  
ColumnNumber AS Integer  
ColumnWidth AS Long (measured in Twips)  
CurrentLanguage AS Languages  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionsRequired AS Integer  
EffectiveColumnWidth AS Long (measured in Twips)  
FieldType AS String  
First AS String  
Font AS Font  
FormatCheckLevel AS Integer  
FormatCheckRule AS String  
FormatCheckSuggestion AS String  
HasLocalTabs AS CommandResponse  
HelpText AS String  
Indent AS Indent  
IsChanged AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsInBulletEditMode AS Integer (Boolean)  
IsMarkerValid AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)

IsParaDemandLoad AS Integer (Boolean)  
IsParagraphParent AS Integer (Boolean)  
IsPrivate AS Integer (Boolean)  
IsPrompting AS Integer (Boolean)  
IsRegistered AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsRevisionMark AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Kinsoku AS Kinsoku  
Language AS Language  
Last AS String  
LastEditorName AS String  
Layout AS Layout  
LayoutName AS String  
MaintainEditor AS Integer  
MarkerClass AS MarkerType  
MultiCompareParaTag AS String  
MultiCompareParaTagSet AS Integer (Boolean)  
Name AS String  
NextClickHere AS String  
NextText AS String  
NormalParagraph AS Integer (Boolean)  
Numbering AS Numbering  
NumberOfCharacters AS Long  
NumberOfRevisions AS Integer  
NumCharsInParagraph AS Long  
NumCols AS Integer  
NumRows AS Integer  
ObjectType AS String  
PageNumber AS Integer  
PageNumberAsText AS String  
ParagraphHasText AS Integer (Boolean)  
ParagraphBorder AS ParagraphBorder  
ParagraphStyle AS ParagraphStyle  
ParagraphStyleName AS String  
Parent AS BaseObject  
Partial AS Integer (Boolean)  
PositionXInContainer AS Long (measured in Twips)  
PositionXOnPage AS Long (measured in Twips)  
PositionYInContainer AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
PreviousClickHere AS String  
Prompt AS Text  
PromptHidden AS Integer (Boolean)  
RelativeIndent AS RelativeIndent  
RenderedPageNumber AS String  
RevisionMark AS Revision

SectionName AS String  
SelectionHidden AS Integer (Boolean)  
SelectionType AS SelectionType  
Spacing AS Spacing  
StartColumns AS Integer  
StartRow AS Integer  
StateID AS Long  
StyleExceptions AS Long  
TabExits AS Integer (Boolean)  
TabOrder AS Long  
TabRack AS TabRack  
TextViewAttributes AS TextView  
UsesHelp AS Integer (Boolean)  
VersionID AS Long  
WPDataSets AS WPDataSetCollection  
WPDataSets AS WPDataSetCollection

#### **Methods**

Add  
AdjustShade  
Backspace  
Backward  
Bisect  
CalcSmartLevels  
CalculateSmartLevels  
Clear  
ClearInternalSpellInfo  
CloseObject  
ContractOutlineLevel  
DeleteChars  
DeleteContents  
DeleteMarker  
Demote  
Deselect  
DivideText  
Embed  
ExpandOutline  
Find  
FormatCheckAction  
FormatCheckReplace  
Forward  
GetContents  
GetCopyFormatCategories  
GetCount  
GetCurrentMarkerName  
GetMarkedText  
GetMispelledWord

GetNamedProperty  
GetParagraphNumber  
GetParaNumber  
GetPasteFormatCategories  
GetPosition  
GetSpellStatus  
GetSpellUserDictStatus  
GetText  
GetWordMisspelled  
GoTo  
HasNamedProperty  
InsertBreak  
InsertDocInfo  
InsertHardSpace  
InsertMarker  
InsertNumber  
InsertPageNumber  
InsertTab  
InsertText  
InternalCopy  
InternalCut  
InternalPaste  
IsMarkerEqualToSelection  
IsPointWithin  
Mark  
MarkIndexAll  
MorphSelectionToTable  
MoveDown  
MoveParagraph  
MoveToEnd  
MoveToStart  
MoveUp  
Next  
OpenObject  
Previous  
Promote  
RegisterWPDataSet  
RegisterWPDataSet  
Remove  
RemoveNamedProperty  
Replace  
ReplaceContents  
RevertToStyle  
RevisionAccept  
RevisionCancel  
Select  
SelectMarker

SetNamedProperty

SetStyle

Shade

ShowCursor

Skip

SortParagraphs

SpellWord

SplitParagraph

SRReplace

TerminateFormatCheck

TextNumber

UnregisterWPDataSet

UnregisterWPDataSet

**Events**

EnterClickHere

ExitClickHere

## **Word Pro: Color class members**

### **Properties**

Application AS WPAApplication

Blue AS Integer

Description AS String

Green AS Integer

IsValid AS Integer (Boolean)

Name AS String

Override AS ColorOverride

Parent AS BaseObject

Red AS Integer

VersionID AS Long

### **Methods**

GetRGB

RevertToStyle

SetRGB

### **Events**

None



## Word Pro: ColumnGroupLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: Column class members**

### **Properties**

Application AS WPApplication

ColumnGap AS Long (measured in Twips)

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

WidthOfColumn AS Long (measured in Twips)

### **Methods**

None

### **Events**

None

## **Word Pro: ConnectedLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: ConnectedLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
Formula AS String  
GetValue AS String  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)



IsBadReference AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)  
IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS Layout  
Layer AS String  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)

LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer  
LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)

TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer  
UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

#### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet

Update

ValidateValue

**Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: ContentCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: Content class members**

### **Properties**

Application AS WPAApplication

CanEmbed AS Integer (Boolean)

ClassName AS String

ContentType AS ContentType

Description AS String

IsEmpty AS Integer (Boolean)

IsReplaceable AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: ContextMenuOptions class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsCellMenuEnabled AS Integer (Boolean)

IsFrameMenuEnabled AS Integer (Boolean)

IsGraphicMenuEnabled AS Integer (Boolean)

IsParallelColumnsMenuEnabled AS Integer (Boolean)

IsTextMenuEnabled AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: DdeLinkCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: DdeLinkManager class members**

### **Properties**

Application AS WPAApplication

DdeLinks AS DdeLinkCollection

DdeLinksFromMarker AS DdeLinkCollection

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddDdeLink

GetMarkerName

InsertLink

MakeUniqueLinkName

RemoveDdeLink

RequestAndProcessData

### **Events**

None

## **Word Pro: DdeLink class members**

### **Properties**

Application AS WPAApplication

Description AS String

GetConversationHandle AS Long

GetFormatName AS String

GetItemName AS String

GetServerName AS String

GetStatus AS Integer

GetTopicName AS String

IsExportedToNotesFX AS Integer (Boolean)

IsUpdateAutomatic AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

UserLinkName AS String

VersionID AS Long

### **Methods**

EditLinkInfo

### **Events**

None

## **Word Pro: DivisionCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: DivisionInfo class members**

### **Properties**

AnyOLEDDLinks AS Integer (Boolean)

Application AS WPApplication

Changed AS Integer (Boolean)

ClassName AS String

Color AS Color

ContentName AS String

Description AS String

ExternalFileID AS String

ExternalFileName AS String

ExternalName AS String

ExternalType AS String

FillerPageText AS Text

HasContents AS Integer (Boolean)

IgnoreTab AS Integer (Boolean)

IsExpandRight AS Integer (Boolean)

IsExternalFile AS Integer (Boolean)

IsScrollable AS Integer (Boolean)

IsValid AS Integer (Boolean)

LayoutName AS String

LayoutOverride AS LayoutOverride

Name AS String

PageNumberStyle AS NumberingStyle

Parent AS BaseObject

ShowTabs AS Integer (Boolean)

SuppressHeaders AS Integer

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: DivisionOptions class members**

### **Properties**

Application AS WPAApplication

Description AS String

HyphenationOptions AS HyphenationOptions

IsTextLocked AS Integer (Boolean)

IsValid AS Integer (Boolean)

Language AS Language

Name AS String

Parent AS BaseObject

ShowHiddenText AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## Word Pro: Division class members

### Properties

Application AS WPAApplication  
BookmarkManager AS BookmarkManager  
CanCreatePreviewBitmap AS Integer (Boolean)  
CreatePreviewBitmap AS Integer (Boolean)  
DdeLinkManager AS DdeLinkManager  
DdeOutboundInfo AS Long  
DemandLoad AS ReservedParam  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionNames AS StringCollection  
DivisionOptions AS DivisionOptions  
Divisions AS DivisionCollection  
DocControl AS DocControl  
DocInfo AS DocInfo  
DocOptions AS Options  
EditorManager AS EditorManager  
EditorName AS String  
Epoch AS String  
FinishedSpellChecking AS Integer  
FirstChild AS String  
FirstDivision AS String  
FirstName AS String  
FirstPage AS Integer  
FootnoteOptions AS FootnoteOptions  
Foundry AS Foundry  
HasIndex AS Integer (Boolean)  
HasTOC AS Integer (Boolean)  
IsChanged AS Integer (Boolean)  
IsChangedOtherThanLinkTo AS Integer (Boolean)  
IsChangedSinceTimeSave AS Integer (Boolean)  
IsChangedToLinkTo AS Integer (Boolean)  
IsDivisionExternal AS Integer (Boolean)  
IsDocLoading AS Integer (Boolean)  
IsEmptyDoc AS Integer (Boolean)  
IsLockedForRevisions AS Integer (Boolean)  
IsProtected AS Integer  
IsSpellBarUp AS Integer (Boolean)  
IsStyleSheet AS Integer (Boolean)  
IsUndoOn AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
LastChild AS String  
LastDivision AS String  
LastName AS String  
LastPage AS Integer

LineNumberOptions AS LineNumberOptions  
Locked AS Integer  
Master AS TextDocument  
Name AS String  
NextName AS String  
NextNeighbor AS String  
NonUserDocument AS Integer (Boolean)  
NumPagesInDoc AS Integer  
NumWindowsViewingDoc AS Integer  
Parent AS BaseObject  
ParentName AS String  
PathName AS String  
PreviousName AS String  
PreviousNeighbor AS String  
RevisionMarkMode AS Integer  
SelectionType AS SelectionType  
SortOptions AS SortOptions  
StateID AS Long  
TabSpacing AS Long (measured in Twips)  
UseContents AS Integer  
UsedFirstPageHeight AS Long (measured in Twips)  
VersionID AS Long  
VersionManager AS VersionManager  
WPDataSets AS WPDataSetCollection

#### **Methods**

AddDivision  
AnyEdits  
BeginChange  
ChangeAllEditsToEditor  
Clear  
Copy  
EndChange  
ForceDocToLoad  
FXGetNotesString  
FXGetNotesWriteHandle  
FXSetNotesString  
GetActiveList  
GetNamedProperty  
GetNameFromPage  
HasNamedProperty  
Hit  
Link  
Localize  
Move  
Purge  
RegisterWPDataSet

RemoveDivision

RemoveNamedProperty

SanityCheck

SaveDivision

SetDocumentEpoch

SetNamedProperty

Unlink

UnregisterWPDataSet

UpdatePowerFields

UpdatePowerFieldsOnNew

**Events**

EnterClickHere

EnterLayout

ExitClickHere

KeyStroke

MouseDown

MouseUp

WMCommand



## **Word Pro: DocControl class members**

### **Properties**

AllowAlternateVerification AS Integer (Boolean)  
Application AS WPApplication  
AutoVersion AS AutoVersion  
Description AS String  
DisableClickHeres AS Integer (Boolean)  
DisableExportToNotes AS Integer (Boolean)  
DisableVersionReview AS Integer (Boolean)  
DocControlRestrictedToEditor AS String  
EditorVerificationType AS EditorVerifyType  
FileProtectionType AS FileProtectType  
Greeting AS String  
IsValid AS Integer (Boolean)  
Name AS String  
NotesFlow AS Integer (Boolean)  
OCXDesignMode AS Integer (Boolean)  
Parent AS BaseObject  
RequestRemarkOnClose AS Integer (Boolean)  
RequireStartupScripts AS Integer (Boolean)  
ShowDivisionTabs AS Integer (Boolean)  
UseGreeting AS Integer (Boolean)  
VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: DocInfoFieldCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: DocInfoFieldManager class members**

### **Properties**

Application AS WPApplication

Description AS String

Fields AS DocInfoFieldCollection

IsValid AS Integer (Boolean)

Name AS String

NumFields AS Integer

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddField

DeleteField

### **Events**

None

## **Word Pro: DocInfoField class members**

### **Properties**

Application AS WPApplication

Contents AS String

Description AS String

ExportToNotesFX AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: DocInfo class members**

### **Properties**

Application AS WPAApplication

AuthorName AS String

CreationDateString AS String

CreationTimeString AS String

CreationTimeValue AS Long

Description AS String

DocSize AS Long

FieldManager AS DocInfoFieldManager

IsValid AS Integer (Boolean)

Keywords AS String

LockForNotesUserName AS String

ModifiedDateString AS String

ModifiedTimeString AS String

ModifiedTimeValue AS Long

Name AS String

NumCharsInDoc AS Long

NumPagesInDoc AS Long

NumWordsInDoc AS Long

Parent AS BaseObject

TotalEditingTime AS Long

VersionID AS Long

### **Methods**

ExportAllAsNotesFX

ExportAsNotesFX

IsExportedAsNotesFX

UpdateSelectedFields

### **Events**

None

## **Word Pro: Documents class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: Document class members**

### **Properties**

Application AS WPAApplication  
Changed AS Integer (Boolean)  
Description AS String  
Embedded AS Integer (Boolean)  
FullName AS String  
IsOpen AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Location AS String  
Name AS String  
Parent AS BaseObject  
Path AS String  
PrintSettings AS PrintSettings  
ReadOnly AS Integer (Boolean)  
Saved AS Integer (Boolean)  
VersionID AS Long

### **Methods**

Activate  
Close  
CopySelection  
CutSelection  
Paste  
Print  
PrintOut  
Save  
SaveAs

### **Events**

PreClose  
Save  
SaveAs  
Saved  
SavedAs

## **Word Pro: DocWindowCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## Word Pro: DocWindow class members

### Properties

Active AS Integer (Boolean)  
Application AS WPApplication  
BlockPaint AS Integer (Boolean)  
Caption AS String  
ClientWndHeight AS Long (measured in Twips)  
ClientWndWidth AS Long (measured in Twips)  
Description AS String  
Document AS TextDocument  
ExcludeRectBottom AS Long (measured in Twips)  
ExcludeRectLeft AS Long (measured in Twips)  
ExcludeRectRight AS Long (measured in Twips)  
ExcludeRectTop AS Long (measured in Twips)  
FitType AS FitType  
GapBetweenPanels AS Long (measured in Twips)  
HasFocus AS Integer (Boolean)  
Height AS Long (measured in Twips)  
HorizScrollBarVisible AS Integer (Boolean)  
Hwnd AS Long  
IsCentered AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Left AS Long (measured in Twips)  
MaxHorzPaneDistance AS Long (measured in Twips)  
MaxVertPaneDistance AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRowsThatFit AS Integer  
PageNumFirstPageShowing AS Integer  
PageNumLastPageShowing AS Integer  
Parent AS BaseObject  
TileWindow AS TileType  
Top AS Long (measured in Twips)  
UseExcludeRect AS Integer (Boolean)  
UsesPalette AS Integer (Boolean)  
VersionID AS Long  
VertScrollBarVisible AS Integer (Boolean)  
ViewLevel AS Integer  
ViewType AS PresentationType  
Visible AS Integer (Boolean)  
Width AS Long (measured in Twips)  
WindowId AS Long  
WinViewPrefs AS WinViewPrefs  
XOffset AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)

### Methods

Close

Close

DarkMode

DestroyDocWindow

EndChange

Hide

Invalidate

Maximize

Minimize

Move

Open

RenderClipBitmap

RenderClipDIB

RenderClipMetafile

RenderClipPalette

Resize

Restore

SetFocus

Show

ShowScrollBar

Update

**Events**

Moved

Moved

## **Word Pro: DropCapContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: DropCapLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: DropCapLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer



UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: EditorCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: EditorManager class members**

### **Properties**

Application AS WPApplication

CurrentEditor AS Editor

Description AS String

Editors AS EditorCollection

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddEditorManager

RemoveEditor

### **Events**

None

## **Word Pro: Editor class members**

### **Properties**

Abilities AS EditAbil

Application AS WPApplication

DeleteFont AS Font

Description AS String

EditorInitials AS String

HiLiteColor AS Color

InsertFont AS Font

IsValid AS Integer (Boolean)

Locks AS EditLocks

Name AS String

Parent AS BaseObject

Suggestions AS EditSugg

TextAttributes AS Attributes

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: EndnoteDivisionGroupNum class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

LeadingText AS String

Name AS String

Parent AS BaseObject

ResetWhen AS ResetOption

StartingNumber AS Integer

TrailingText AS String

UseSuperscriptReferenceNum AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: EndnoteDivisionNum class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

LeadingText AS String

Name AS String

Parent AS BaseObject

ResetWhen AS ResetOption

StartingNumber AS Integer

TrailingText AS String

UseSuperscriptReferenceNum AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: EndnoteDocNum class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

LeadingText AS String

Name AS String

Parent AS BaseObject

ResetWhen AS ResetOption

StartingNumber AS Integer

TrailingText AS String

UseSuperscriptReferenceNum AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None



## **Word Pro: EndnoteLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: EndnoteLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: FileProtection class members**

### **Properties**

Application AS WPAApplication

ContactUponPermissionDenied AS String

Description AS String

EditorAttemptingOpen AS String

FileProtectionType AS FileProtectType

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

UserPassword AS String

VerificationResult AS Verify

VersionID AS Long

### **Methods**

GetDocDescription

GetProtectionType

GetStorageProtectionType

### **Events**

None

## **Word Pro: FilterHelper class members**

### **Properties**

Application AS WPApplication

Description AS String

GetFilterExtension AS String

GetFilterExtForDialogBox AS String

GetFilterId AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None



## **Word Pro: Filter class members**

### **Properties**

Application AS WPAApplication

AsciiCodePage AS Integer

AsciiCRLFType AS AsciiLineEnding

Description AS String

FilterHelper AS FilterHelper

GraphicExports AS StringCollection

GraphicImports AS StringCollection

IsAsciiCRLF AS Integer (Boolean)

IsAsciiKeepStyle AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

TableExports AS StringCollection

TableImports AS StringCollection

TextandTableExports AS StringCollection

TextandTableImports AS StringCollection

VersionID AS Long

### **Methods**

GetDisplayableFilterName

GetFileDescription

GetLastUsedFilter

IsFilterTypePresent

SetLastUsedFilter

### **Events**

None

## **Word Pro: FindAndReplace class members**

### **Properties**

Application AS WPAApplication  
CharacterSet AS CharacterSet  
Description AS String  
FindExactCase AS Integer (Boolean)  
FindFont AS Font  
FindForwardDirection AS Integer (Boolean)  
FindString AS String  
FindStyleName AS String  
FindWithProperties AS Integer (Boolean)  
IncludeList AS IncludeList  
IsValid AS Integer (Boolean)  
MatchType AS FindMatch  
Name AS String  
NumberFound AS Long  
NumberOfReplacements AS Long  
Parent AS BaseObject  
ReplaceAttributes AS Attributes  
ReplaceExactCase AS Integer (Boolean)  
ReplaceFont AS Font  
ReplaceLanguage AS Language  
ReplaceString AS String  
ReplaceStyleName AS String  
ReplaceWithProperties AS Integer (Boolean)  
SearchAttributes AS Attributes  
SearchLanguage AS Language  
UseFindStyle AS Integer  
UseReplaceStyle AS Integer  
VersionID AS Long  
Where AS LookWhere

### **Methods**

Reset

### **Events**

None

## **Word Pro: FontMetrics class members**

### **Properties**

Application AS WPAApplication

Description AS String

FontName AS String

IsSymbolic AS Integer (Boolean)

IsTrueType AS Integer (Boolean)

IsValid AS Integer (Boolean)

Italic AS Integer (Boolean)

Name AS String

Oblique AS Integer (Boolean)

Outline AS Integer (Boolean)

Parent AS BaseObject

PitchAndFamily AS Integer

Shadow AS Integer (Boolean)

SmallCaps AS Integer (Boolean)

VersionID AS Long

Weight AS Integer

### **Methods**

None

### **Events**

None

## Word Pro: Font class members

### Properties

ActualName AS String  
Align AS Integer  
AlternateName AS String  
Application AS WPApplication  
Ascent AS Points  
BackColor AS Color  
BackColorIndex AS Integer  
Bold AS Integer (Boolean)  
Case AS Case  
DefaultPitch AS Integer  
Descent AS Points  
Description AS String  
DoubleUnderline AS Integer (Boolean)  
FaceName AS String  
FontColor AS Color  
FontMetrics AS FontMetrics  
FontName AS String  
ForeColorIndex AS Integer  
Height AS Points  
IsTrueType AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Italic AS Integer (Boolean)  
LowerCase AS Integer (Boolean)  
LowerCaseAscent AS Points  
Name AS String  
Normal AS Integer (Boolean)  
Overstrike AS Integer (Boolean)  
OverstrikeCharacter AS Long  
Parent AS BaseObject  
Plain AS Integer  
Size AS Points  
SmallCaps AS Integer (Boolean)  
StrikeThrough AS Integer  
Subscript AS Integer (Boolean)  
Superscript AS Integer (Boolean)  
TextTightness AS Integer  
Underline AS Integer (Boolean)  
UpperCase AS Integer (Boolean)  
VersionID AS Long  
Width AS Points  
WindowsName AS String  
WordDoubleUnderline AS Integer (Boolean)  
WordUnderline AS Integer (Boolean)

### Methods

Clear

RevertToStyle

**Events**

None

## **Word Pro: FooterLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: FooterLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer



LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: FootnoteCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: FootnoteContSep class members**

### **Properties**

Application AS WPAApplication

BorderLines AS BorderLines

CustomLength AS Long (measured in Twips)

Description AS String

IndentFromLeft AS Long (measured in Twips)

IsFixedLength AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

SpaceAbove AS Long (measured in Twips)

SpaceBelow AS Long (measured in Twips)

UseSeparatorLine AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: FootnoteLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: FootnoteLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer



LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: FootnoteNumbering class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

LeadingText AS String

Name AS String

Parent AS BaseObject

ResetWhen AS ResetOption

StartingNumber AS Integer

TrailingText AS String

UseSuperscriptReferenceNum AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Footnote class members**

### **Properties**

Application AS WPAApplication

Content AS Content

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Number AS Integer

Parent AS BaseObject

Type AS FnType

VersionID AS Long

### **Methods**

GoTo

### **Events**

None

**Word Pro: Accelerators property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACCELERATORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the Accelerators class which is a short-cut key assignment for any Word Pro script or Ami Pro macro.

**Data Type**

[Accelerators](#)

**Syntax**

acceleratorsvalue = [objectreference].Accelerators

**Legal values**

Always contains an instance of the Accelerators class.

**Usage**

With the methods in this class, you can add or remove any accelerator key assignment.

## **Word Pro: ActiveDocument property**

{button ,AL('H\_APPLICATION\_CLASS;H\_APPLICATIONWINDOW\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACTIVEDOCUMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the TextDocument object which is active when this property is read. Its contents depend on which document is active at the time. You can use the global variable, CurrentDocument, instead of this property.

This is a current context property in the WPAApplication class.

### **Data Type**

[TextDocument](#)

### **Syntax**

activedocumentvalue = [objectreference].ActiveDocument

### **Legal values**

Always contains an instance of the TextDocument class.

### **Usage**

WPAApplication - Use this property from the WPAApplication object when you want to access the active Word Pro document.

ApplicationWindow - Use this property from the ApplicationWindow object when you want to access the active Word Pro document.

## **Word Pro: ActiveDocWindow property**

{button ,AL('H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACTIVEDOCWINDOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the currently active DocWindow object. The DocWindow object provides access to the most basic functions of a document window, including the position, size, and appearance of a particular document's window. This is a current context property. Its contents depend on which document is active at the time.

### **Data Type**

[DocWindow](#)

### **Syntax**

activedocwindowvalue = [objectreference].ActiveDocWindow

### **Legal values**

Always contains an instance of the DocWindow class.

### **Usage**

Use this property when you want to access the currently active Word Pro document's window.



## **Word Pro: Afid property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AFID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Variant

### **Syntax**

afid = [objectreference].Afid

### **Legal values**

### **Usage**

## **Word Pro: Alignment property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASSES;H\_OLEOBJECT\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ALIGNMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An Alignment object used for controlling the alignment attributes of a paragraph within a Text, TextMarker, or ClickHere object.

### **Data Type**

[Alignment](#)

### **Syntax**

alignmentvalue = [objectreference].Alignment

### **Legal values**

Always contains an instance of the Alignment class.

### **Usage**

**Word Pro: AllBorders property**

{button ,AL(^H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ALLBORDERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Border](#)

**Syntax**

allbordersvalue = [objectreference].AllBorders

**Legal values**

Always contains an instance of the Border class.

**Usage**

## **Word Pro: Amikake property**

{button ,AL(^H\_CHARACTERSTYLE\_CLASS;H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_AMIKAKE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Holds a text background object for the Asian language versions of Word Pro. If you are using an English language version of Word Pro, this property is not available.

### **Data Type**

[Amikake](#)

### **Syntax**

amikakevalue = [objectreference].Amikake

### **Legal values**

Always contains an instance of the Amikake class.

### **Usage**

**Word Pro: AnyNumber property**

{button ,AL('H\_NUMERICFORMAT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ANYNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[NumericFormatSubset](#)

**Syntax**

anynumbervalue = [objectreference].AnyNumber

**Legal values**

Always contains an instance of the NumericFormatSubset class.

**Usage**

## Word Pro: AppFoundry property

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPFOUNDRY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A Foundry object which contains all the objects located in the Word Pro Clipboard.

### Data Type

[Foundry](#)

### Syntax

appfoundryvalue = [objectreference].AppFoundry

### Legal values

Always contains an instance of the Foundry class.

### Usage

AppFoundry is a property on the WPApplication object (always stored in the CurrentApplication variable). It contains a Foundry object which Word Pro uses as the Clipboard. This is the same Clipboard you use when you copy or cut items in a Word Pro document. When you cut or copy a selection, Word Pro takes all the objects from your selection and places them in their respective collection objects in the Foundry object stored in the AppFoundry property.

For example, if you select some text and a table and choose Edit - Copy, Word Pro places all the objects that comprise that text and table into their respective collection objects in AppFoundry. This means that all the Layout objects are stored in the corresponding layout collection objects. All CharacterStyle objects are stored in the CharacterStyleCollection object. All CellEngine objects are stored in the CellCollection object. The text objects are stored in the TextCollection object. When you choose Edit - Paste, all of these objects are reassembled in their original form and displayed in the document at the insertion point.

Because Word Pro uses the Foundry object in AppFoundry as its Clipboard, you must exercise caution when working with AppFoundry. Any objects you place in AppFoundry will be included in the next Paste operation. Any objects you remove from AppFoundry will be excluded from the next Paste operation and may adversely affect the user's ability to paste from the Clipboard.

You can get an object from AppFoundry and store it in a variable, using the following statement:

```
myobject = CurrentApplication.AppFoundry.collectionpropertyname(itemreference)
```

In this statement, *myobject* is the variable in which you want to store the object; CurrentApplication is a global variable that always contains the WPApplication object; *collectionpropertyname* is the name of the property that contains the collection object where the object you want is stored; *itemreference* is the index that specifies the object you want.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection Classes](#).

**Note** While you may retrieve objects from AppFoundry, you should not use LotusScript to place objects in the AppFoundry collections. This could interfere with normal user operations such as Cut and Copy. When creating and storing your own Word Pro objects, use the Foundry object in the TempFoundry property.

## **Word Pro: ApplicationWindow property**

{button ,AL('H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLICATIONWINDOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the ApplicationWindow class. The object in this property represents the Word Pro application window that acts as the container for all of your document windows. Also known as the application workspace, this is the window that remains after you close all of your documents and leave Word Pro running.

### **Data Type**

[ApplicationWindow](#)

### **Syntax**

applicationwindowvalue = [objectreference].ApplicationWindow

### **Legal values**

Always contains an instance of the ApplicationWindow class.

### **Usage**

The ApplicationWindow object allows you to control the size, position, and appearance of the Word Pro application window. See the definition of the [ApplicationWindow](#) class for more information on how to make use of this property.

## **Word Pro: Application property**

{button ,AL('H\_BASEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLICATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the WPAApplication object. The Application property is inherited from BaseObject and provides universal access to the WPAApplication object.

### **Data Type**

[WPAApplication](#)

### **Syntax**

applicationvalue = [objectreference].Application

### **Legal values**

Always contains an instance of the WPAApplication class.

### **Usage**

This property always contains a pointer to the WPAApplication object so you can reach the WPAApplication object, regardless of where your focus is. In most cases, you will simply use the leading dot feature to gain access to the WPAApplication object. However, if you are controlling a Word Pro object from another application, you cannot use the leading dot notation to get the WPAApplication object. In that circumstance, you can use the Application property on the object to access the WPAApplication object and thereby all of Word Pro.



## **Word Pro: AppViewPrefs property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPVIEWPREFS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the AppViewPrefs class. The object in this property represents the view preferences for a session of Word Pro.

### **Data Type**

[AppViewPrefs](#)

### **Syntax**

appviewprefsvalue = [objectreference].AppViewPrefs

### **Legal values**

Always contains an instance of the AppViewPrefs class.

### **Usage**

Use the object in this property to manipulate the color of margins, window panes, selection borders, spelling errors, and the currently selected spelling error.

### **Word Pro: Attributes property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_FORMATPREFERENCES\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASSES',0)} [See list of classes](#)

{button ,AL(^H\_ATTRIBUTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Attributes](#)

### **Syntax**

attributesvalue = [objectreference].Attributes

### **Legal values**

Always contains an instance of the Attributes class.

### **Usage**

## **Word Pro: AutoRunMacro property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTORUNMACRO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[AutoRunMacro](#)

### **Syntax**

autorunmacrovalue = [objectreference].AutoRunMacro

### **Legal values**

Always contains an instance of the AutoRunMacro class.

### **Usage**

## **Word Pro: BackColor property**

{button ,AL(^H\_BACKGROUND\_CLASS;H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_BACKCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Color](#)

### **Syntax**

backcolorvalue = [objectreference].BackColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

## Word Pro: Background property

```
{button ,AL('H_AMIKAKE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUP  
LAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLA  
SS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAM  
ELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAY  
OUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLA  
YOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEFILL_CL  
ASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)  
} See list of classes
```

```
{button ,AL('H_BACKGROUND_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates the current background settings for a specific layout object.

[TableFill]

(Read-write) Allows you to set the foreground color, background color and pattern of a specific table object.

[Layout]

Allows you to set the foreground color, background color and pattern of a specific layout object.

## Data Type

Background

## Syntax

backgroundvalue = [objectreference].Background

[objectreference].Background = backgroundvalue

## Legal values

Always contains an instance of the Background class.

## Usage

Amikake - If you are using an English language version of Word Pro, this property as a member of Amikake is not available.

TableFill - Equivalent to choosing Table - Table Properties, selecting Table cell in the "Properties for" box, clicking Options on the Table Cell Lines & Colors panel, and selecting the desired options from the "Fill," "Background color," "Pattern," and "Pattern color" boxes.

Layout - Equivalent to opening the layout object InfoBox and selecting the desired options from the "Background color," "Pattern," and "Pattern color" boxes in the Lines & Colors panel.

### **Word Pro: BackupPaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BACKUPPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores multiple backup paths (drive and directory) for Word Pro documents.

### **Data Type**

[StringCollection](#)

### **Syntax**

backuppathsvalue = [objectreference].BackupPaths

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "Backups" field on the Locations panel of the Word Pro Preferences dialog box. Although a user can enter multiple paths in this field, Word Pro only uses the first one listed. The first path listed corresponds to the BackupPath property.

## Word Pro: Bags property

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BAGS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the BagCollection class. This object provides access to Bag objects.

### Data Type

[BagCollection](#)

### Syntax

bagsvalue = [objectreference].Bags

### Legal values

Always contains an instance of the BagCollection class.

### Usage

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Bag objects contained in that Division object. When accessed through the AppFoundry property on the WPAApplication object, the collection object in this property provides access to all the Bag objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, the collection object in this property provides access to all the Bag objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, the collection object in this property provides access to all the Bag objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection Classes](#)

[H\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## Word Pro: BaseTable property

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BASetable\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

An instance of one of the following classes:

- FootnoteTable
- Glossary
- ParallelColumns
- TableHeading
- Table

This is a current context property. The content of this property is determined by context of the Word Pro focus.

### Data Type

[BaseTable](#)

### Syntax

basetablevalue = [objectreference].BaseTable

### Legal values

This property has a data type of BaseTable. This data type allows this property to contain any object created from one of BaseTable's derived classes, including FootnoteTable, Glossary, ParallelColumns, TableHeading, and Table.

### Usage

In Word Pro, tables take many different forms and have many different uses. In addition to the standard Table object, there are Glossary objects, TableHeading objects, FootnoteTable objects, and ParallelColumns objects. Each of these objects is unique and serves a different purpose. But the classes for these objects are all derived from the same BaseTable class. By using that BaseTable class as the data type for the BaseTable property, Word Pro allows the BaseTable property to contain any object that is created from one of BaseTable's derived classes.

Whatever object is stored in BaseTable is determined by the context of Word Pro's focus. If you have a document that has a table, a footnote, a glossary, and a table heading, Word Pro gives you access to all of those objects through the BaseTable property. However, only one of those objects can be found in BaseTable at any given time. If the focus (usually your cursor) is on the Table object, that Table object will be contained in the BaseTable property. Similarly, if the focus is on a FootnoteTable object, then the BaseTable property contains that FootnoteTable object. When your focus is on a page and not in a table, this property contains the currently active ParallelColumns object.



**Word Pro: BinNames property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BINNAMES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

binnamesvalue = [objectreference].BinNames

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

**Word Pro: BookmarkManager property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BOOKMARKMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[BookmarkManager](#)

**Syntax**

bookmarkmanagervalue = [objectreference].BookmarkManager

**Legal values**

Always contains an instance of the BookmarkManager class.

**Usage**

**Word Pro: BookmarksByMarkerName property**

{button ,AL('H\_BOOKMARKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BOOKMARKSBYMARKERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enumerates bookmarks by their internal (hexidecimal) names which cannot be changed; for internal use only.

**Data Type**

[BookmarkCollection](#)

**Syntax**

bookmarksbymarkernamevalue = [objectreference].BookmarksByMarkerName

**Legal values**

Always contains an instance of the BookmarkCollection class.

**Usage**

Enumerates bookmarks by their internal (hexidecimal) names which cannot be changed. This property contains objects created from the BookmarkCollection class.

## **Word Pro: Bookmarks property**

{button ,AL('H\_BOOKMARKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BOOKMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enumerates bookmarks in a document and lists them by their user-assigned names.

### **Data Type**

[BookmarkCollection](#)

### **Syntax**

bookmarksvalue = [objectreference].Bookmarks

### **Legal values**

Always contains an instance of the BookmarkCollection class.

### **Usage**

As String. Provides the user-assigned names of all bookmarks in a document. This is the same list that displays in the Bookmarks dialog box.

## Word Pro: BorderLines property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_CHARACTERBORDER\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTE LAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTE LAYOUT\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS; H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_N OTE LAYOUT\_CLASS;H\_PAGE LAYOUT\_CLASS;H\_PARAGRAPHBORDER\_CLASS;H\_ROWGROUPLAYOUT\_C LASS;H\_ROW LAYOUT\_CLASS;H\_RUBY LAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPE RTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TABLELINE\_CLA SS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BORDERLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-Only) Allows you to access the border line information for a specific layout or table object.

## Data Type

[BorderLines](#)

## Syntax

borderlinesvalue = [objectreference].BorderLines

## Legal values

Always contains an instance of the BorderLines class.

## Usage

[TableLine]

[CellGroupLayout]

[CellLayout]

[FrameGroupLayout]

You can use this property to access the type, color or width of the border lines in a specific layout object.

[FrameLayout]

You can use this property to access the type, color or width of the border lines in a specific layout object.

[PageLayout]

You can use this property to access the type, color or width of the border lines in a specific layout object.

[SuperTableLayout]

[TOCSuperTableLayout]

You can use this property to access the type, color or width of the border lines in a specific layout object.

[ConnectedLayout]

[EndnoteLayout]

[FooterLayout]

[FootnoteLayout]

[GroupLayout]

[HeaderLayout]

[NoteLayout]

[RowLayout]

[RubyLayout]

[SuperTableGroupLayout]

[TableLine]

You can use this property to access the type, color or width of the border lines in a specific table object.

[TableHeadingLayout]

[TableLayout]

**Word Pro: BottomBorder property**

{button ,AL('H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BOTTOMBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Border](#)

**Syntax**

bottombordervalue = [objectreference].BottomBorder

**Legal values**

Always contains an instance of the Border class.

**Usage**

### **Word Pro: Breaks property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',  
0)} [See list of classes](#)

{button ,AL(^H\_BREAKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Breaks](#)

### **Syntax**

breaksvalue = [objectreference].Breaks

### **Legal values**

Always contains an instance of the Breaks class.

### **Usage**

### **Word Pro: Bullet property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',  
0)} [See list of classes](#)

{button ,AL(^H\_BULLET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Bullet](#)

### **Syntax**

bulletvalue = [objectreference].Bullet

### **Legal values**

Always contains an instance of the Bullet class.

### **Usage**



## **Word Pro: CellEngines property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CELLENGINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the CellCollection class. This object provides access to CellEngine objects.

### **Data Type**

[CellCollection](#)

### **Syntax**

cellenginesvalue = [objectreference].CellEngines

### **Legal values**

Always contains an instance of the CellCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the CellEngine objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, the collection object in this property provides access to all the CellEngine objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, the collection object in this property provides access to all the CellEngine objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, the collection object in this property provides access to all the CellEngine objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

**Word Pro: CellEngine property**

{button ,AL('H\_TABLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CELLENGINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Allows you to access the CellEngine object within a table.

**Data Type**

[CellEngine](#)

**Syntax**

cellenginevalue = [objectreference].Cell Engine

**Legal values**

Always contains an instance of the CellEngine class.

**Usage**

## Word Pro: CellLayoutStyles property

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CELLLAYOUTSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the CellLayoutCollection class. This object provides access to CellLayout objects that are used as cell styles. If there are no cell styles defined for a document, the collection object in this property will be empty.

### Data Type

[CellLayoutCollection](#)

### Syntax

celllayoutstylesvalue = [objectreference].CellLayoutStyles

### Legal values

Always contains an instance of the CellLayoutCollection class.

### Usage

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the CellLayout objects used as cell styles in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, the collection object in this property provides access to all the CellLayout objects used as styles contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, the collection object in this property provides access to all the CellLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, the collection object in this property provides access to all the CellLayout objects used as cell styles in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## Word Pro: CellLayouts property

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FOUNDRY\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CELLLAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the CellLayoutCollection class. This object provides access to all CellLayout objects including those used as cell styles.

### Data Type

[CellLayoutCollection](#)

### Syntax

celllayoutsvalue = [objectreference].CellLayouts

### Legal values

Always contains an instance of the CellLayoutCollection class.

### Usage

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the CellLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, the collection object in this property provides access to all the CellLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, the collection object in this property provides access to all the CellLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, the collection object in this property provides access to all the CellLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[Classes](#).

### **Word Pro: CellLayout property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_CELLCONTAINER\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CELLLAYOUT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the cell layout object from a cell container.

### **Data Type**

[CellLayout](#)

### **Syntax**

celllayoutvalue = [objectreference].CellLayout

### **Legal values**

Always contains an instance of the CellLayout class.

### **Usage**

**Word Pro: Cell property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CELL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the CellContainer class. This is a current context property that only contains an object when the focus of Word Pro includes a cell in a table. If there is no cell in the focus, this property is empty.

**Data Type**

[CellContainer](#)

**Syntax**

propertycellvalue = [objectreference].Cell

**Legal values**

An instance of the CellContainer class.

**Usage**

When the focus includes a cell in a table, this property contains the CellContainer object that groups together the objects that comprise the cell with the focus. You can use this property to access the Layout or other objects related to that cell.

### **Word Pro: CharacterBorder property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARACTERBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[CharacterBorder](#)

### **Syntax**

characterbordervalue = [objectreference].CharacterBorder

### **Legal values**

Always contains an instance of the CharacterBorder class.

### **Usage**

## Word Pro: CharacterSet property

{button ,AL(^H\_FINDANDREPLACE\_CLASS;H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CHARACTERSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enables the user to use special characters to expand a find and replace search. Depends on the language set in Word Pro Preferences.

### Data Type

[CharacterSet](#)

### Syntax

charactersetvalue = [objectreference].CharacterSet

### Legal values

Always contains an instance of the CharacterSet class.

### Usage

Data type is String in the specific language. Equivalent to choosing Edit - Find & Replace Text, clicking Options, and selecting an option in the "Special characters help" list box. You can then enter these values in the "Find" and "Replace with" boxes on the Find & Replace bar. The values are:

^?	Any one character	Finds and replaces any one character. Can be used with other text to find and replace variations of a word.
^*	Any characters	Finds and replaces zero or more characters in a word. Can be used with other text to find and replace variations of a word.
^+	Phrase	Finds and replaces zero or more characters across multiple words. Can be used with other text to find and replace phrases.
^p	To end of paragraph	Finds and replaces zero or more characters across multiple words to the end of a paragraph.
^t	Tab character	Finds and replaces spaces with a tab character.
^r	Return character	Finds and replaces a hard return.
^^	^character	Finds and replaces the ^ character.



## **Word Pro: CharacterStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARACTERSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the CharacterStyleCollection class. This object provides access to CharacterStyle objects.

### **Data Type**

[CharacterStyleCollection](#)

### **Syntax**

characterstylesvalue = [objectreference].CharacterStyles

### **Legal values**

Always contains an instance of the CharacterStyleCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the CharacterStyle objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, the collection object in this property provides access to all the CharacterStyle objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, the collection object in this property provides access to all the CharacterStyle objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, the collection object in this property provides access to all the CharacterStyle objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

**Word Pro: CharacterStyle property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARACTERSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[CharacterStyle](#)

**Syntax**

characterstylevalue = [objectreference].CharacterStyle

**Legal values**

Always contains an instance of the CharacterStyle class.

**Usage**

## Word Pro: ChildLayouts property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_CHILDLAYOUTS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns a collection of a layout object's children.

## Data Type

[LayoutCollection](#)

## Syntax

childlayoutsvalue = [objectreference].ChildLayouts

## Legal values

Always contains an instance of the LayoutCollection class.

## Usage

This method returns a collection of a layout object's children. For instance, a page layout object's children could include header and footer layout objects, as well as certain frames or tables.

## **Word Pro: ClickHeres property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLICKHERES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the ClickHereCollection class. This object provides access to ClickHere objects.

### **Data Type**

[ClickHereCollection](#)

### **Syntax**

clickheresvalue = [objectreference].ClickHeres

### **Legal values**

Always contains an instance of the ClickHereCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the ClickHere objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, the collection object in this property provides access to all the ClickHere objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, the collection object in this property provides access to all the ClickHere objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, the collection object in this property provides access to all the ClickHere objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## Word Pro: Color property

```
{button ,AL('H_BACKGROUND_CLASS;H_BORDER_CLASS;H_DIVISIONINFO_CLASS;H_INDEXSECTION_CLAS  
S;H_NOTELAYOUT_CLASS;H_NUMERICFORMATSUBSET_CLASS;H_SECTION_CLASS;H_SHADOW_CLAS  
S',0)} See list of classes
```

```
{button ,AL('H_COLOR_PROPERTY_EXSCRIPT',1)} See example
```

(Read-Write) Sets the color of a shadow for text, frames, headers, footers, and pages in a document.

### Data Type

[Color](#)

### Syntax

colorvalue = [objectreference].Color

[objectreference].Color = colorvalue

### Legal values

Always contains an instance of the Color class.

### Usage

You can set the Color property for text, frames, tables, headers, footers, and pages in a document. Equivalent to:

- Clicking the right mouse button anywhere in the text, choosing Text Properties, clicking the Lines & Colors tab, and selecting an option in the "Shadow" box.
- Clicking the right mouse button anywhere in the frame, choosing Frame Properties, clicking the Lines & Colors tab, and selecting an option in the "Shadow" box.
- Clicking the right mouse button anywhere in the table, choosing Table Properties, clicking the Lines & Colors tab, and selecting an option in the "Shadow" box.
- Clicking the right mouse button anywhere in the header, choosing Header Properties, clicking the Lines & Colors panel, and selecting an option in the "Shadow" box.
- Clicking the right mouse button anywhere in the footer, choosing Footer Properties, clicking the Lines & Colors tab, and selecting an option in the "Shadow" box.
- Clicking the right mouse button anywhere in the page, choosing Page Properties, clicking the Lines & Colors tab, and selecting an option in the "Shadow" box.

### **Word Pro: ColumnLayouts property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_COLUMNLAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) This object provides the names of column layout objects within a table.

### **Data Type**

[StringCollection](#)

### **Syntax**

columnlayoutsvalue = [objectreference].ColumnLayouts

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Column layout objects are created when the layout of a column is modified. For example, if you modify the width of the first column in a table, a column layout object will be created for that column only. The name of the modified column layout object will then be stored in the ColumnLayouts property.

**Word Pro: CommentColor property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COMMENTCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Color](#)

**Syntax**

commentcolorvalue = [objectreference].CommentColor

**Legal values**

Always contains an instance of the Color class.

**Usage**

## **Word Pro: ConnectedLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONNECTEDLAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the ConnectedLayoutCollection class. This object provides access to ConnectedLayoutCollection objects.

### **Data Type**

[ConnectedLayoutCollection](#)

### **Syntax**

connectedlayoutsvalue = [objectreference].ConnectedLayouts

### **Legal values**

Always contains an instance of the ConnectedLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the CellLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the CellLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the CellLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the CellLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.



## Word Pro: Container property

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTAINER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A "catch all" container property that always contains the topmost container object in the focus. Container uses the abstract class, BaseContainer, as its data type, which allows Container to store any kind of container object.

### Data Type

[BaseContainer](#)

### Syntax

containervalue = [objectreference].Container

### Legal values

An instance of the BaseContainer class.

### Usage

Use this property to access the topmost container object's layout, regardless of that container object's contents. For example, you might write a script which allows the user to select an object whose background he wants to turn red. After the user selects the object (thus setting the focus on that object), you can issue this statement:

```
.Container.Layout.Background.Color.SetRGB 255, 0, 0
```

This statement sets the color of the layout's background to red, regardless of the kind of layout or the kind of container object the user selected.

**Word Pro: ContentLayouts property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CONTENTLAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) This object provides access to all ContentLayout objects including those used as content styles.

**Data Type**

[StringCollection](#)

**Syntax**

contentlayouts[objectreference].ContentsLayouts

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

This object is created from the ContentLayoutCollection class.

## Word Pro: Content property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTE_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTE_LAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WPAAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_CONTENT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) An instance of one of the content classes. A content class is any class derived from the abstract class called Content. Each content class describes a different kind of content, such as text, a graphic, or a cell formula. This property uses Content as its data type so it can store any object created from one of these content classes.

For more details on the Content class, content classes, and content objects, click the data type below.

## Data Type

[Content](#)

## Syntax

contentvalue = [objectreference].Content

## Legal values

This property can contain an instance of one of the following classes:

- FootnoteTable
- Formula
- Glossary
- Graphic
- OleObjects
- ParallelColumns
- SuperTable
- TableHeading
- Table
- Text

## Usage

The type of content object you find in this property depends on the object you call the property from, and what objects are in the focus when you call the Content property.

### WPAApplication.Content

If you call the Content property on the WPAApplication object, you will get the uppermost content object within the focus. Thus, the content object in this property changes as the focus changes in your document. This content object can be any of the types listed above, under Legal values.

### Footnote.Content

If you call the Content property on a Footnote object, you will get the content object for that footnote. Again, this content object can be any of the types listed above, under Legal values.

### layoutobject.Content

The Layout class has a Content property which is inherited by each layout class. A layout class is any class derived from the Layout class. A layout object is any object created from one of the layout classes. If you call the Content property on a layout object, you get the content object of that layout object. Usually, the type of content object found in the Content property corresponds to the type of object represented by that layout.

For example, the Content property on a CellLayout object may contain a Formula content object. But the Content property on a PageLayout object will most likely be a Text content object. However, you must keep in mind that a cell or a page can also contain a graphic or an OLE object, and this will affect the type of content object stored in the Content property. The types of layout objects that have a Content property are:

CellGroupLayout  
CellLayout  
ConnectedLayout  
EndnoteLayout  
FooterLayout  
FootnoteLayout  
FrameLayout  
GroupLayout  
HeaderLayout  
NoteLayout  
PageLayout  
RowLayout  
RubyLayout  
SuperTableGroupLayout  
SuperTableLayout  
TableHeadingLayout  
TableLayout  
TOCSuperTableLayout

Note that some of these layout objects are stored in properties of other objects, such as CellLayout, which can be found in the CurrentCell property on WPApplication, and the CellLayout property on CellContainer. Other layout objects listed above may not be stored in a property, but can still be accessed through the appropriate collection object.

**Word Pro: ContextMenuOptions property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTEXTMENUOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the ContextMenuOptions class which is menu options in a context sensitive dialog box or bar.

**Data Type**

[ContextMenuOptions](#)

**Syntax**

contextmenuoptionsvalue = [objectreference].ContextMenuOptions

**Legal values**

Always contains an instance of the ContextMenuOptions class.

**Usage**

Use this property to see if menus or menu items are grayed.

## **Word Pro: CurrentCell property**

{button ,AL(^H\_BASetable\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CURRENTCELL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The CellLayout object for the table cell which is uppermost in the focus. If no cell is in the focus, this property is empty.

### **Data Type**

[CellLayout](#)

### **Syntax**

currentcellvalue = [objectreference].CurrentCell

### **Legal values**

Always contains an instance of the CellLayout class.

### **Usage**

In most cases, you can use this property to access the cell in which the insertion point is located. However, if the insertion point is in a frame that is in a table cell, this property still contains the CellLayout object for the cell containing the frame.

## Word Pro: CurrentColumn property

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CURRENTCOLUMN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the layout object for the column or columns uppermost in the focus.

### Data Type

[Layout](#)

### Syntax

currentcolumnvalue = [objectreference].CurrentColumn

### Legal values

This property can contain an instance of the Layout class or any of its derived classes.

### Usage

If there is no table in the focus, this property is empty.

If only one cell is selected, this property contains the CellLayout object for that cell.

If more than one cell in the same column is selected, this property contains a CellGroupLayout object representing the cells selected in that column.

If more than one cell is selected across multiple columns, this property contains a CellGroupLayout object representing the cells selected in that column.

## **Word Pro: CurrentEditor property**

{button ,AL('H\_EDITORMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CURRENTEDITOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The editor object for the current editor of a document.

### **Data Type**

[Editor](#)

### **Syntax**

currenteditorvalue = [objectreference].CurrentEditor

### **Legal values**

Always contains an instance of the Editor class.

### **Usage**

This property gives you access to the editor object for the current editor. Therefore, you can use the CurrentEditor property to find out the name of the current editor of a document.



### **Word Pro: CurrentRow property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CURRENTROW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the layout object for the row(s) uppermost in the focus.

### **Data Type**

[RowLayout](#)

### **Syntax**

currentrowvalue = [objectreference].CurrentRow

### **Legal values**

This property can contain an instance of the Layout class or any of its derived classes.

### **Usage**

If there is no table in the focus, this property is empty.

If only one cell is selected, this property contains the CellLayout object for that cell.

If more than one cell in the same row is selected, this property contains a CellGroupLayout object representing the cells selected in that row.

If more than one cell is selected across multiple rows, this property contains a CellGroupLayout object representing the cells selected in that row.

**Word Pro: CurrentVersion property**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CURRENTVERSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Version](#)

**Syntax**

currentversionvalue = [objectreference].CurrentVersion

**Legal values**

Always contains an instance of the Version class.

**Usage**

**Word Pro: DataNames property**

{button ,AL('H\_SCRIPTDATASET\_CLASS;H\_WPDATASET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATANAMES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) All the variable names in a data set.

**Data Type**

[StringCollection](#)

**Syntax**

datanamesvalue = [objectreference].DataNames

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

You can use the DataNames property to return all the existing variable names in a specific data set.

**Word Pro: DdeLinkManager property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DDELINKMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[DdeLinkManager](#)

**Syntax**

ddelinkmanagervalue = [objectreference].DdeLinkManager

**Legal values**

Always contains an instance of the DdeLinkManager class.

**Usage**

**Word Pro: DdeLinksFromMarker property**

{button ,AL('H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DDELINKSFROMMARKER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enumerates Dde links by their internal (hexidecimal) names which cannot be changed; for internal use only.

**Data Type**

[DdeLinkCollection](#)

**Syntax**

ddelinkssfrommarkervalue = [objectreference].DdeLinksFromMarker

**Legal values**

Always contains an instance of the DdeLinkCollection class.

**Usage**

Enumerates Dde links by their internal (hexidecimal) names which cannot be changed.

**Word Pro: DdeLinks property**

{button ,AL('H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DDELINKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enumerates DdeLinks in a document and lists them by their readable names.

**Data Type**

[DdeLinkCollection](#)

**Syntax**

ddelinksvalue = [objectreference].DdeLinks

**Legal values**

As String (name). Always contains an instance of the DdeLinkCollection class.

**Usage**

Provides the readable names of all DdeLinks in a document.

## **Word Pro: DeleteFont property**

{button ,AL(^H\_EDITOR\_CLASS;H\_REVISIONDISPLAY\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETEFONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The collection of attributes associated with a font object that marks deleted text in a document.

### **Data Type**

[Font](#)

### **Syntax**

deletefontvalue = [objectreference].DeleteFont

### **Legal values**

Always contains an instance of the Font class.

### **Usage**

[Editor]

This property is equivalent to the "Markup for deletions" option, which is located in the Markup Options dialog. The Markup Options dialog can be opened by pressing the Markup Options button, which is located in the General panel of the Word Pro Preferences dialog.

**Word Pro: DirectiveColor property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIRECTIVECOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Color](#)

**Syntax**

directivecolorvalue = [objectreference].DirectiveColor

**Legal values**

Always contains an instance of the Color class.

**Usage**



## Word Pro: DivisionInfo property

```
{button ,AL(^H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CLICKHERE_CLASS;H_DIVISION_CLASS;H_DROPCAPCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_MARKER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_POWERFIELD_CLASSES;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYMARKER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEMARKER_CLASS;H_TABLEONLYCONT_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS',0)} See list of classes
```

```
{button ,AL(^H_DIVISIONINFO_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only)

[BaseContainer]

Returns the division info object from any container.

## Data Type

[DivisionInfo](#)

## Syntax

divisioninfovalue = [objectreference].DivisionInfo

## Legal values

Always contains an instance of the DivisionInfo class.

## Usage

**Word Pro: DivisionNames property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIVISIONNAMES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

divisionnamesvalue = [objectreference].DivisionNames

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

## **Word Pro: DivisionOptions property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIVISIONOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[DivisionOptions](#)

### **Syntax**

divisionoptionsvalue = [objectreference].DivisionOptions

### **Legal values**

Always contains an instance of the DivisionOptions class.

### **Usage**

### **Word Pro: Divisions property**

{button ,AL(^H\_DIVISION\_CLASS;H\_PRINTSETTINGS\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DIVISIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains a DivisionCollection object. This DivisionCollection object contains Division objects.

### **Data Type**

[DivisionCollection](#)

### **Syntax**

divisionsvalue = [objectreference].Divisions

### **Legal values**

Always contains an instance of the DivisionCollection class.

### **Usage**

When you call this property from the WPAApplication object, the DivisionCollection object contains all the divisions in the currently active document.

**Word Pro: Division property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DIVISION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains the currently active Division object.

**Data Type**

[Division](#)

**Syntax**

divisionvalue = [objectreference].Division

**Legal values**

Always contains an instance of the Division class.

**Usage**

Use this property to access the currently active Division object and any of its members.

**Word Pro: DocControl property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCCONTROL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[DocControl](#)

**Syntax**

doccontrolvalue = [objectreference].DocControl

**Legal values**

Always contains an instance of the DocControl class.

**Usage**

**Word Pro: DocInfo property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCINFO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[DocInfo](#)

**Syntax**

docinfovalue = [objectreference].DocInfo

**Legal values**

Always contains an instance of the DocInfo class.

**Usage**

## **Word Pro: DocOptions property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

## **Data Type**

[Options](#)

## **Syntax**

docoptionsvalue = [objectreference].DocOptions

## **Legal values**

Always contains an instance of the Options class.

## **Usage**



## **Word Pro: DocumentPaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores multiple paths (drive and directory) where Word Pro looks for documents to open.

### **Data Type**

[StringCollection](#)

### **Syntax**

documentpathsvalue = [objectreference].DocumentPaths

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "Documents" field on the Locations panel of the Word Pro Preferences dialog box. The "Documents" field can contain multiple document paths. You can use this property to read these multiple document paths, including the primary (default) document path which is stored in the DocPath property.

## **Word Pro: Documents property**

{button ,AL(^H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains a Documents object. This Documents object is a collection object that contains all the currently open TextDocument objects.

### **Data Type**

[Documents](#)

### **Syntax**

documentsvalue = [objectreference].Documents

### **Legal values**

Always contains an instance of the Documents class.

### **Usage**

Use this property when you want to access a TextDocument object which is open but not currently active. For more information about accessing objects in a collection, choose Help - Word Pro Objects. Click the Find tab and search for "collection."

**Word Pro: Document property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[TextDocument](#)

**Syntax**

documentvalue = [objectreference].Document

**Legal values**

Always contains an instance of the TextDocument class.

**Usage**

**Word Pro: DocWindows property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCWINDOWS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the DocWindowCollection class that is a collection of all open document windows in an application.

**Data Type**

[DocWindowCollection](#)

**Syntax**

docwindowsvalue = [objectreference].DocWindows

**Legal values**

Always contains an instance of the DocWindowCollection class.

**Usage**

Use this property to get a list of all open document windows in the currently active application window.

**Word Pro: EditorManager property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_EDITORMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[EditorManager](#)

**Syntax**

editormanagervalue = [objectreference].EditorManager

**Legal values**

Always contains an instance of the EditorManager class.

**Usage**

## **Word Pro: Editors property**

{button ,AL('H\_EDITORMANAGER\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EDITORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The collection or array of editors assigned to a document.

### **Data Type**

[EditorCollection](#)

### **Syntax**

editorsvalue = [objectreference].Editors

### **Legal values**

Always contains an instance of the EditorCollection class.

### **Usage**

**Word Pro: EndnoteDivisionGroupNum property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENDNOTEDIVISIONGROUPNUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[EndnoteDivisionGroupNum](#)

**Syntax**

endnotedivisiongroupnumvalue = [objectreference].EndnoteDivisionGroupNum

**Legal values**

Always contains an instance of the EndnoteDivisionGroupNum class.

**Usage**

**Word Pro: EndnoteDivisionNum property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENDNOTEDIVISIONNUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[EndnoteDivisionNum](#)

**Syntax**

endnotedivisionnumvalue = [objectreference].EndnoteDivisionNum

**Legal values**

Always contains an instance of the EndnoteDivisionNum class.

**Usage**



**Word Pro: EndnoteDocNum property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENDNOTEDOCNUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[EndnoteDocNum](#)

**Syntax**

endnotedocnumvalue = [objectreference].EndnoteDocNum

**Legal values**

Always contains an instance of the EndnoteDocNum class.

**Usage**

## **Word Pro: Endnotes property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENDNOTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the EndnoteLayoutCollection class. This object provides access to EndnoteLayoutCollection objects.

### **Data Type**

[EndnoteLayoutCollection](#)

### **Syntax**

endnotesvalue = [objectreference].Endnotes

### **Legal values**

Always contains an instance of the EndnoteLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the EndnoteLayoutCollection objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the EndnoteLayoutCollection objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the EndnoteLayoutCollection objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the EndnoteLayoutCollection objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## **Word Pro: ErrorColor property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ERRORCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Color](#)

### **Syntax**

errorcolorvalue = [objectreference].ErrorColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

**Word Pro: FaceNames property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FACENAMES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

facenamesvalue = [objectreference].FaceNames

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

**Word Pro: FieldManager property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIELDMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The DocInfoFieldManager object for a document.

**Data Type**

[DocInfoFieldManager](#)

**Syntax**

fieldmanagervalue = [objectreference].FieldManager

**Legal values**

Always contains an instance of the DocInfoFieldManager class.

**Usage**

You can use this property to access any DocInfo fields in a document.

**Word Pro: Fields property**

{button ,AL(`H\_DOCINFOFIELDMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_FIELDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A collection of document fields.

**Data Type**

[DocInfoFieldCollection](#)

**Syntax**

fieldsvalue = [objectreference].Fields

**Legal values**

Always contains an instance of the DocInfoFieldCollection class.

**Usage**

In Word Pro, document fields are created in the Fields panel of the Document Properties dialog box.

**Word Pro: FileProtection property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILEPROTECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FileProtection](#)

**Syntax**

fileprotectionvalue = [objectreference].FileProtection

**Legal values**

Always contains an instance of the FileProtection class.

**Usage**

### **Word Pro: FillerPageText property**

{button ,AL(^H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FILLERPAGETEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Text](#)

### **Syntax**

fillerpagetextvalue = [objectreference].FillerPageText

### **Legal values**

Always contains an instance of the Text class.

### **Usage**



**Word Pro: FilterHelper property**

{button ,AL(^H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FILTERHELPER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FilterHelper](#)

**Syntax**

filterhelpervalue = [objectreference].FilterHelper

**Legal values**

Always contains an instance of the FilterHelper class.

**Usage**

## **Word Pro: Filter property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the filter class which converts non-Word Pro file formats to a Word Pro file format.

### **Data Type**

[Filter](#)

### **Syntax**

filtervalue = [objectreference].Filter

### **Legal values**

Always contains an instance of the Filter class.

### **Usage**

Use this property to get to filter objects without having a document open.

**Word Pro: FindAndReplace property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDANDREPLACE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains the FindAndReplace object for the currently active session of Word Pro. The settings stored in this object are used by all documents opened within the active session.

**Data Type**

[FindAndReplace](#)

**Syntax**

findandreplacevalue = [objectreference].FindAndReplace

**Legal values**

Always contains an instance of the FindAndReplace class.

**Usage**

Use this property to access the FindAndReplace object, and to check and manipulate the settings for find and replace operations.

## **Word Pro: FindFont property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDFONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enables the user to find a text font in Find & Replace.

### **Data Type**

[Font](#)

### **Syntax**

findfontvalue = [objectreference].FindFont

### **Legal values**

Always contains an instance of the Font class. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property to find a font in Find & Replace. If True, finds the font that matches the user setting. Equivalent to choosing Edit - Find & Replace Text, clicking Options, clicking the Font button in the "Find options" box, and selecting a font in the "Font name" list box on the Find panel.

## **Word Pro: FontColor property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Color](#)

### **Syntax**

fontcolorvalue = [objectreference].FontColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

**Word Pro: FontMetrics property**

{button ,AL(^H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FONTMETRICS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FontMetrics](#)

**Syntax**

fontmetricsvalue = [objectreference].FontMetrics

**Legal values**

Always contains an instance of the FontMetrics class.

**Usage**

## Word Pro: Font property

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_CLICKHERE\_CLASS;H\_FORMATPREFERENCES\_CLASS;H\_FOR  
MULA\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS;H\_PARAGRA  
PHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

## Data Type

Font

## Syntax

fontvalue = [objectreference].Font

## Legal values

Always contains an instance of the Font class.

## Usage

## **Word Pro: Footers property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the FooterLayoutCollection class. This object provides access to FooterLayout objects.

### **Data Type**

[FooterLayoutCollection](#)

### **Syntax**

footersvalue = [objectreference].Footers

### **Legal values**

Always contains an instance of the FooterLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the FooterLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the FooterLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the FooterLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the FooterLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).



## Word Pro: Footer property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_FOOTER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns a footer object in a layout object.

## Data Type

[Layout](#)

## Syntax

footervalue = [objectreference].Footer

## Legal values

Always contains an instance of the Layout class.

## Usage

Use this property to access the footer layout object of a specific layout. Not all layout objects have footer layout objects. You can check the UseFooter property of a layout object to see if there is a footer layout object currently available for that layout object.

**Word Pro: FootnoteContSep property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTNOTECONTSEP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FootnoteContSep](#)

**Syntax**

footnotecontsepvalue = [objectreference].FootnoteContSep

**Legal values**

Always contains an instance of the FootnoteContSep class.

**Usage**

## **Word Pro: FootnoteLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTNOTELAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the FootnoteLayoutCollection class. This object provides access to FootnoteLayout objects.

### **Data Type**

[FootnoteLayoutCollection](#)

### **Syntax**

footnotelayoutsvalue = [objectreference].FootnoteLayouts

### **Legal values**

Always contains an instance of the FootnoteLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the FootnoteLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the FootnoteLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the FootnoteLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the FootnoteLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

**Word Pro: FootnoteNumbering property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTNOTENUMBERING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FootnoteNumbering](#)

**Syntax**

footnotenumberingvalue = [objectreference].FootnoteNumbering

**Legal values**

Always contains an instance of the FootnoteNumbering class.

**Usage**

**Word Pro: FootnoteOptions property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTNOTEOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FootnoteOptions](#)

**Syntax**

footnoteoptionsvalue = [objectreference].FootnoteOptions

**Legal values**

Always contains an instance of the FootnoteOptions class.

**Usage**

**Word Pro: FootnoteSeparator property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTNOTESEPARATOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FootnoteSeparator](#)

**Syntax**

footnoteseparatorvalue = [objectreference].FootnoteSeparator

**Legal values**

Always contains an instance of the FootnoteSeparator class.

**Usage**

## **Word Pro: Footnotes property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FOOTNOTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the FootnoteCollection class. This object provides access to Footnote objects.

### **Data Type**

[FootnoteCollection](#)

### **Syntax**

footnotesvalue = [objectreference].Footnotes

### **Legal values**

Always contains an instance of the FootnoteCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Footnote objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the Footnote objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the Footnote objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the Footnote objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## **Word Pro: FormatCheckPreferences property**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_FORMATCHECKPREFERENCES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the FormatCheckPref class which represents the options for the Format Check tool.

### **Data Type**

[FormatCheckPref](#)

### **Syntax**

formatcheckpreferencesvalue = [objectreference].FormatCheckPreferences

### **Legal values**

Always contains an instance of the FormatCheckPref class.

### **Usage**

Setting the properties in the FormatCheckPref object is equivalent to selecting options in the Format Check Options dialog box. To open this dialog box, choose Edit - Check Format and leave the Format Check bar open. Click Options on the Format Check bar.



**Word Pro: Format property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FORMAT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[FormatPreferences](#)

**Syntax**

formatvalue = [objectreference].Format

**Legal values**

Always contains an instance of the FormatPreferences class.

**Usage**

## **Word Pro: Action property**

This property is defined in the following classes:

[ClickHere](#)

[MenuItem](#)

## **Word Pro: Foundry property**

This property is defined in the following classes:

Division

TextDocument

WPAApplication

## **Word Pro: Hwnd property**

This property is defined in the following classes:

[ApplicationWindow](#)

[DocWindow](#)

[LwpBaseCtrl](#)

[LwpCustomDialog](#)

## **Word Pro: Shadow property**

This property is defined in the following classes:

FontMetrics

Layout

ParagraphBorder

Example: ActivateAs method

'This example script has not yet been created.'

Example: AddACondition method

'This example script has not yet been created.'

Example: AddNewSectionTabs method

'This example script has not yet been created.'



Example: BinNameFromNumber method

'This example script has not yet been created.'

Example: CheckFieldEntries method

'This example script has not yet been created.'

Example: ConvertTo method

'This example script has not yet been created.'

Example: CreateEmptyList method

'This example script has not yet been created.'

Example: CreateLayer method

'This example script has not yet been created.'

Example: DeleteBag method

```
' This example creates a bag in the active division and then writes some data  
' to the bag. The data from the created bag is read and printed to the Lotus  
' Script Output panel. Next, data from all bags in the Bag Collection is  
' printed and then each bag is deleted.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim BagName As String  
Dim MyBag As Bag  
Dim BagData As String
```

```
BagData = "This is data for the bag."  
LenBagData = Len(BagData)
```

```
BagName = .Division.Foundry.Create($LwpFoundryCreateTypeBag)  
Set MyBag = .Division.Foundry.Bags.Item(BagName)
```

```
Stat = MyBag.Write(BagData, LenBagData)  
If Stat = True Then  
    Print "BagData= " & MyBag.Read(LenBagData)  
End If
```

```
Forall ThisBag In .Division.Foundry.Bags  
    ThisBag.Reset  
    Print "Name = " ThisBag.Name  
    Print "Length = " ThisBag.Length  
    Print ThisBag.Read(ThisBag.Length)  
    ThisBag.DeleteBag  
End Forall
```

Example: DeleteItemByPosition method

'This example script has not yet been created.'

Example: EndPrinting method

'This example script has not yet been created.'



Example: EnvelopeBarCode method

'This example script has not yet been created.'

Example: GetAspectMetafilePict method

'This example script has not yet been created.'

Example: GetListName method

'This example script has not yet been created.'

Example: GetPageRange method

'This example script has not yet been created.'

Example: GetParagraphNumber method

'This example script has not yet been created.'

Example: GetString method

'This example script has not yet been created.'

Example: GoToClickHere method

'This example script has not yet been created.'

Example: HasProperty method

'This example script has not yet been created.'



Example: ImportWatermarkGraphic method

'This example script has not yet been created.'

Example: InternetExtraFile method

'This example script has not yet been created.'

Example: IsCaseExact method

'This example script has not yet been created.'

Example: LeastRecentVersion method

'This example script has not yet been created.'

Example: LinkContainers method

'This example script has not yet been created.'

Example: NewItemByPosition method

'This example script has not yet been created.'

Example: NextCycleAlign method

'This example script has not yet been created.'

Example: NextCycleAttribute method

'This example script has not yet been created.'



Example: NextCycleBullet method

'This example script has not yet been created.'

Example: NextCycleFontSize method

'This example script has not yet been created.'

Example: NextCycleFont method

'This example script has not yet been created.'

Example: NextCycleIndent method

'This example script has not yet been created.'

Example: NextCycleNumber method

'This example script has not yet been created.'

Example: NextCycleStyle method

'This example script has not yet been created.'

Example: OpenDocumentFromNotes method

'This example script has not yet been created.'

Example: RemoveIndexEntry method

'This example script has not yet been created.'



Example: RemoveList method

'This example script has not yet been created.'

Example: RemoveTOCEntry method

'This example script has not yet been created.'

Example: Repaint method

'This example script has not yet been created.'

Example: ReplaceContents method

'This example script has not yet been created.'

Example: ResetNumberOpts method

'This example script has not yet been created.'

Example: ResetPrinting method

'This example script has not yet been created.'

Example: RestorePreviousView method

'This example script has not yet been created.'

Example: RestoreWindowFromCleanScreen method  
'This example script has not yet been created.



Example: SanityCheck method

'This example script has not yet been created.'

Example: SaveMergeDataFile method

'This example script has not yet been created.'

Example: SetAllMargins method

'This example script has not yet been created.'

Example: SetCustomNumber method

'This example script has not yet been created.'

Example: SetIndexInfo method

'This example script has not yet been created.'

Example: SetJapanIndexInfo method

'This example script has not yet been created.'

Example: SetNoFields method

'This example script has not yet been created.'

Example: SetNumberingLevelInfo method

'This example script has not yet been created.'



Example: SetTOCLevelContent method

'This example script has not yet been created.'

Example: SetTOCLevelPageInfo method

'This example script has not yet been created.'

Example: SpecialView method

'This example script has not yet been created.'

Example: SplitDivision method

'This example script has not yet been created.'

Example: SplitWindow method

'This example script has not yet been created.'

Example: UnLinkContainers method

'This example script has not yet been created.'

Example: Unlink method

'This example script has not yet been created.'

Example: UpdatePowerFieldsOnNew method

'This example script has not yet been created.'



Example: UpdatePrinterBins method

'This example script has not yet been created.'

Example: UpdateTabs method

'This example script has not yet been created.'

Example: UpdateUI method

'This example script has not yet been created.'

Example: WordCount method

'This example script has not yet been created.'

**Word Pro: ActivateAs method**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ACTIVATEAS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

**Word Pro: AddACondition method**

{button ,AL(^H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ADDACONDITION\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: AddNewSectionTabs method**

{button ,AL('H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDNEWSECTIONTABS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: AddStringToList method**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ADDSTRINGTOLIST\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



**Word Pro: BinNameFromNumber method**

{button ,AL(`H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_BINNAMEFROMNUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: CheckFieldEntries method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CHECKFIELDENTRIES\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

**Word Pro: ConvertTo method**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONVERTTO\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: CreateEmptyList method**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEEMPTYLIST\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## Word Pro: CreateLayer method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_CREATELAYER_METHOD_EXSCRIPT',1)} See example
```

Creates a watermark layer within the layout object.

### Syntax

[objectreference].createlayer

### Parameters

### Return value

The return value for this method will always be -1.

### Usage

When you call the CreateLayer method, a watermark layer is created for the specified layout object.

Because the return value of CreateLayer is always -1, you will not be able to tell if a watermark layer is created successfully when you call the method. Use the LayerName property of a layout object to determine if its layer property actually contains a watermark layer object.

## **Word Pro: DeleteBag method**

{button ,AL('H\_BAG\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETEBAG\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: DeleteltemByPosition method**

{button ,AL(`H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DELETEITEMBYPOSITION\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: EndPrinting method**

{button ,AL(`H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ENDPRINTING\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



## **Word Pro: EnvelopeBarCode method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ENVELOPEBARCODE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

**Word Pro: GetAspectMetafilePict method**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETASPECTMETAFILEPICT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: GetListName method**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLISTNAME\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: GetPageRange method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETPAGERANGE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: GetParagraphNumber method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GETPARAGRAPHNUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: GetString method**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSTRING\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

### **Word Pro: GoToClickHere method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GOTOCCLICKHERE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

#### **Syntax**

Unknown

#### **Parameters**

Unknown

#### **Return value**

Unknown

#### **Usage**

## **Word Pro: HasProperty method**

{button ,AL('H\_CHARACTERSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HASPROPERTY\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



## Word Pro: ImportWatermarkGraphic method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERGROUPLAYOUT_CLASS;H_SUPERLAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERLAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_IMPORTWATERMARKGRAPHIC_METHOD_EXSCRIPT',1)} See example
```

Imports a graphic into the watermark layer of a layout object.

### Syntax

```
[objectreference].ImportWatermarkGraphic(FilePath, FileFormat, Link, ScratchOutFrame, [FrameStyle], [ShowPipesWhenDone])
```

### Parameters

#### *FilePath*

A String expression which specifies the directory path and name of the file which is the source of the imported graphic.

#### *FileFormat*

A String expression which specifies the file format for the graphic you are importing. The string expression for each file format is unique and registered with Microsoft Windows 95. The values listed in the table below were valid at the time of publication.

#### **If you are importing this type of graphic: Use this value for the FileFormat parameter:**

WordProWMF	.wmf
WordProEPS	.eps
WordProPaint	.bmp
WordProImage	.tif
WordProDraw	.sdw
Boxes	.box
WordProEqn	.tex
WordProPCX	.pcx
WordProGIF	.gif
CGM	.cgm
HPGL	.plt
PCD	.pcd
DRW	.drw
CDR	.cdr
PIC	.pic
ImpWPG	.wpg
ExpSDW2WPG	.sdw
ExpWMF2WPG	.wmf
ExpBMP2WPG	.bmp
ImpWP2	.wp2
ExpSDW2WP2	.sdw
ExpWMF2WP2	.wmf
ExpBMP2WP2	.bmp
ImpJPG	.jpg
ExpJPG	.bmp

### *Link*

An Integer value of -1 or 0 indicating whether the imported graphic will be receive updates from the original (-1) or will remain independent of the original (0). You can use the LotusScript constants of True (-1) and False (0) as the value for this parameter.

### *ScratchOutFrame*

An Integer value which indicates whether you want to draw the new graphic frame by hand or let Word Pro draw the frame based on a frame style. If you want to draw the frame yourself, use the value of True (-1) for this parameter. If you want Word Pro to draw the frame based on an existing style, use a value of False (0) for this parameter.

### *FrameStyle*

A String expression which specifies the frame style you want to use for the imported graphic's frame. Optional parameter. If the imported graphic is an equation and you do not specify a frame style, Word Pro will use the default equation frame style. All other imported graphics will be placed in the default GraphicOle frame style, unless you specify another frame style using this parameter.

### *ShowPipesWhenDone*

An Integer value which determines what object will have the focus after the method is called. This applies only to graphics imported into watermark layers of frame layout objects. If you want the frame to be selected after the method is called, use a value of True in this parameter. If you want the content of the frame to have the focus, use a value of False in this parameter. Optional parameter. Default is True.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: InternetExtraFile method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INTERNETEXTRAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: IsCaseExact method**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCASEEXACT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

**Word Pro: LeastRecentVersion method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LEASTRECENTVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

### **Word Pro: LinkContainers method**

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASSES;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_LINKCONTAINERS_METHOD_EXSCRIPT',1)} See example
```

Links the contents of selected container objects.

### **Syntax**

```
[objectreference].LinkContainers()
```

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing Frame - Link Frame Contents. The LinkContainers method is only valid when being called from frame container objects.

## **Word Pro: NewItemByPosition method**

{button ,AL(^H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEWITEMBYPOSITION\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleAlign method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLEALIGN\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



## **Word Pro: NextCycleAttribute method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLEATTRIBUTE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleBullet method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLEBULLET\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleFontSize method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLEFONTSIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleFont method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLEFONT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleIndent method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLEINDENT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleNumber method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLENUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: NextCycleStyle method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTCYCLESTYLE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: OpenDocumentFromNotes method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_OPENDOCUMENTFROMNOTES\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



## **Word Pro: RemoveIndexEntry method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REMOVEINDEXENTRY\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: RemoveList method**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVELIST\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: RemoveTOCEntry method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REMOVETOCENTRY\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: Repaint method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPAINT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: ReplaceContents method**

{button ,AL(^H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REPLACECONTENTS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: ResetNumberOpts method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RESETNUMBEROPTS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: ResetPrinting method**

{button ,AL(^H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RESETPRINTING\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: RestorePreviousView method**

{button ,AL(`H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_RESTOREPREVIOUSVIEW\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



## **Word Pro: RestoreWindowFromCleanScreen method**

{button ,AL(^H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RESTOREWINDOWFROMCLEANSCREEN\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SanityCheck method**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SANITYCHECK\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SaveMergeDataFile method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SAVEMERGEDATAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## Word Pro: SetAllMargins method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SETALLMARGINS_METHOD_EXSCRIPT',1)} See example
```

Allows you to simultaneously set all margin and external margin values for a Layout object.

### Syntax

[objectreference].SetAllMargins(flag, [l], [r], [t], [b], [lx], [rv], [tx], [bx])

### Parameters

#### *flag*

The value of this Variant parameter must be one of the hexadecimal values below or a combination of the values.

<u>Value</u>	<u>Effect</u>
&H01	Allows you to set the left margin value.
&H02	Allows you to set the right margin value.
&H04	Allows you to set the top margin value.
&H08	Allows you to set the bottom margin value.
&H10	Allows you to set the left external margin value.
&H20	Allows you to set the right external margin value.
&H40	Allows you to set the top external margin value.
&H80	Allows you to set the bottom external margin value.

*l*

Data type is Long. Optional parameter which specifies the left margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*r*

Data type is Long. Optional parameter which specifies the right margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*t*

Data type is Long. Optional parameter which specifies the top margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*b*

Data type is Long. Optional parameter which specifies the bottom margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*lx*

Data type is Long. Optional parameter which specifies the left external margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*rx*

Data type is Long. Optional parameter which specifies the right external margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*tx*

Data type is Long. Optional parameter which specifies the top external margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*bx*

Data type is Long. Optional parameter which specifies the bottom external margin value. The unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You must provide values for any margins which are selected by the flag parameter. For example, if you set the flag parameter to &H01, then you must provide a value in the left margin parameter or the method will fail. However, since the l parameter is optional, no run-time error will occur.

## **Word Pro: SetCustomNumber method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETCUSTOMNUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SetIndexInfo method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETINDEXINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SetJapanIndexInfo method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETJAPANINDEXINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**



**Word Pro: SetNoFields method**

{button ,AL(^H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETNOFIELDS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: SetNumberingLevelInfo method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETNUMBERINGLEVELINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SetTOCLevelContent method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETTOCLEVELCONTENT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SetTOCLevelPageInfo method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SETTOCLEVELPAGEINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

**Word Pro: SpecialView method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPECIALVIEW\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: SplitDivision method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPLITDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: SplitWindow method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPLITWINDOW\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## Word Pro: UnLinkContainers method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCO  
LSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_UNLINKCONTAINERS_METHOD_EXSCRIPT',1)} See example
```

Unlinks the contents of the selected frame container objects.

### Syntax

[objectreference].UnLinkContainers()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Equivalent to choosing Frame - Unlink. The UnlinkContainers method is only valid when called from frame container objects.



**Word Pro: Unlink method**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UNLINK\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

**Syntax**

Unknown

**Parameters**

Unknown

**Return value**

Unknown

**Usage**

## **Word Pro: UpdatePowerFieldsOnNew method**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEPOWERFIELDSONNEW\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: UpdatePrinterBins method**

{button ,AL(^H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEPRINTERBINS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: UpdateTabs method**

{button ,AL('H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATETABS\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: UpdateUI method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEUI\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

## **Word Pro: WordCount method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_WORDCOUNT\_METHOD\_EXSCRIPT',1)} [See example](#)

This language element is not yet defined.

### **Syntax**

Unknown

### **Parameters**

Unknown

### **Return value**

Unknown

### **Usage**

'Example: Hwnd property

'This example script has not yet been created.

```
'Example: GetProfileString method
' This example prints the last 6 open files names to the Lotus Script Output
' panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Key = "LastOpen1"
Section = "WordProUser"
Defstring0 = ""
IniFileType0 = $LwpIniUserPrefs
WhichIniLocation0 = ""
IniName0 = "lwpuser.ini"
For x = 1 To 6
    Key = "LastOpen" & x
    MsgBox .GetProfileString(Section, Key, Defstring0, IniFileType0,
WhichIniLocation0,IniName0)
Next
```



'Example: GetSource method

'This example script has not yet been created.

'Example: GetSpellStatus method

'This example script has not yet been created.

'Example: GetSpellUserDictStatus method

'This example script has not yet been created.

```
'Example: GetStandardButtonId method
' This example simulates clicking on the font status bar button.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim StatBar As StatusBar
Dim ButtonId as Integer

Set StatBar = .ApplicationWindow.StatusBar
Forall Button In StatBar.StatusBarButtons
    ButtonId = Button.GetButtonId
    If (ButtonId = StatBar.GetStandardButtonId($LwpStandButtFontButton)) Then
        Button.SimulateButtonClick
    End If
End Forall
```

'Example: GetStatus property

'This example script has not yet been created.

'Example: GetStorageProtectionType method

'This example script has not yet been created.

```
'Example: GetText method
' This example prints the current word, sentence and paragraph as well as
' each word of the current paragraph to the Script Editor Output panel.
' RUNTIME DEPENDENCIES: You must have a document open with the cursor
' positioned on a line with text for this script to work.

Print .Text.GetText($LwpGetObjectWord, False)
Print .Text.GetText($LwpGetObjectSentence, False)
Print .Text.GetText($LwpGetObjectParagraph, False)
While Not .Text.AtEndOfParagraph
    Print .Text.GetText($LwpGetObjectWord,True)
Wend
```

'Example: GetTOCProperties method

'This example script has not yet been created.



'Example: GetTopicName property

'This example script has not yet been created.

'Example: GetUndoWhatDesc property

'This example script has not yet been created.

'Example: GetUniqueName method

' This example prints a unique name for a bookmark to the Lotus Script Output panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .Division.BookMarkManager.GetUniqueName

'Example: GetUserClassNameFull method

'This example script has not yet been created.

'Example: GetUserClassNameShort method

'This example script has not yet been created.

'Example: GetValue property

'This example script has not yet been created.

```
'Example: GetWordMisspelled method
' This example inserts a sentence of text into the current document. Each word
' starting with the first is then selected and checked for misspelling.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "This is a Sentence of text."
.Text.MoveToStart $LwpLocationTypeLine

Do
    .SelectWord
    Stat = .Text.GetWordMisspelled
    If Stat = False Then
        MsgBox "Misspelled " & .Text.GetText($LwpGetObjectWord, False)
    End If
    NextWord = .Text.Forward ($LwpNavigateObjectTypeWord, 1)
Loop Until (.Text.AtBeginningOfParagraph = True) Or (NextWord = False)
```

'Example: Get method

'This example script has not yet been created.



'Example: Glossarize method

'This example script has not yet been created.

'Example: GlossaryDataFileName property

'This example script has not yet been created.

'Example: GlossaryDataFiles property

'This example script has not yet been created.

'Example: GlossaryDataPaths property  
'This example script has not yet been created.

'Example: GlossaryInsert method

'This example script has not yet been created.

```
'Example: GlossaryOpen method
' This example opens the glossary file named GLOSSARY.GLS and prints all of
' the glossary entries to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = False
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = True
.GlossaryOpen "GLOSSARY.GLS", "Lotus Word Pro"
.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = True
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = False
Forall Gloss In .Division.Foundry.Glossarys
    Count% = Gloss.NumRows
    For Item% = 1 To (Count% - 1)
        GlossItem$ = Gloss.EnumerateTerm(Item%)
        GlossText$ = Gloss.ExtractText(GlossItem$)
        Print GlossText$      Next
End Forall
.Documents("GLOSSARY.GLS").Close
```

'Example: GlossaryPath property

'This example script has not yet been created.

'Example: Glossary's property

'This example script has not yet been created.



```
'Example: GoToBookmark method
' This example first creates a bookmark in the current document, then moves
' the cursor out of the bookmark by splitting the paragraph, and then displays
' a message.
' When you click OK, the cursor goes back to the bookmark.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim MarkerName as String
MarkerName = .Mark($LwpMarkerTypeBookmark)
.Division.BookmarkManager.AddBookmark "ExampleBookmark", MarkerName
.Text.SplitParagraph
MessageBox "Click OK to go to the bookmark.",MB_OK,"Example Script"
```

```
.GoToBookmark("ExampleBookmark")
```

'Example: GoToContainer method

' This example selects the first frame in the current division.

' RUNTIME DEPENDENCIES: You must have a document open which contains a frame  
' for this script to work.

.Page.GoToContainer \$LwpGoToLocationFrame

'Example: GoToIndexLastItemAccessed property  
'This example script has not yet been created.

'Example: GoToItemName property

'This example script has not yet been created.

```
'Example: GoToLayout method
' This example creates a table with 5 rows and 5 columns into the current
' document. The cursor is then positioned to row 1, col 1.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MyTable As Table
.CreateTable False, "Default Table", 5,5
Set MyTable = .Table

MyTable.CellLayout(1,1).GotoLayout
```

'Example: GotoNextParallelColumn method

' This example creates a parallel column and navigate to the second column.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.GotoNextParallelColumn

'Example: GoToObject method

' This example moves the insertion point to document header.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.GoToObject ("Header", 1)

'Example: GoToPageLoadingBackground property  
'This example script has not yet been created.



'Example: GoToPageNumber property

'This example script has not yet been created.

'Example: GoToPage method

' This example goes to the second page of the current document.

' RUNTIME DEPENDENCIES: You must have a document open with two or more pages  
' for this script to work.

.GoToPage 2

```
'Example: GoToSection method
' This example inserts two sections and then goes to each one while selecting
' the sections' contents.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
Forall Section In .Division.Foundry.Sections
    Section.GoToSection
    .SelectSection
End Forall
```

'Example: GoToSelection property

'This example script has not yet been created.

'Example: GoToTableCell method

' This example creates a table with 5 rows and 5 columns into the current document. The cursor is then advanced to

' the next cell.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MyTable As Table

.CreateTable False, "Default Table", 5,5

Set MyTable = .Table

MyTable.GoToTableCell \$LwpCellObjectType,True

```
'Example: GoTo method
' This examples illustrates how to validate the contents of a Click Here
' Block. After the Click Here Block loses focus, the Exitclickhere event is
' called. If more than 15 characters have been entered, a message box is
' displayed and the Click Here Block is then re-selected.
' RUNTIME DEPENDENCIES: You must have a document open and a Click Here Block
' for this script to work.
```

```
Sub Exitclickhere(Source As Clickhere, Clickherename As String)
    Dim Temp as String
    Const MaximumLen = 15
    Temp = Source.GetMarkedText
    If Len(Temp) > MaximumLen Then
        Messagebox "Please Enter less than 15 characters."
        Source.Goto(True)
    End If
End Sub
```

'Example: GrammarFormalityLevel property

'This example script has not yet been created.

'Example: GrammarOptions property

'This example script has not yet been created.



'Example: GrammarProofLevel property  
'This example script has not yet been created.

'Example: GrammarSetName property

'This example script has not yet been created.

'Example: GraphicExports property

'This example script has not yet been created.

'Example: GraphicImports property

'This example script has not yet been created.

'Example: GraphicOleObject property  
'This example script has not yet been created.

'Example: Graphics property

'This example script has not yet been created.

'Example: Graphic property

'This example script has not yet been created.

'Example: Green property

'This example script has not yet been created.



'Example: Greeting property

'This example script has not yet been created.

'Example: GridDistance property

'This example script has not yet been created.

'Example: GridType property

'This example script has not yet been created.

'Example: GroupDivision method

'This example script has not yet been created.

'Example: Groups property

'This example script has not yet been created.

'Example: Gutter property

'This example script has not yet been created.

'Example: HandsOffStorage method

'This example script has not yet been created.

'Example: Hang property

'This example script has not yet been created.



'Example: HasContents property

'This example script has not yet been created.

'Example: HasFocus property

'This example script has not yet been created.

'Example: HasLocalTabs property

'This example script has not yet been created.

```
'Example: HasNamedProperty method
' This example creates a named property, 'ExampleProp' on the active document
' and assigns it a value. The value is then printed to the Lotus Script Output
' panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim stat as Integer
```

```
stat = .ActiveDocument.HasNamedProperty("ExampleProp")
If stat = False Then
    .ActiveDocument.SetNamedProperty "ExampleProp", "Here is some data."
End If
Print .ActiveDocument.GetNamedProperty ("ExampleProp")
```

'Example: HasTabs property

'This example script has not yet been created.

'Example: Headers property

'This example script has not yet been created.

'Example: Header property

'This example script has not yet been created.

'Example: Heading property

'This example script has not yet been created.



```
'Example: Height property
Dim CR As String*1
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager

IcnPallet = "Comment Tools"
CR = Chr(10)
Set IcnMgr = .ApplicationWindow.IconBarManager

With IcnMgr.IconBars(IcnPallet)
    MsgStr = "Height = " & .Height & CR
    MsgStr = MsgStr & "IconBarPositionState = " & .IconBarPositionState & CR
    MsgStr = MsgStr & "PositionType = " & .PositionType & CR
    MsgStr = MsgStr & "ScreenPositionX = " & .ScreenPositionX & CR
    MsgStr = MsgStr & "ScreenPositionY = " & .ScreenPositionY
    MessageBox MsgStr, 64, "Script Example - " & .Name
End With
```

'Example: HelpText property

'This example script has not yet been created.

'Example: Help method

' This example display the Word Pro 97 Lotus Script Object Model help.

.Help "C:\Lotus\WordPro\Wp0n71en.hlp"

'Example: HiddenMode property

'This example script has not yet been created.

'Example: HideFormula property

'This example script has not yet been created.

'Example: HideIconBars method

' This example hides all SmartIcon bars the redisplays them after the message  
' box is closed.

**.ApplicationWindow.IconBarManager.HideIconBars**

MessageBox "Click OK to show SmartIcon bars.",MB\_OK,"Example Script"

.ApplicationWindow.IconBarManager.ShowIconBars

```
'Example: HideIconBar method
' This example displays and then hides the "Comment Tools" SmartIcon bar
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager

IcnPallet = "Comment Tools"
Set IcnMgr = .ApplicationWindow.IconBarManager

' Set the context and show the bar
IcnMgr.IconBars(IcnPallet).ShowInContext = True
IcnMgr.IconBars(IcnPallet).Show

MsgStr = "|" & IcnPallet
MsgStr = MsgStr & "|" pallet is now displayed, click OK to hide this pallet"|
MessageBox MsgStr, 48, "Example Script"

' Reset the context and hide the bar.
IcnMgr.IconBars(IcnPallet).ShowInContext = False

IcnMgr.IconBars(IcnPallet).HideIconBar
```

'Example: HideOutlineLevels property  
'This example script has not yet been created.



```
'Example: HideStatusBar method
' This example hides the status bar if it is visible, and shows it if it is
' hidden.
If .ApplicationWindow.StatusBar.Visible = True Then
    .ApplicationWindow.StatusBar.HideStatusBar
Else
    .ApplicationWindow.StatusBar.ShowStatusBar
End If
```

'Example: Hide method

'This example script has not yet been created.

'Example: HighLightMode property

'This example script has not yet been created.

'Example: HighlightToggle method

' This example toggles the Review & Comment tools highlighter on or off.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.HighlightToggle

'Example: HiLiteColor property

'This example script has not yet been created.

'Example: Hit method

'This example script has not yet been created.

'Example: HorizontalSplitWindow property  
'This example script has not yet been created.

'Example: HorizScrollBarVisible property  
'This example script has not yet been created.



'Example: HorzRuler property

'This example script has not yet been created.

```
'Example: HourGlass method
' This example displays an hourglass mouse pointer while counting to 50000
' after which the original pointer is restored.
' RUNTIME DEPENDENCIES: This method is only available in the
' 32-bit (WINDOWS95) version of Word Pro.

.HourGlass(True)
For i = 1 to 50000
Next
.HourGlass(False)
```

'Example: HyphenateLastWordInColumnOrPage property  
'This example script has not yet been created.

'Example: HyphenateLastWordInPara property  
'This example script has not yet been created.

'Example: HyphenationOptions property

'This example script has not yet been created.

'Example: HyphZoneAfter property

'This example script has not yet been created.

'Example: HyphZoneBefore property

'This example script has not yet been created.

'Example: IconBarManager property

'This example script has not yet been created.



```
'Example: IconBarPositionState property
Dim CR As String*1
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager

IcnPallet = "Comment Tools"
CR = Chr(10)
Set IcnMgr = .ApplicationWindow.IconBarManager

With IcnMgr.IconBars(IcnPallet)
    MsgStr = "Height = " & .Height & CR
    MsgStr = MsgStr & "IconBarPositionState = " & .IconBarPositionState & CR
    MsgStr = MsgStr & "PositionType = " & .PositionType & CR
    MsgStr = MsgStr & "ScreenPositionX = " & .ScreenPositionX & CR
    MsgStr = MsgStr & "ScreenPositionY = " & .ScreenPositionY
    MessageBox MsgStr, 64, "Script Example - " & .Name
End With
```

'Example: IconBarSets property

'This example script has not yet been created.

'Example: IconBars property

'This example script has not yet been created.

'Example: IconHelpText property

'This example script has not yet been created.

'Example: IconPaths property

'This example script has not yet been created.

'Example: IconPath property

'This example script has not yet been created.

'Example: IconSetName property

'This example script has not yet been created.

'Example: IconShowingBeforeCleanScreen property  
'This example script has not yet been created.



'Example: Hwnd property

'This example script has not yet been created.

**Word Pro: Hwnd property**

{button ,AL(^H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_APPLICATIONWINDOW\_DOCWINDOW\_HWND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read) The Windows handle for the current application window.

**Data Type**

Long

**Syntax**

hwndvalue = [objectreference].Hwnd

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

Use this property to send script calls to the application window. For example, you can use this property to send .DLL files to the application window.

## **Word Pro: ExternalName property**

{button ,AL('H\_DIVISIONINFO\_CLASS;H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXTERNALNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The user assigned name that displays on a section or division tab.

### **Data Type**

[String](#)

### **Syntax**

externalnamevalue = [objectreference].ExternalName

[objectreference].ExternalName = externalnamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

You cannot change the name of a section or division tab by using this LotusScript property. You can change the section or division tab name by double clicking the section or division tab, typing a name, and clicking outside the tab or pressing ENTER.

### **Word Pro: ExternalType property**

{button ,AL('H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXTERNALTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

externaltypevalue = [objectreference].ExternalType

[objectreference].ExternalType = externaltypevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: FaceStyleName property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FACESTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the style of

**Data Type**

String

**Syntax**

[objectreference].FaceStyleName()

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: FastFormatType property**

{button ,AL('H\_FORMATPREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FASTFORMATTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

FastFormatType

### **Syntax**

fastformattypevalue = [objectreference].FastFormatType

[objectreference].FastFormatType = fastformattypevalue

### **Legal values**

\$LwpFastFormatTypeHighlight (253)

\$LwpFastFormatTypeNone (250)

\$LwpFastFormatTypeStyle (252)

\$LwpFastFormatTypeText (251)

### **Usage**

## **Word Pro: FaxNumber property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FAXNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

faxnumbervalue = [objectreference].FaxNumber

[objectreference].FaxNumber = faxnumbervalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: FieldDelimiterText property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIELDDDELIMITERTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to specify the string character used to find a sort field in a document.

**Data Type**

[String](#)

**Syntax**

fielddelimitertextvalue = [objectreference].FieldDelimiterText

[objectreference].FieldDelimiterText = fielddelimitertextvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to choosing Text - Sort, selecting "Text" and entering a string character in the box next to it.



## **Word Pro: FieldDelimiter property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIELDDELIMITER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to indicate whether a string character or a tab is used to find sort fields in a document.

### **Data Type**

Variant (Enumerated)

DelimiterType

### **Syntax**

fielddelimitervalue = [objectreference].FieldDelimiter

[objectreference].FieldDelimiter = fielddelimitervalue

### **Legal values**

\$LwpDelimiterTypeTabdelimited (180) Setting this value indicates that a tab is being used to find sort fields in a document.

\$LwpDelimiterTypeTextdelimited (181) Setting this value indicates that a string character is being used to find sort fields in a document.

### **Usage**

Equivalent to choosing Text - Sort and selecting "Tab" or "Text" under "Field delimiter."

## **Word Pro: FieldNumber property**

{button ,AL('H\_SORTKEY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIELDNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies which field or column that you want to sort. A field is separated by tabs or a character that is not used in the text. If the data is in a table, the columns are the fields.

### **Data Type**

[Integer](#)

### **Syntax**

fieldnumbervalue = [objectreference].FieldNumber

[objectreference].FieldNumber = fieldnumbervalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Text - Sort and selecting a number from the "Field/col." box in either Level 1 "First sort by," Level 2 "Then by," or Level 3 "Then by:"

### **Word Pro: FieldType property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIELDTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the Powerfield at the insertion point.

### **Data Type**

String

### **Syntax**

fieldtypevariablevalue = [objectreference].FieldType

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: FileName property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

filenamevalue = [objectreference].FileName

[objectreference].FileName = filenamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: FileProtectionType property

{button ,AL(^H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FILEPROTECTIONTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

[DocControl]

Restricts access to a specific file.

[FileProtection]

Stores the type of protection assigned to a document.

### Data Type

Variant (Enumerated)

### Syntax

fileprotectiontypevalue = [objectreference].FileProtectionType

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpFileProtectTypeEditors (261)	Only those listed as editors on the Editing Rights panel can open the document.
\$LwpFileProtectTypeNone (260)	Anyone with access to the folder where the document is stored can open the document.
\$LwpFileProtectTypeOrigAuthor (262)	Only the person who created the document can open it.
\$LwpFileProtectTypePassword (263)	Anyone who knows the document password can open the document.
\$LwpFileProtectTypeReserved (264)	Word Pro uses this type internally whenever the file protection type stored within the document is not recognized. For example, an existing file protection type is not recognized in a corrupted file or in a file created with a later version of Word Pro that added new verification types. Because Word Pro uses this property type internally, a user would never assign this property type.

### Usage

[DocControl]

Equivalent to choosing File - TeamSecurity and selecting one of the options in the "Who can open (access) this file" section on the Access panel.

[FileProtection]

Use this property to track the type of protection assigned to a document. If a TeamSecurity option is selected, this property is equivalent to options available in the "Who can access this document field" on the Access panel of the TeamSecurity dialog box.

## **Word Pro: FilesToCompare property**

{button ,AL('H\_REVIEWVERSIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILESTOCOMPARE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

### **Data Type**

[String](#)

### **Syntax**

[objectreference].FilesToCompare = filestocomparevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: FileType property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILETYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

filetypevalue = [objectreference].FileType

[objectreference].FileType = filetypevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: FillStyle property

{button ,AL('H\_TABLEFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILLSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to specify whether or not the background color and pattern fills the entire table, every other column, or every other row. You can specify more than one type of fill style.

### Data Type

Variant (Enumerated)

TableFillStyle

### Syntax

fillstylevalue = [objectreference].FillStyle

[objectreference].FillStyle = fillstylevalue

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpTableFillStyleAll (1870)	Allows you to set a fill style for an entire table object
\$LwpTableFillStyleEveryothercol (1872)	Allows you to set a fill style for every other column in a table object.
\$LwpTableFillStyleEveryotherrow (1871)	Allows you to set a fill style for every other row in a table object.
\$LwpTableFillStyleMixed (1873)	Allows you to set more than one type of fill style for a table cell object.
\$LwpTableFillStyleNone (1874)	Indicates that no fill style is set for a table cell object.

### Usage

You can use this property to return the current fill style value stored in the Background property of a specific table object. You can then use this value to specify whether or not the background color and pattern fills the entire table, every other column, or every other row. You can also specify more than one type of fill style. After you set fill style property for a specific table object, it is stored in the Background property of that table object.

For more information, see Background property.



**Word Pro: FilterName property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FILTERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The field names provided by an external data file.

**Data Type**

String

**Syntax**

filternamevalue = [objectreference].FilterName

[objectreference].FilterName = filternamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: FindExactCase property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDEACTCASE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables a user to find the exact case of a word or phrase in Find & Replace.

### **Data Type**

[Integer](#)

### **Syntax**

findexactcasevalue = [objectreference].FindExactCase

[objectreference].FindExactCase = findexactcasevalue

### **Legal values**

Always contains an instance of the Text class. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property to check and manipulate the setting for finding the exact case of a word or phrase in Find & Replace. If True, finds the word that matches the user setting. Equivalent to choosing Edit - Find & Replace Text, clicking Options, and selecting "Match case" in the "Find options" section.

## **Word Pro: FindForwardDirection property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDFORWARDIRECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to set a forward or backward direction for a search in Find & Replace.

### **Data Type**

[Integer](#)

### **Syntax**

findforwarddirectionvalue = [objectreference].FindForwardDirection

[objectreference].FindForwardDirection = findforwarddirectionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

You should set the forward or backward direction before using Find & Replace. If True, sets Find & Replace to go forward in the document. If False, sets Find & Replace to go backward in the document.

## **Word Pro: FindString property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDSTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to type what to search for in Find & Replace.

### **Data Type**

[String](#)

### **Syntax**

findstringvalue = [objectreference].FindString

[objectreference].FindString = findstringvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text and typing a text or character string in the "Find" box.

**Word Pro: FindStyleName property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to find a paragraph style in Find & Replace.

**Data Type**

[String](#)

**Syntax**

findstylevalue = [objectreference].FindStyleName

[objectreference].FindStyleName = findstylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to choosing Edit - Find & Replace Text, typing the character, ^p, in the "Find" box, clicking Options, clicking the Font button in the "Find options" section, and choosing a paragraph style in the "Style" list box.

**Word Pro: FindWithProperties property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDWITHPROPERTIES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to find font properties in Find & Replace.

**Data Type**

[Integer](#)

**Syntax**

findwithpropertiesvalue = [objectreference].FindWithProperties

[objectreference].FindWithProperties = findwithpropertiesvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Use this property to find font properties in Find & Replace. If True, finds the font properties that match the user setting. Equivalent to choosing Edit - Find & Replace Text, clicking Options, selecting "Include properties," clicking the Font button in the "Find options" section, and selecting properties.

**Word Pro: FirstChild property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIRSTCHILD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

firstchildvalue = [objectreference].FirstChild

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: FirstName property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIRSTNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

firstnamevalue = [objectreference].FirstName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: FirstPage property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIRSTPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

firstpagevalue = [objectreference].FirstPage

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: First property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIRST\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the first ClickHere block in the division (uses Tab order).

**Data Type**

Long

**Syntax**

firstvalue = [objectreference].First

[objectreference].First = firstvalue

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

## **Word Pro: FitType property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FITTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer (Enumerated Bitmask)

FitType

### **Syntax**

fittypevalue = [objectreference].FitType

[objectreference].FitType = fittypevalue

### **Legal values**

LwpFitHorz (&H2)

LwpFitHorzmargin (&H4)

LwpFitVertical (&H1)

### **Usage**

**Word Pro: FixAcronymns property**

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIXACRONYMNS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

fixacronymnsvalue = [objectreference].FixAcronymns

[objectreference].FixAcronymns = fixacronymnsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: FixBullets property**

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIXBULLETS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

fixbulletsvalue = [objectreference].FixBullets

[objectreference].FixBullets = fixbulletsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: FixMargins property**

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIXMARGINS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

fixmarginsvalue = [objectreference].FixMargins

[objectreference].FixMargins = fixmarginsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: FontName property**

{button ,AL('H\_FONT\_CLASS;H\_FONTMETRICS\_CLASS;H\_PREFERENCES\_CLASS;H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

fontnamevalue = [objectreference].FontName

[objectreference].FontName = fontnamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: FontSize property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Points](#)

### **Syntax**

fontsizevalue = [objectreference].FontSize

[objectreference].FontSize = fontsizevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: FontStyleName property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

fontstylevalue = [objectreference].FontStyleName

[objectreference].FontStyleName = fontstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: FontUnitName property

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTUNITNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores the name of the unit of measurement used for displaying and setting font size.

### Data Type

[String](#)

### Syntax

fontunitnamevalue = [objectreference].FontUnitName

### Legal values

### Usage

To actually work with the unit of measurement for fonts, use the FontUnits property. Although the FontUnits property is independent of this property, its value will correspond with the value of this property. For example, if the value of FontUnits is \$LtsScaleModeCentimeter (1056964840), then the value of this property is "centimeters."

## **Word Pro: FontUnits property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FONTUNITS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the unit of measurement used for displaying and setting font size.

### **Data Type**

Variant (Enumerated)

ScaleMode

### **Syntax**

fontunitsvalue = [objectreference].FontUnits

[objectreference].FontUnits = fontunitsvalue

### **Legal values**

\$LtsScaleModeCentimeter (1056964840)

\$LtsScaleModeInch (1056964838)

\$LtsScaleModePoint (1056964837)

\$LwpScaleModePica (1728)

### **Usage**

Equivalent to the FontUnitName property. Default is "points." Changes are only valid during the current session.

## **Word Pro: FormatCheckLevel property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORMATCHECKLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

formatchecklevelvalue = [objectreference].FormatCheckLevel

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: FormatCheckRule property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORMATCHECKRULE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

formatcheckrulevalue = [objectreference].FormatCheckRule

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: FormatCheckSuggestion property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORMATCHECKSUGGESTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

formatchecksuggestionvalue = [objectreference].FormatCheckSuggestion

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: FormatType property

{button ,AL('H\_NUMERICFORMAT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORMATTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Variant (Enumerated)

NumberFormat

### Syntax

formattypevalue = [objectreference].FormatType

[objectreference].FormatType = formattypevalue

### Legal values

\$LtsNumberFormatComma (1056964626)

\$LtsNumberFormatFixed (1056964623)

\$LtsNumberFormatGeneral (1056964622)

\$LtsNumberFormatPercent (1056964627)

\$LtsNumberFormatScientific (1056964624)

\$LwpNumberFormatArgentineanpeso (1532)

\$LwpNumberFormatAustraliandollar (1533)

\$LwpNumberFormatAustrianschilling (1534)

\$LwpNumberFormatBelgianfranc (1535)

\$LwpNumberFormatBraziliancruzeiro (1536)

\$LwpNumberFormatBritishpound (1537)

\$LwpNumberFormatCanadiandollar (1538)

\$LwpNumberFormatChineseyuan (1539)

\$LwpNumberFormatCzechkoruna (1540)

\$LwpNumberFormatDanishkrone (1541)

\$LwpNumberFormatDefault (1576)

\$LwpNumberFormatEcu (1542)

\$LwpNumberFormatFinnishmarkka (1543)

\$LwpNumberFormatFrenchfranc (1544)

\$LwpNumberFormatGermanmark (1545)

\$LwpNumberFormatGreekdrachma (1546)

\$LwpNumberFormatHongkongdollar (1547)

\$LwpNumberFormatHungarianforint (1548)

\$LwpNumberFormatIndianrupee (1549)

\$LwpNumberFormatIndonesianrupiah (1550)

\$LwpNumberFormatIrishpunt (1551)

\$LwpNumberFormatItalianlira (1552)

\$LwpNumberFormatJapaneseyen (1553)

\$LwpNumberFormatLabel (1582)

\$LwpNumberFormatLuxembourgfranc (1554)

\$LwpNumberFormatMalaysianringgit (1555)

\$LwpNumberFormatMexicanpeso (1556)

\$LwpNumberFormatNetherlandsguilder (1557)

\$LwpNumberFormatNewzealanddollar (1558)

\$LwpNumberFormatNone (1531)  
\$LwpNumberFormatNorwegiankroner (1559)  
\$LwpNumberFormatOthercurrency (1575)  
\$LwpNumberFormatPolishzloty (1560)  
\$LwpNumberFormatPortugueseescudo (1561)  
\$LwpNumberFormatRomanianlei (1562)  
\$LwpNumberFormatRussianruble (1563)  
\$LwpNumberFormatSingaporedollar (1564)  
\$LwpNumberFormatSlovakiankoruna (1565)  
\$LwpNumberFormatSloveniantholar (1566)  
\$LwpNumberFormatSouthafricanrand (1567)  
\$LwpNumberFormatSouthkoreanwon (1568)  
\$LwpNumberFormatSpanishpeseta (1569)  
\$LwpNumberFormatSwedishkrona (1570)  
\$LwpNumberFormatSwissfranc (1571)  
\$LwpNumberFormatTaiwandollar (1572)  
\$LwpNumberFormatThaibaht (1573)  
\$LwpNumberFormatUsdollar (1574)  
\$LwpNumberLtsNumberFormatComma (1579)  
\$LwpNumberLtsNumberFormatFixed (1578)  
\$LwpNumberLtsNumberFormatGeneral (1577)  
\$LwpNumberLtsNumberFormatPercent (1580)  
\$LwpNumberLtsNumberFormatScientific (1581)

## Usage



**Word Pro: Formula property**

{button ,AL(^H\_CELLLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_FORMULA\_CLASS;H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FORMULA\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to set or return the formula in a table cell.

**Data Type**

String

**Syntax**

formulavalue = [objectreference].Formula

[objectreference].Formula = formulavalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

If there is no formula in the specified cell, the Formula property will contain an empty string.

**Word Pro: FrameStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FRAMESTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

framestylevalue = [objectreference].FrameStyleName

[objectreference].FrameStyleName = framestylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: FullName property**

{button ,AL(^H\_APPLICATION\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FULLNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The full name and path of the currently active Word Pro session or document.

### **Data Type**

[String](#)

### **Syntax**

fullnamevalue = [objectreference].FullName

### **Legal values**

The value of this property cannot be set by a script.

### **Usage**

Use this property to retrieve the full name and path of the application or a document for OLE automation or any other use. Call this property from the WPAApplication object to get the full directory path and executable name of the application which is running the currently active session of Word Pro. Call this property from a TextDocument object to get the full directory path and name of that document.

## **Word Pro: GapBetweenPanels property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GAPBETWEENPANELS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

gapbetweenpanelsvalue = [objectreference].GapBetweenPanels

[objectreference].GapBetweenPanels = gapbetweenpanelsvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

**Word Pro: GetAfidHelpFileName property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETAFIDHELPPFILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

getafidhelpfilenamevalue = [objectreference].GetAfidHelpFileName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: GetAfidHelpInfo property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETAFIDHELPIFO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

getafidhelpinfovalue = [objectreference].GetAfidHelpInfo

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: GetButtonId property**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETBUTTONID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the ID of the status bar button.

**Data Type**

Long

**Syntax**

getbuttonidvalue = [objectreference].GetButtonId

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

This property lets you determine the ID for the selected button.

## Word Pro: GetButtonType property

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETBUTTONTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the type of status bar button (text, graphic, or popup). The button type can correspond to one or more of the parameters found in the GetStandardButtonId method (StatusBar class).

### Data Type

Long

### Syntax

getbuttontypevalue = [objectreference].GetButtonType

### Legal values

The legal values for this property are determined by the following ButtonType parameters in the CreateNewButton method (StatusBar class):

LwpButtonBehaviorClickable (&H8) A value that allows the button to be left-clicked.

LwpButtonBehaviorCollapsible (&H10) A value that allows the button to shrink or grow so that the status bar can fill up the window. Only one is allowed per status bar. Word Pro's collapsible button is the date/time button.

LwpButtonBehaviorContainer (&H20) A value that allows the button to contain child buttons.

LwpButtonBehaviorLeftclick (&H8) A value that allows the button to be left-clicked.

LwpButtonBehaviorPopup (&H4) A value that allows the button to pop up a list of alternatives.

LwpButtonBehaviorThermometer (&H80000) A value that allows the button to display a thermometer graphic with percentages.

LwpButtonCanBeDepressed (&H40000) A value that allows the button to stay depressed.

LwpButtonContentsCenterAligned (&H80) A value that allows the button contents to be center-aligned.

LwpButtonContentsGray (&H200) A value that allows the button contents to be grayed.

LwpButtonContentsHilited (&H400) A value that allows the button contents to be highlighted (red in Word Pro).

LwpButtonContentsLeftAligned (&H40) A value that allows the button contents to be left-aligned.

LwpButtonContentsRightAligned (&H100) A value that allows the button contents to be right-aligned.

LwpButtonHasAutorepeat (&H4000) A value that allows the button to repeat a command.

LwpButtonHasUpdownCtrl (&H20000) A value that allows the button to have up/down control.

LwpButtonNoTextFromHost (&H800) A value that allows the button to keep its user-defined text without changing; in other words, the text on this button is never going to require text from a host.

LwpButtonReserved (&H8000)

LwpButtonSpacer (&H10000) A spacer status bar button.

LwpButtonSupportDbiClick (&H2000) A value that allows the button to respond to a double-click .

LwpButtonSupportRightClick (&H1000) A value that allows the button to support a right mouse click.

LwpButtonTypeGraphics (&H2) A value that allows the button to display a graphic.

LwpButtonTypeText (&H1) A value that allows the button to display text.

### Usage

This property lets you determine the properties for the selected button.



**Word Pro: GetConversationHandle property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETCONVERSATIONHANDLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is used internally and cannot be changed.

**Data Type**

Long

**Syntax**

getconversationhandlevalue = [objectreference].GetConversationHandle

[objectreference].GetConversationHandle = getconversationhandlevalue

**Legal values****Usage**

This property is used internally and cannot be changed.

**Word Pro: GetFilterExtension property**

{button ,AL('H\_FILTERHELPER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETFILTEREXTENSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

getfilterextensionvalue = [objectreference].GetFilterExtension

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: GetFilterExtForDialogBox property**

{button ,AL('H\_FILTERHELPER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETFILTEREXTFORDIALOGBOX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

getfilterextfordialogboxvalue = [objectreference].GetFilterExtForDialogBox

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: GetFilterId property**

{button ,AL('H\_FILTERHELPER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETFILTERID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

getfilteridvalue = [objectreference].GetFilterId

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: GetFormatName property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETFORMATNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Specifies the format of the data (for example, CF\_TEXT, CF\_BITMAP, and so on).

**Data Type**

[String](#)

**Syntax**

getformatnamevalue = [objectreference].GetFormatName

**Legal values****Usage**

Allows you to inquire about the format of the data.

**Word Pro: GetHomeDirectory property**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_GETHOMEDIRECTORY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains the name of the home directory for the Windows operating system.

**Data Type**

[String](#)

**Syntax**

gethomedirectoryvalue = [objectreference].GetHomeDirectory

**Legal values****Usage**

## **Word Pro: GetItemName property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETITEMNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) ItemName is the name of a Dde link to another application. For example, if you link to Lotus 1-2-3 and paste link a range into a Word Pro document, the name of the range is the ItemName. If you do not name the range, Word Pro names it for you.

### **Data Type**

String

### **Syntax**

getitemnamevalue = [objectreference].GetItemName

[objectreference].GetItemName = getitemnamevalue

### **Legal values**

### **Usage**

You can inquire about the ItemName or change the ItemName. If you change the ItemName and update the link, you will get different data.

## **Word Pro: GetRedoWhatDesc property**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETREDOWHATDESC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

getredowhatdescvalue = [objectreference].GetRedoWhatDesc

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



### **Word Pro: GetServerName property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSERVERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The actual executable name of the server (for example, "1-2-3.EXE").

### **Data Type**

[String](#)

### **Syntax**

getservernamevalue = [objectreference].GetServerName

### **Legal values**

### **Usage**

You can inquire about the server name if you paste link an object. If you create the object manually, you must provide the name of the server, the topic name (usually the file name), and the item name (for example, a range, a bookmark, and so on).

## **Word Pro: GetStatus property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSTATUS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not a DdeLink is active or inactive.

### **Data Type**

[Integer](#)

### **Syntax**

getstatusvalue = [objectreference].GetStatus

[objectreference].GetStatus = getstatusvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

DdeLinks can be active or inactive. If the update flag is set, you automatically get an update when changes are made to a DdeLink. For example, if you paste link a range from Lotus 1-2-3 into a Word Pro document and make changes to the range in 1-2-3, the Word Pro document is automatically updated when you save, if the update flag is set,.

**Word Pro: GetTopicName property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETTOPICNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The topic name (usually the file name) of the DdeLink.

**Data Type**

[String](#)

**Syntax**

gettopicnamevalue = [objectreference].GetTopicName

[objectreference].GetTopicName = gettopicnamevalue

**Legal values****Usage**

This property tells you the file name of the DdeLink. You can use it for listing linked files.

## **Word Pro: GetUndoWhatDesc property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETUNDOWHATDESC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

getundowhatdescvalue = [objectreference].GetUndoWhatDesc

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: GetValue property**

{button ,AL('H\_CELLLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETVALUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns a string representing the numeric content of a cell.

**Data Type**

String

**Syntax**

getvaluevalue = [objectreference].GetValue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The value returned by this property is rounded to two decimal places. If a cell's content is not numeric, the GetValue property will contain an empty string.

## **Word Pro: GlossaryDataFileName property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYDATAFILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the default (first) Glossary file.

### **Data Type**

String

### **Syntax**

glossarydatafilenamevalue = [objectreference].GlossaryDataFileName

[objectreference].GlossaryDataFileName = glossarydatafilenamevalue

### **Legal values**

### **Usage**

Equivalent to the "Default glossary file(s)" field on the Default files panel of the Word Pro Preferences dialog box. In the Word Pro interface, the "Default glossary file(s)" field can contain multiple file names. You can use this property to change or read the default or first Glossary file name, or you can use the property, `GlossaryDataFiles`, to set all the files to one file.

## **Word Pro: GlossaryPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The default path (drive and directory) where Word Pro looks for the Glossary file.

### **Data Type**

String

### **Syntax**

glossarypathvalue = [objectreference].GlossaryPath

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to the "Glossaries" field on the Locations panel of the Word Pro Preferences dialog box. In the Word Pro interface, the "Glossaries" field can contain multiple paths. You can use this property to clear all paths and set the default or first glossary path, or you can use the property, GlossaryDataPaths, to read multiple paths entered by the user.

## **Word Pro: GoToIndexLastItemAccessed property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOINDEXLASTITEMACCESSED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The value of the last item a user accesses from the GoTo dialog box.

### **Data Type**

[Integer](#)

### **Syntax**

gotoindexlastitemaccessedvalue = [objectreference].GoToIndexLastItemAccessed

[objectreference].GoToIndexLastItemAccessed = gotoindexlastitemaccessedvalue

### **Legal values**

Index of zero-based integers that correspond to the items in the second combo box in the Go To dialog box. For example, if the box contains First and Last, First corresponds to 0 and Last corresponds to 1. If the second box is empty, the value for this property is -1.

### **Usage**

Equivalent to the "Type of document part to go to" field in the Go To dialog box. Word Pro stores this value only during the current session.



**Word Pro: GoToItemName property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOITEMNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the last item a user accesses from the GoTo dialog box.

**Data Type**

[String](#)

**Syntax**

gotoitemnamevalue = [objectreference].GoToItemName

[objectreference].GoToItemName = gotoitemnamevalue

**Legal values****Usage**

This property can contain a page number, First, Last, Next, Prev, or the name of a document part. For example, if the user chooses "ClickHere" in the "Type of document part to go to" box, this property contains the name of the Click Here Block (if the Click Here Block is named). If the user chooses "Page" in this box, this property could contain First, Last, a page number, or a page name.

## **Word Pro: GoToPageLoadingBackground property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOPAGELOADINGBACKGROUND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The value to indicate if "Generate all page listings now" in the Go To dialog box is enabled.

### **Data Type**

Integer

### **Syntax**

GoToPageLoadingBackgroundvalue = [objectreference.]GoToPageLoadingBackground

[objectreference.]GoToPageLoadingBackground = GoToPageLoadingBackgroundvalu

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default value is False (0).

### **Usage**

Equivalent to the "Generate all page listings now" box in the Go To dialog box. If the legal value for this property is False, Word Pro does not update page descriptions. If the legal value for this property is True, Word Pro updates page descriptions when you add new pages to the current document. Also, it will update and display all page descriptions in a lengthy or imported document.

This property cannot be accessed by using the Go To accelerators.

## **Word Pro: GoToPageNumber property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOPAGENUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The last page number accessed in the Go to dialog box

### **Data Type**

[Integer](#)

### **Syntax**

gotopagenumvalue = [objectreference].GoToPageNumber

[objectreference].GoToPageNumber = gotopagenumvalue

### **Legal values**

### **Usage**

This property is only valid if "Page" was the last document part specified in the "Type of document part to go to" box in the Go To dialog box.

## **Word Pro: GoToSelection property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The last selection in the "Type of document part to go to" list in the Go To dialog box.

### **Data Type**

Variant (Enumerated)

### **Syntax**

gotoselectionvalue = [objectreference].GoToSelection

[objectreference].GoToSelection = gotoselectionvalue

### **Legal values**

\$LwpGoToTypeFirstpage (378)

\$LwpGoToTypeLastpage (379)

\$LwpGoToTypeNextitem (381)

\$LwpGoToTypePagenumber (380)

\$LwpGoToTypePrevitem (382)

### **Usage**

The value for this property depends on the document part the user chooses in the "Type of document part to go to:" box. For example, if a user chooses "Page," the value for this property can be Firstpage, Lastpage, or a Pagenumber. However, if a user chooses "Page break," the value for this property can only be Next or Previous.

This property is used to provide the last Go to selection for the Go to accelerator.

## **Word Pro: GrammarFormalityLevel property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAMMARFORMALITYLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates the formality level when Grammar Check reviews a document.

### **Data Type**

[Integer](#)

### **Syntax**

grammarformalitylevelvalue = [objectreference].GrammarFormalityLevel

[objectreference].GrammarFormalityLevel = grammarformalitylevelvalue

### **Legal values**

The legal values for this property are the enum values of 0 (Informal), 1 (Standard), and 2 (Formal). Default is 1.

### **Usage**

Use this property when you want to set the formality level for proofing a document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the formality level in the "Formality" field on the Rules panel. There are three formality levels: informal, standard and formal.

**Word Pro: GrammarProofLevel property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAMMARPROOFLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates the level of proofing when Grammar Check reviews a document.

**Data Type**

[Integer](#)

**Syntax**

grammarprooflevelvalue = [objectreference].GrammarProofLevel

[objectreference].GrammarProofLevel = grammarprooflevelvalue

**Legal values**

The legal values for this property are 0 (full proof) and 1 (quick proof). Default is 1.

**Usage**

Use this property when you are choosing the proofing level for a document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the proofing level in the "Grammar Check level" field on the Rules panel. There are two proofing levels: full proof and quick proof.

## **Word Pro: GrammarSetName property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAMMARSETNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

grammarsetnamevalue = [objectreference].GrammarSetName

[objectreference].GrammarSetName = grammarsetnamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: Green property**

{button ,AL('H\_COLOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GREEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The green component of a color.

**Data Type**

Integer

**Syntax**

greenvalue = [objectreference].Green

[objectreference].Green = greenvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

You can set or read the current level of green in a specific object's color.



**Word Pro: Greeting property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GREETING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The text that displays after Greeting option is set.

**Data Type**

String

**Syntax**

greetingvalue = [objectreference].Greeting

[objectreference].Greeting = greetingvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to choosing File - TeamSecurity and entering text in the "Display Greeting with this text" box on the Editing Rights panel. You must set the UseGreeting property which tells Word Pro to display a Greeting box before you can enter text to display in the Greeting box.

## Word Pro: GridDistance property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_GRIDDISTANCE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the grid spacing for layout objects.

### Data Type

Long

### Syntax

griddistancevalue = [objectreference].GridDistance

[objectreference].GridDistance = griddistancevalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

Equivalent to the "Grid settings" value that is located on the Misc panel of the InfoBox for certain layout objects.

## Word Pro: GridType property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_GRIDTYPE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the type of grid that is displayed in a layout object.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

gridtypevalue = [objectreference].GridType

[objectreference].GridType = gridtypevalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpGridTypeDots (2061)	Setting this value displays a specific type of dotted grid in the layout object.
\$LwpGridTypeLines (2060)	Setting this value displays a specific type of lined grid in the layout object.
\$LwpGridTypeNone (2059)	Setting this value prevents a grid from showing in the layout object.

## Usage

Equivalent to the "Grid settings" value which is located on the Misc panel of the InfoBox for certain layout objects.

**Word Pro: Hang property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HANG\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

hangvalue = [objectreference].Hang

[objectreference].Hang = hangvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: HasContents property**

{button ,AL('H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HASCONTENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

hascontentsvalue = [objectreference].HasContents

[objectreference].HasContents = hascontentsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: HasFocus property**

{button ,AL(^H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_HASFOCUS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

hasfocusvalue = [objectreference].HasFocus

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: HasLocalTabs property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HASLOCALTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not the current paragraph has local tab settings, as opposed to tab settings found in the paragraph style.

**Data Type**

Variant (Enumerated)

**Syntax**

haslocaltabsvalue = [objectreference].HasLocalTabs

**Legal values**

\$LwpCommandResponseNo (148)

\$LwpCommandResponseYes (149)

**Usage**

**Word Pro: HasTabs property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HASTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

hastabsvalue = [objectreference].HasTabs

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## **Word Pro: Heading property**

{button ,AL('H\_NUMBERING\_CLASS;H\_OUTLINESYSEQUENCE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)}

[See list of classes](#)

{button ,AL('H\_HEADING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

headingvalue = [objectreference].Heading

[objectreference].Heading = headingvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: Height property

```
{button ,AL('H_APPLICATIONWINDOW_CLASS;H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DOCWINDOW_CLASS;H_DROPCAPCONTAINER_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GRAPHIC_CLASS;H_GRAPHICOLEBJECT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_ICONBAR_CLASS;H_LAYOUT_CLASS;H_NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS;H_OLEOBJECT_CLASSES;H_PAGECONTAINER_CLASS;H_PAGELAYOUT_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASSES;H_RUBYLAYOUT_CLASS;H_STATUSBAR_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WINDOW_CLASS',0)} See list of classes
```

```
{button ,AL('H_HEIGHT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The distance in Twips from the top border of an object to the bottom border of an object. The actual returned height of the entire object.

[ApplicationWindow]

The actual height of the application window.

[IconBar]

The actual height of an icon bar object.

[StatusBar]

The returned actual height of the status bar window.

[Layout]

Allows you to set or return the actual height of a layout object.

(Read-Only) [Container]

Allows you to return the actual height of a container object.

## Data Type

Long

## Syntax

heightvalue = [objectreference].Height

[objectreference].Height = heightvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

This property is not valid for IconBarManager.

Used to represent the actual returned height of an object, such as the status bar or the application window.

**Word Pro: HelpText property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HELPTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The string that goes in the bubble help for a ClickHere block.

**Data Type**

[String](#)

**Syntax**

helptextvalue = [objectreference].HelpText

[objectreference].HelpText = helptextvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: HiddenMode property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIDDENMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

hiddenmodevalue = [objectreference].HiddenMode

[objectreference].HiddenMode = hiddenmodevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: HideFormula property**

{button ,AL('H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIDEFORMULA\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

hideformulavalue = [objectreference].HideFormula

[objectreference].HideFormula =hideformulavalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: HideOutlineLevels property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIDEOUTLINELEVELS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

hideoutlinelevelsvalue = [objectreference].HideOutlineLevels

[objectreference].HideOutlineLevels = hideoutlinelevelsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: HighLightMode property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIGHLIGHTMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

highlightmodevalue = [objectreference].HighLightMode

[objectreference].HighLightMode = highlightmodevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: HorizontalSplitWindow property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HORIZONTALSPLITWINDOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if Word Pro will split the .MDI window horizontally when the NewWindow method is called.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

horizontalSplitWindowvalue = [objectreference].HorizontalSplitWindow

[objectreference].HorizontalSplitWindow = horizontalSplitWindowvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default value is False (0).

### **Usage**

Equivalent to the Split Top-Bottom option on the View menu.



**Word Pro: HorizScrollBarVisible property**

{button ,AL(^H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_HORIZSCROLLBARVISIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

horizscrollbarvisiblevalue = [objectreference].HorizScrollBarVisible

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: HyphenateLastWordInColumnOrPage property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HYPHENATELASTWORDINCOLUMNORPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

hyphenatelastwordincolumnorpagevalue = [objectreference].HyphenateLastWordInColumnOrPage

[objectreference].HyphenateLastWordInColumnOrPage = hyphenatelastwordincolumnorpagevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: HyphenateLastWordInPara property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HYPHENATELASTWORDINPARA\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

HyphenateLastWordInParavalue = [objectreference].HyphenateLastWordInPara

[objectreference].HyphenateLastWordInPara = HyphenateLastWordInParavalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: HyphZoneAfter property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HYPHZONEAFTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

hyphzoneaftervalue = [objectreference].HyphZoneAfter

[objectreference].HyphZoneAfter = hyphzoneaftervalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: HyphZoneBefore property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HYPHZONEBEFORE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

HyphZoneBeforevalue = [objectreference].HyphZoneBefore

[objectreference].HyphZoneBefore = [objectreference].HyphZoneBefore

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: IconBarPositionState property

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONBARPOSITIONSTATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Tells you whether or not the icon bar object is fixed or floating on the workspace.

### Data Type

Variant (Enumerated)

### Syntax

iconbarpositionstatevalue = [objectreference].IconBarPositionState

### Legal values

\$LwplIconBarPositionStateFixed (392) Displays a set of SmartIcons at the sides of the workspace in a fixed position (left, right, top, or bottom). If you drag a SmartIcons bar to an edge, it will snap to and stay with that edge, even when the workspace window is moved.

\$LwplIconBarPositionStateFloating (393) Displays a set of SmartIcons in a floating position anywhere inside or outside the workspace. Dragging SmartIcons to a place other than an edge creates a floating palette. You can also drag one palette of SmartIcons over another.

### Usage

Tells you whether or not the icon bar object is fixed or floating. A fixed icon bar object will be attached on the left, right, top, or bottom of the workspace. A icon bar object floats anywhere except at an edge, inside or outside the workspace

**Word Pro: IconHelpText property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONHELPTXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to inquire about or change the text that appears in the bubble help when the cursor hovers over an icon.

**Data Type**

[String](#)

**Syntax**

iconhelptextvalue = [objectreference].IconHelpText

[objectreference].IconHelpText = iconhelptextvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

If you change the IconHelpText for a specific icon using this property, your changes will display the next time the cursor hovers over the icon. You must first select the icon using either the SelectStandardIcon or SelectCustomIcon method for this property to function.

## **Word Pro: IconPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the default path (drive and directory) for SmartIcons.

### **Data Type**

String

### **Syntax**

iconpathvalue = [objectreference].IconPath

[objectreference].IconPath = iconpathvalue

### **Legal values**

A valid path including drive and directory.

### **Usage**

Equivalent to the "SmartIcons" field on the Locations panel of the Word Pro Preferences dialog box. In the Word Pro interface, the "SmartIcons" field can contain multiple paths. You can use this property to clear all paths before setting the default or first icon path, or you can use the property, IconsPaths, to read multiple paths entered by the user.



**Word Pro: IconSetName property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONSETNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An indicator that tells you the name of the icon bar object. Found in the "Bar name" field in the SmartIcons Setup dialog box.

**Data Type**

[String](#)

**Syntax**

iconsetnamevalue = [objectreference].IconSetName

**Legal values****Usage**

A read-only indicator that returns the icon bar object name. For example, you can enumerate the names in the IconBarManager. In order to get into the Icon Bar class, you can select the icon bar you want from the IconBarManager and confirm its name to make sure it is the bar you want.

## **Word Pro: IconShowingBeforeCleanScreen property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONSHOWINGBEFORECLEANSCREEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not any SmartIcons were displayed before Clean Screen was enabled.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

iconshowingbeforecleanscreenvalue = [objectreference].IconShowingBeforeCleanScreen

[objectreference].IconShowingBeforeCleanScreen = iconshowingbeforecleanscreenvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is dependent upon the default value for showing icons.

### **Usage**

Use this property to indicate if Word Pro should display SmartIcons when the user leaves Clean Screen mode.

## Word Pro: Action property

```
{button ,AL(##H_CLICKHERE_CLASS;H_MENUITEM_CLASS;H_MENUITEM_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_MENUITEM_ACTION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The Action property is either the name of the script function that executes when you select a menu item or another menu ID.

### Data Type

String

### Syntax

actionvalue = [objectreference].Action

[objectreference].Action = actionvalue

### Legal values

### Usage

When you select a menu item, Word Pro executes the Action property for the item. The Action property can be the name of a script function that you create, or it can be the name of a predefined Word Pro WmCommand. WmCommands are unique IDs that Word Pro uses to differentiate each menu item.

When you use the Action property as another menu ID, it allows you to emulate the actions of a predefined Word Pro menu item and apply them to your customized menu item.

### **Word Pro: Foundry property**

{button ,AL('==H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS;H\_DIVISION\_CLASSES;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTDOCUMENT\_FOUNDRY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) This property is not used in this version of LotusScript. Use the Foundry property on the Division object or one of the WPAApplication Foundry objects.

### **Data Type**

Foundry

### **Syntax**

foundryvalue = [objectreference].Foundry

### **Legal values**

### **Usage**

'Example: ApplyDerogatory property

'This example script has not yet been created.

'Example: ApplyDifferentPrep property  
'This example script has not yet been created.

'Example: ApplyDoubleNegative property

'This example script has not yet been created.

'Example: ApplyDoublePlural property

'This example script has not yet been created.



'Example: ApplyDoubleWordCheck property

'This example script has not yet been created.

'Example: ApplyElision property

'This example script has not yet been created.

'Example: ApplyEnglishDerived property

'This example script has not yet been created.

'Example: ApplyEnglishWords property

'This example script has not yet been created.

'Example: ApplyExotic property

'This example script has not yet been created.

'Example: ApplyExtraPrepositionCheck property  
'This example script has not yet been created.

'Example: ApplyFalseFriend property

'This example script has not yet been created.

'Example: ApplyFemaleOccupation property

'This example script has not yet been created.



'Example: ApplyFixedExpression property  
'This example script has not yet been created.

'Example: ApplyForeignWord property

'This example script has not yet been created.

'Example: ApplyFormalTerms property

'This example script has not yet been created.

'Example: ApplyFormatErrors property

'This example script has not yet been created.

'Example: ApplyGallicisms property

'This example script has not yet been created.

'Example: ApplyGenderExpressions property  
'This example script has not yet been created.

'Example: ApplyGermanisms property

'This example script has not yet been created.

'Example: ApplyHomoGraphs property  
'This example script has not yet been created.



'Example: ApplyHomonymsEasy property

'This example script has not yet been created.

'Example: ApplyHomonymsHard property

'This example script has not yet been created.

'Example: ApplyHomonyms property

'This example script has not yet been created.

'Example: ApplyHomoPhonel property

'This example script has not yet been created.

'Example: ApplyHomoPhone2 property

'This example script has not yet been created.

'Example: ApplyHomoPhones property

'This example script has not yet been created.

'Example: ApplyIncorrectPlural property  
'This example script has not yet been created.

'Example: ApplyInformalExpressions property  
'This example script has not yet been created.



'Example: ApplyJargonWords property

'This example script has not yet been created.

'Example: ApplyLowercaseAdjective property  
'This example script has not yet been created.

'Example: ApplyLowercaseColor property

'This example script has not yet been created.

'Example: ApplyLowercaseNumbers property  
'This example script has not yet been created.

'Example: ApplyLowercasePhrases property

'This example script has not yet been created.

'Example: ApplyLowercasePronouns property

'This example script has not yet been created.

'Example: ApplyMassVsCount property  
'This example script has not yet been created.

'Example: ApplyMisspelledExpressions property  
'This example script has not yet been created.



'Example: ApplyMisspelledForeignExpressions property  
'This example script has not yet been created.

'Example: ApplyMisspelledItalian property  
'This example script has not yet been created.

'Example: ApplyMisspelledWords property

'This example script has not yet been created.

'Example: ApplyMisusedWords property

'This example script has not yet been created.

'Example: ApplyNonStandardExpression property  
'This example script has not yet been created.

'Example: ApplyNonStandardModifiers property  
'This example script has not yet been created.

'Example: ApplyNoudModifierOrderCheck property  
'This example script has not yet been created.

'Example: ApplyNounConsistency property

'This example script has not yet been created.



'Example: ApplyNSContract property  
'This example script has not yet been created.

'Example: ApplyNSNegation property

'This example script has not yet been created.

'Example: ApplyNSPrep property

'This example script has not yet been created.

'Example: ApplyNSSpell property

'This example script has not yet been created.

'Example: ApplyOpenClosedSpelling property  
'This example script has not yet been created.

'Example: ApplyPrepositionalPhrases property  
'This example script has not yet been created.

'Example: ApplyPretentiousWords property  
'This example script has not yet been created.

'Example: ApplyPronounErrors property

'This example script has not yet been created.



'Example: ApplyRedundantExpressions property  
'This example script has not yet been created.

'Example: ApplyRegionalExpression property  
'This example script has not yet been created.

'Example: ApplyRelatedWord property

'This example script has not yet been created.

'Example: ApplySexistExpressions property  
'This example script has not yet been created.

'Example: ApplySpellStandard property

'This example script has not yet been created.

'Example: ApplySplitInfinitives property  
'This example script has not yet been created.

'Example: ApplySubjectVerbAgreement property  
'This example script has not yet been created.

'Example: ApplyTwoGender property

'This example script has not yet been created.



'Example: ApplyTypicalMisspell property

'This example script has not yet been created.

'Example: ApplyVagueQuantifiers property  
'This example script has not yet been created.

'Example: ApplyVerbGroupConsistency property  
'This example script has not yet been created.

```
'Example: ApplyWeakModifiers property  
'This example script has not yet been created.
```

```
'Example: AppViewPrefs property
''This example changes the color of the application window by
manipulating the PaneColor property.
'You should close all must have a document open for this script
to work.
'Paste this script into Sub Main in the Globals section.
Print .ApplicationWindow.Height
.ApplicationWindow.Height = (.5 * .ApplicationWindow.Height)
Print .ApplicationWindow.Height
```

'Example: AreDisabledIconsGrayed property  
'This example script has not yet been created.

'Example: Ascent property

'This example script has not yet been created.

'Example: AsciiCRLFType property

'This example script has not yet been created.



'Example: AtBeginningOfLine property

'This example script has not yet been created.

'Example: AtBeginningOfObject property

'This example script has not yet been created.

'Example: AtBeginningOfParagraph property

'This example script has not yet been created.

'Example: AtBeginningOfStream property

'This example script has not yet been created.

'Example: AtBeginningOfWord property

'This example script has not yet been created.

'Example: AtBeginning property

'This example script has not yet been created.

'Example: AtEndOfLine property

'This example script has not yet been created.

'Example: AtEndOfObject property

'This example script has not yet been created.



'Example: AtEndOfParagraph property

'This example script has not yet been created.

'Example: AtEndOfStream property

'This example script has not yet been created.

'Example: AtEndOfWord property

'This example script has not yet been created.

'Example: AtEnd property

'This example script has not yet been created.

'Example: Attributes property

'This example script has not yet been created.

'Example: AttrStyleName property

'This example script has not yet been created.

```
'Example: AuthorName property
With WordPro.ActiveDocument
MsgTxt = "Current Word Pro Doc is " & .FullName
MsgTxt = MsgTxt & ", the author is " & .Docinfo.AuthorName
Msgbox MsgTxt,64,"Word Pro Information"
MsgTxt = "It was created on " & .DocInfo.CreationDateString & " at "
& .DocInfo.CreationTimeString
Msgbox MsgTxt,64,"Word Pro Information"
End With
```

'Example: AutoBackup property

'This example script has not yet been created.



'Example: AutoHyphenate property

'This example script has not yet been created.

'Example: AutomaticLink property

'This example script has not yet been created.

'Example: AutoRunMacro property

'This example script has not yet been created.

'Example: AutoSaveMinutes property

'This example script has not yet been created.

'Example: AutoSave property

'This example script has not yet been created.

'Example: AutoVersion property

'This example script has not yet been created.

'Example: BackColor property

'This example script has not yet been created.

'Example: BackgroundPrintingOn property

'This example script has not yet been created.



'Example: Background property

'This example script has not yet been created.

```
'Example: Backspace method
' This example inserts 5 words into the current document and then backspaces
' 5 characters.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim WordNumber as Integer
For WordNumber = 1 To 5
    .Text.InsertText "Word" & Format$(WordNumber) & " "
Next
MessageBox "Click OK to backspace 5 characters.",MB_OK,"Example Script"
.Text.BackSpace (5)
```

'Example: BackupPaths property

'This example script has not yet been created.

'Example: BackupPath property

'This example script has not yet been created.

```
'Example: Backward method
' This example inserts 5 sentences with 5 words each into the current
' document. The cursor location is then moved backward 2 sentences and then
' moved forward one word.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim SentenceNumber as Integer
Dim WordNumber as Integer
For SentenceNumber = 1 To 5
    For WordNumber = 1 To 5
        .Text.InsertText "Word" & Format$(WordNumber) & " "
    Next
    .Type (". ")
Next
MessageBox "Click OK to backup 2 senteces.",MB_OK,"Example Script"
.Text.Backward $LwpNavigateObjectTypeSentence, 2
MessageBox "Click OK to go forward 1 word.",MB_OK,"Example Script"
.Text.Forward $LwpNavigateObjectTypeWord, 1
```

'Example: Bags property

'This example script has not yet been created.

## **Word Pro: ApplyNSContract property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSCONTRACT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynscontractvalue = [objectreference].ApplyNSContract

[objectreference].ApplyNSContract = applynscontractvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSGeography property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSGEOGRAPHY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynsgeographyvalue = [objectreference].ApplyNSGeography

[objectreference].ApplyNSGeography = applynsgeographyvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyNSInflection property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSINFLECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynsinflectionvalue = [objectreference].ApplyNSInflection

[objectreference].ApplyNSInflection = applynsinflectionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSNegation property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSNEGATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynsnegationvalue = [objectreference].ApplyNSNegation

[objectreference].ApplyNSNegation = applynsnegationvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyNSPrep property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSPREP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applynsprepvalue = [objectreference].ApplyNSPrep

[objectreference].ApplyNSPrep = applynsprepvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSPronoun property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSPRONOUN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynspronounvalue = [objectreference].ApplyNSPronoun

[objectreference].ApplyNSPronoun = applynspronounvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyNSSpell property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSSPELL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applynspellvalue = [objectreference].ApplyNSSpell

[objectreference].ApplyNSSpell = applynspellvalu

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyNSUsage property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSUSAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applynsusagevalue = [objectreference].ApplyNSUsage

[objectreference].ApplyNSUsage = applynsusagevalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyNSVerbForm property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYNSVERBFORM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applynsverbformvalue = [objectreference].ApplyNSVerbForm

[objectreference].ApplyNSVerbForm = applynsverbformvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyOpenClosedSpelling property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYOPENCLOSEDSPELLING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking incorrect use of spaces in words or phrases.

### Data Type

[Integer](#)

### Syntax

applyopenclosedspellingvalue = [objectreference].ApplyOpenClosedSpelling

[objectreference].ApplyOpenClosedSpelling = applyopenclosedspellingvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Open vs. closed spelling" in the "Rule type" field on the Rules panel.

This rule flags spelling errors that result from incorrect use of spaces. The correct spelling is offered as an alternative (for example, "in a while" to "in awhile").

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyOpenUsage property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYOPENUSAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyopenusagevalue = [objectreference].ApplyOpenUsage

[objectreference].ApplyOpenUsage = applyopenusagevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyOverusedPhrases property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYOVERUSEDPHRASES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking overused phrases.

### Data Type

[Integer](#)

### Syntax

applyoverusedphrasesvalue = [objectreference].ApplyOverusedPhrases

[objectreference].ApplyOverusedPhrases = applyoverusedphrasesvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Overused phrases" in the "Rule type" field on the Rules panel.

This rule flags overused expressions that have lost their original impact. In a spoken context, they might be acceptable, but in writing, the phrases are too casual and should be replaced with expressions that are less colloquial and more precise. In some cases, the sentence must be rephrased to avoid an overused expression. For example, the phrase, "blissful ignorance" might be replaced by just the word, "ignorance."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyPassiveVerbErrors property

{button ,AL(^H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_APPLYPASSIVEVERBERRORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking the use of passive voice.

### Data Type

[Integer](#)

### Syntax

applypassiveverberrorsvalue = [objectreference].ApplyPassiveVerbErrors

[objectreference].ApplyPassiveVerbErrors = applypassiveverberrorsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Passive voice usage" in the "Rule type" field on the Rules panel.

This rule flags using the passive voice where the subject is acted upon but completes no action, as in "Our proposal was accepted by the board." The counterpart in active voice is, "The board accepted our proposal."

Sentences written in the passive voice can sound weak and indirect; it is often better to rephrase them as active. Occasionally, a sentence makes sense only in the passive voice, but most sentences are more coherent when the subject completes the action.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyPostClitAgree property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYPOSTCLITAGREE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applypostclitagreevalue = [objectreference].ApplyPostClitAgree

[objectreference].ApplyPostClitAgree = applypostclitagreevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, Italian.

## **Word Pro: ApplyPrepExpression property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYPREPEXPRESSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyprepexpressionvalue = [objectreference].ApplyPrepExpression

[objectreference].ApplyPrepExpression = applyprepexpressionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyPrepositionalPhrases property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYPREPOSITIONALPHRASES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking prepositional phrases.

### Data Type

[Integer](#)

### Syntax

applyprepositionalphrasesvalue = [objectreference].ApplyPrepositionalPhrases

[objectreference].ApplyPrepositionalPhrases = applyprepositionalphrasesvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Unnecessary prepositions" in the "Rule type" field on the Rules panel.

This rule flags expressions that include an unnecessary preposition and suggests deleting it to make the expression more concise. For example, in the sentence, "I sat down on the lawn," the preposition "down" is superfluous since it is implied by "sat."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyPretentiousWords property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYPRETENTIOUSWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking pretentious words.

### **Data Type**

[Integer](#)

### **Syntax**

applypretentiouswordsvvalue = [objectreference].ApplyPretentiousWords [objectreference].ApplyPretentiousWords = applypretentiouswordsvvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Pretentious words" in the "Rule type" field on the Rules panel.

This rule flags unnecessarily complex words and offers simple, straightforward alternatives. For example, the word, "eventuate," can usually be replaced with the expression, "takes place," or "happens."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyPronounErrors property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYPRONOUNERRORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking pronoun errors.

### Data Type

[Integer](#)

### Syntax

applypronounerrorsvalue = [objectreference].ApplyPronounErrors

[objectreference].ApplyPronounErrors = applypronounerrorsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Pronoun errors" in the "Rule type" field on the Rules panel.

This rule checks pronouns for errors in case and order. For example, one rule flags the sentence, "They are baking a cake for my sister and I," and notes the pronoun "I" is incorrect. The error message explains that because "I" is the object of a preposition, it should be the objective case (me). This rule will also flag the relative pronoun, "which," when used in a restrictive clause and recommends using "that."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyPunctuationErrors property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYPUNCTUATIONERRORS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking punctuation errors.

### Data Type

[Integer](#)

### Syntax

applypunctuationerrorsvalue = [objectreference].ApplyPunctuationErrors

[objectreference].ApplyPunctuationErrors = applypunctuationerrorsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Punctuation errors" in the "Rule type" field on the Rules panel.

This rule flags punctuation errors, such as the incorrect placement of commas in specific expressions and inappropriate punctuation of parenthetical or quoted material. It also flags doubled punctuation.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyRedundantExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYREDUNDANTEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking redundant expressions.

### Data Type

[Integer](#)

### Syntax

applyredundantexpressionsvalue = [objectreference].ApplyRedundantExpressions

[objectreference].ApplyRedundantExpressions = applyredundantexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Redundant expressions" in the "Rule type" field on the Rules panel.

This rule flags expressions containing multiple words that mean or imply the same thing. Redundancy can often be eliminated by deleting part of the expression. For example, "sufficient enough" might become either "sufficient" or "enough."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyRegionalExpression property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYREGIONALEXPRESSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyregionalexpressionvalue = [objectreference].ApplyRegionalExpression

[objectreference].ApplyRegionalExpression = applyregionalexpressionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyRelatedWord property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYRELATEDWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyrelatedwordvalue = [objectreference].ApplyRelatedWord

[objectreference].ApplyRelatedWord = applyrelatedwordvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplySensitiveExp property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSENSITIVEEXP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

appliesensitiveexpvalue = [objectreference].ApplySensitiveExp

[objectreference].ApplySensitiveExp = appliesensitiveexpvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplySexistExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSEXISTEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking sexist expressions.

### Data Type

[Integer](#)

### Syntax

applysexistexpressionsvalue = [objectreference].ApplySexistExpressions

[objectreference].ApplySexistExpressions = applysexistexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Sexist expressions" in the "Rule type" field on the Rules panel.

This rule flags expressions that are considered to be sexist or stereotypical, based on gender. The error messages may offer alternatives or suggest rephrasing the sentence. For example, the rule will flag the sentence, "Say hello to the little woman," and suggest the words, "wife," "partner," or "spouse," as alternatives.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplySpellStandard property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSPELLSTANDARD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyspellstandardvalue = [objectreference].ApplySpellStandard

[objectreference].ApplySpellStandard = applyspellstandardvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. This property can only be found in specific language versions of Word Pro, for example, French.

## Word Pro: ApplyStockPhrase property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSTOCKPHRASE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking commonly used phrases.

### Data Type

[Integer](#)

### Syntax

applystockphrasevalue = [objectreference].ApplyStockPhrase

[objectreference].ApplyStockPhrase = applystockphrasevalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Stock phrases" in the "Rule type" field on the Rules panel.

This rule flags stock phrases that may often be deleted without changing the meaning or emphasis of a sentence. Unlike clichés, these expressions are not worn-out metaphors or conventional descriptive phrases. They are introductory or parenthetical expressions often used as filler (for example, "it goes without saying that.." or "in fact").

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## **Word Pro: ApplyStyleParameters property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSTYLEPARAMETERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applystyleparametersvalue = [objectreference].ApplyStyleParameters

[objectreference].ApplyStyleParameters = applystyleparametersvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplySubjectVerbAgreement property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSUBJECTVERBAGREEMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking agreement between subjects and verbs.

### Data Type

[Integer](#)

### Syntax

applysubjectverbagreementvalue = [objectreference].ApplySubjectVerbAgreement

[objectreference].ApplySubjectVerbAgreement = applysubjectverbagreementvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Subject-verb agreement errors" in the "Rule type" field on the Rules panel.

This rule flags errors of agreement between verbs and their subjects. The rule checks to see whether verbs contain the right number (singular or plural) and the right person (first, second, or third) to agree with their subjects. For example, the rule will flag the sentence, "The index are full of errors," because the subject, "index," is singular and does not agree with the plural verb, "are."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplySwedishGender property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSWEDISHGENDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyswedishgendervalue = [objectreference].ApplySwedishGender

[objectreference].ApplySwedishGender = applyswedishgendervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplySwedishNegation property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSWEDISHNEGATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applyswedishnegationvalue = [objectreference].ApplySwedishNegation

[objectreference].ApplySwedishNegation = applyswedishnegationvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplySwedishUsage property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYSWEDISHUSAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applyswedishusagevalue = [objectreference].ApplySwedishUsage

[objectreference].ApplySwedishUsage = applyswedishusagevalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyTrite property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYTRITE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applytritevalue = [objectreference].ApplyTrite

[objectreference].ApplyTrite = applytritevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyTwoGender property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYTWOGENDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applytwogendervalue = [objectreference].ApplyTwoGender

[objectreference].ApplyTwoGender = applytwogendervalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.every language.

## **Word Pro: ApplyTypicalMisspell property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYTYPICALMISSPELL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applytypicalmisspellvalue = [objectreference].ApplyTypicalMisspell

[objectreference].ApplyTypicalMisspell = applytypicalmisspellvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyUnGrammaticalExpressions property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYUNGRAMMATICALEXPRESSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking nongrammatical words and expressions.

### Data Type

[Integer](#)

### Syntax

applyungrammaticalexpressionsvalue = [objectreference].ApplyUnGrammaticalExpressions

[objectreference].ApplyUnGrammaticalExpressions = applyungrammaticalexpressionsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Ungrammatical expressions" in the "Rule type" field on the Rules panel.

This rule flags phrases considered ungrammatical in standard English, although they may occur in regional dialects (for example, "seeing as how" instead of "since").

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyVagueQuantifiers property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYVAGUEQUANTIFIERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule for checking vague quantifiers on or off .

### Data Type

[Integer](#)

### Syntax

applyvaguequantifiersvalue = [objectreference].ApplyVagueQuantifiers [objectreference].ApplyVagueQuantifiers = applyvaguequantifiersvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Vague quantifiers" in the "Rule type" field on the Rules panel.

This rule flags vague, wordy, or informal quantifiers (words or phrases that specify number or amount). For example, the expression, "lots of," can be replaced by a less colloquial term, such as "much," or "many," or a more precise indication of the amount.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyVerbGroupConsistency property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYVERBGROUPCONSISTENCY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking verb group consistency.

### Data Type

[Integer](#)

### Syntax

applyverbgroupconsistencyvalue = [objectreference].ApplyVerbGroupConsistency

[objectreference].ApplyVerbGroupConsistency = applyverbgroupconsistencyvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check.

Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Verb group consistency errors" in the "Rule type" field on the Rules panel.

This rule checks the forms of verbs. It identifies errors in the use of the present, the past, and the past participle, as well as errors in the choice of helping verbs. For example, the rule will flag the sentence, "We could not have drove home that night." It notes that "drove" is the simple past form of the verb and explains that "have" should be followed by the past participle, "driven."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyWeakModifiers property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWEAKMODIFIERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking weak modifiers.

### Data Type

[Integer](#)

### Syntax

applyweakmodifiersvalue = [objectreference].ApplyWeakModifiers

[objectreference].ApplyWeakModifiers = applyweakmodifiersvaluev

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Weak modifiers" in the "Rule type" field on the Rules panel.

This rule flags overused or colloquial modifiers (adjectives or adverbs that limit or modify the sense of another word). Expressions, such as "funny," "pretty well," or "nice," can often be replaced by more specific expressions.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyWordChoice property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWORDCHOICE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applywordchoicevalue = [objectreference].ApplyWordChoice

[objectreference].ApplyWordChoice = applywordchoicevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyWordCompoundingCheck property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWORDCOMPOUNDINGCHECK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking word compounding errors.

### Data Type

[Integer](#)

### Syntax

applywordcompoundingcheckvalue = [objectreference].ApplyWordCompoundingCheck

[objectreference].ApplyWordCompoundingCheck = applywordcompoundingcheckvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Compounding errors" in the "Rule type" field on the Rules panel.

This rule flags compounds with a missing or superfluous linking element or hyphen (for example, words such as "blue-print" that should be written as one word, "blueprint").

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyWordConfusion property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWORDCONFUSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applywordconfusionvalue = [objectreference].ApplyWordConfusion

[objectreference].ApplyWordConfusion = applywordconfusionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## **Word Pro: ApplyWordGender property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWORDGENDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### **Data Type**

[Integer](#)

### **Syntax**

applywordgendervalue = [objectreference].ApplyWordGender

[objectreference].ApplyWordGender = applywordgendervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.



## Word Pro: ApplyWordParts property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWORDPARTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the specific Grammar Check rule on or off.

### Data Type

[Integer](#)

### Syntax

applywordpartsvalue = [objectreference].ApplyWordParts

[objectreference].ApplyWordParts = applywordpartsvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting the appropriate rule in the "Rule type" field on the Rules panel.

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

## Word Pro: ApplyWordyPhraseCheck property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_APPLYWORDYPHRASECHECK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the Grammar Check rule on or off for checking phrases that may be wordy and/or awkward.

### Data Type

[Integer](#)

### Syntax

applywordyphrasecheckvalue = [objectreference].ApplyWordyPhraseCheck

[objectreference].ApplyWordyPhraseCheck = applywordyphrasecheckvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this property when you are setting the corresponding rule for proofing the document in Grammar Check. Equivalent to choosing Edit - Check Grammar, clicking Options, and selecting "Wordy expressions" in the "Rule type" field on the Rules panel.

This rule flags vague or wordy expressions that can be replaced by simpler, more direct expressions to make the writing clearer. For example, the phrase, "in all probability," might be replaced by the adverb, "probably."

**Note** This rule is not applicable in every language. Some Apply properties can only be found in specific language versions of Word Pro.

### **Word Pro: AreDisabledIconsGrayed property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AREDISABLEDICONSGRAYED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) Indicates whether an icon is enabled or disabled. If disabled, the icon will appear grayed. Before an icon can be enabled or disabled, you must first select it, using either the SelectStandardIcon or SelectCustomIcon method.

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].AreDisabledIconsGrayed = aredisablediconsgrayedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

If the constant is True (-1), the icon is enabled and will not be grayed. If the constant is False (0), the icon is disabled and is grayed.

## **Word Pro: Ascent property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ASCENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Points](#)

### **Syntax**

ascentvalue = [objectreference].Ascent

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: AsciiCRLFType property**

{button ,AL('H\_FILTER\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ASCII\_CRLF\_TYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

AsciiLineEnding

### **Syntax**

asciicrlfypsvalue = [objectreference].AsciiCRLFType

[objectreference].AsciiCRLFType = asciicrlfypevalue

### **Legal values**

\$LwpAsciiLineEndingCharset7bit (16)

\$LwpAsciiLineEndingCharset8bit (17)

\$LwpAsciiLineEndingCharsetPlatform (18)

### **Usage**

### **Word Pro: AtBeginningOfLine property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATBEGINNINGOFLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atbeginningoflinevalue = [objectreference].AtBeginningOfLine

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: AtBeginningOfObject property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATBEGINNINGOFOBJECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

atbeginningofobjectvalue = [objectreference].AtBeginningOfObject

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: AtBeginningOfParagraph property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ATBEGINNINGOFPARAGRAPH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atbeginningofparagraphvalue = [objectreference].AtBeginningOfParagraph

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



**Word Pro: AtBeginningOfStream property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ATBEGINNINGOFSTREAM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

atbeginningofstreamvalue = [objectreference].AtBeginningOfStream

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

### **Word Pro: AtBeginningOfWord property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATBEGINNINGOFWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atbeginningofwordvalue = [objectreference].AtBeginningOfWord

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: AtBeginning property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATBEGINNING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not the insertion point is at the beginning of a text stream.

**Data Type**

[Integer](#)

**Syntax**

atbeginningvalue = [objectreference].AtBeginning

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

### **Word Pro: AtEndOfLine property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ATENDOFLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atendoffinevalue = [objectreference].AtEndOfLine

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: AtEndOfObject property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATENDOFOBJECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

atendofobjectvalue = [objectreference].AtEndOfObject

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: AtEndOfParagraph property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ATENDOFPARAGRAPH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atendofparagraphvalue = [objectreference].AtEndOfParagraph

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: AtEndOfStream property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ATENDOFSTREAM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

atendofstreamvalue = [objectreference].AtEndOfStream()

**Legal values**

True False

**Usage**

### **Word Pro: AtEndOfWord property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATENDOFWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atendofwordvalue = [objectreference].AtEndOfWord

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



### **Word Pro: AtEnd property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ATEND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

atendvalue = [objectreference].AtEnd

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: AttrStyleName property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ATTRSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the attribute style from which you get the attribute information for a font. Word Pro uses the same style name for attributes as for a character or group of characters. Bold, Italic, Underline, and so on, are part of this style.

### **Data Type**

String

### **Syntax**

attrstylevalue = [objectreference].AttrStyleName

[objectreference].AttrStyleName = attrstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: AuthorName property**

{button ,AL('H\_DOCINFO\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTHORNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

[DocInfo]

Returns the name of the assigned editor who created the document.

[Version]

### **Data Type**

String

### **Syntax**

authornamevalue = [objectreference].AuthorName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to the User name value, which is located in the Personal panel of the Word Pro Preferences dialog box.

## **Word Pro: AutoBackup property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTOBACKUP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if automatic backup of documents is enabled.

### **Data Type**

[Integer\(bool\)](#)

### **Syntax**

autobackupvalue = [objectreference].AutoBackup

[objectreference].AutoBackup = autobackupvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the "File saving options" on the General panel of the Word Pro Preferences dialog box. If the legal value for this property is False, Word Pro does not automatically back up documents.

**Word Pro: AutoHyphenate property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTOHYPHENATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

autohyphenatevalue = [objectreference].AutoHyphenate

[objectreference].AutoHyphenate = autohyphenatevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: AutomaticLink property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTOMATICLINK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

automaticlinkvalue = [objectreference].AutomaticLink

[objectreference].AutomaticLink = automaticlinkvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: AutoSaveMinutes property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTOSAVEMINUTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the automatic time save interval in minutes.

### **Data Type**

[Integer](#)

### **Syntax**

autosaveminutesvalue = [objectreference].AutoSaveMinutes

[objectreference].AutoSaveMinutes = autosaveminutesvalue

### **Legal values**

Default is 10 minutes.

### **Usage**

Only use this property if the legal value of the AutoSave is True. Equivalent to the "minutes" edit box on the General panel of the Word Pro Preferences dialog box.

## **Word Pro: AutoSave property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTOSAVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if automatic time save is enabled.

### **Data Type**

Integer(bool)

### **Syntax**

autosavevalue = [objectreference].AutoSave

[objectreference].AutoSave = autosavevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "File saving options" on the General panel of the Word Pro Preferences dialog box. If the legal value for this property is False, Word Pro does not automatically time save documents.



## Word Pro: AutoVersion property

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_AUTOVERSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines when a version is automatically created. This property is usually set to occur when the file opens.

### Data Type

Variant (Enumerated)

AutoVersion

### Syntax

autoversionvalue = [objectreference].AutoVersion

[objectreference].AutoVersion = autoversionvalue

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpAutoVersionEveryday (29)	Creates a new version and reuses that version until the file is opened on the following day. If you leave a file open until the next day, Word Pro does not create a new version until you close and reopen the file.
\$LwpAutoVersionEveryweek (30)	Creates a new version and reuses that version until the file is opened after the following Monday. If you leave a file open until Monday, Word Pro does not create a new version until you close and reopen the file. In Word Pro, Monday is the first day of the week.
\$LwpAutoVersionNone (27)	Turns off the AutoVersion property.
\$LwpAutoVersionOnfileopen (28)	Creates a new version every time someone opens the file.
\$LwpAutoVersionOnfilesave (31)	Creates a new version every time someone saves the file.

### Usage

Equivalent to choosing File - Versions, selecting "Auto versioning," and specifying when auto versioning should occur (every day, every week, and so on).

## **Word Pro: BackgroundPrintingOn property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BACKGROUNDPRINTINGON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

backgroundprintingonvalue = [objectreference].BackgroundPrintingOn

[objectreference].BackgroundPrintingOn = backgroundprintingonvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: BackupPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BACKUPPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the default backup path (drive and directory) for Word Pro documents.

### **Data Type**

[String](#)

### **Syntax**

backuppathvalue = [objectreference].BackupPath

[objectreference].BackupPath = backuppathvalue

### **Legal values**

### **Usage**

Equivalent to the "Backups" field on the Locations panel of the Word Pro Preferences dialog box.

## Word Pro: BaseLineOffset property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_BASELINEOFFSET_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set or retrieve a layout object's position, relative to the text baseline.

### Data Type

Long

### Syntax

baselineoffsetvalue = [objectreference].BaseLineOffset

[objectreference].BaseLineOffset = baselineoffsetvalue

### Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

[FrameLayout]

Use this property when the frame layout object's placement is set to "In Text".

## Word Pro: BinName property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_BINNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The name of the bin from which the paper should be taken when printing a page layout.

### Data Type

String

### Syntax

binnamevalue = [objectreference].BinName

[objectreference].BinName = binnamevalue

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage

The BinName property is used only for page layouts. It corresponds to the "Printer bin" setting on the Size & Margins panel of the Page layout InfoBox.

If you set this property to an invalid bin name, Word Pro will use the current printer's default bin selection. You can check for valid BinName values by accessing the [BinNames](#) collection.

**Word Pro: BlockPaint property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BLOCKPAINT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

**Data Type**

[Integer](#)

**Syntax**

[objectreference].BlockPaint = blockpaintvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: Blue property**

{button ,AL('H\_COLOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BLUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The blue component of a color.

**Data Type**

[Integer](#)

**Syntax**

bluevalue = [objectreference].Blue

[objectreference].Blue = bluevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

You can set or read the current level of blue in a specific object's color.

**Word Pro: BodyOnly property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BODYONLY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

bodyonlyvalue = [objectreference].BodyOnly

[objectreference].BodyOnly = bodyonlyvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## **Word Pro: Bold property**

{button ,AL(`H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_BOLD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

boldvalue = [objectreference].Bold

[objectreference].Bold = boldvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: BorderOffset property

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_BORDEROFFSET_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to change the settings to the origin of a border in a layout object.

## Data Type

Long

## Syntax

borderoffsetvalue = [objectreference].BorderOffset

[objectreference].BorderOffset = borderoffsetvalue

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

[PageLayout]

[HeaderLayout]

[FooterLayout]

[CellLayout]

[FrameLayout]

[TableLayout]

## **Word Pro: BorderStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BORDERSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

borderstylevalue = [objectreference].BorderStyleName

[objectreference].BorderStyleName = borderstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: BottomExternalMargin property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_BOTTOMEXTERNALMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the amount of margin space that is present beneath a layout object.

### Data Type

Long

### Syntax

bottomexternalmarginvalue = [objectreference].BottomExternalMargin

[objectreference].BottomExternalMargin = bottomexternalmarginvalue

### Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

[FrameLayout]

This property cannot be set individually for frame layout objects within Word Pro. It is combined with all external margin values in the "Padding around border" setting, located on the Size & Margins panel of the InfoBox.

## Word Pro: BottomIntArea property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_BOTTOMINTAREA_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Not implemented.

The distance from the outer edge of a layout's border to the margin for that layout.

## Data Type

Long

## Syntax

bottomintareavalue = [objectreference].BottomIntArea

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

## **Word Pro: BreaksStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BREAKSSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

breaksstylevalue = [objectreference].BreaksStyleName

[objectreference].BreaksStyleName = breaksstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: BulletFonts property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BULLETFONTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores a list of decorative fonts (separated by commas) that are used to identify possible bullets during Format Check.

### **Data Type**

[String](#)

### **Syntax**

bulletfontvalue = [objectreference].BulletFonts

[objectreference].BulletFonts = bulletfontvalue

### **Legal values**

### **Usage**

When a user runs Format Check on a document and Word Pro finds a single character surrounded by white space, Word Pro tests to see if the character is a possible bullet. If the character is in this list, Word Pro assumes that it could be a bullet and runs further tests to confirm. You can use the SetArrayProp to add or delete fonts from the list, or you can use the GetArrayProp to read the fonts in the list.

## **Word Pro: BulletStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_BULLETSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

bulletstylevalue = [objectreference].BulletStyleName

[objectreference].BulletStyleName = bulletstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



**Word Pro: CanEditProperty property**

{button ,AL('H\_MAILROUTING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CANEDITPROPERTY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An internal flag that indicates if the route can be edited.

**Data Type**

[Integer](#)

**Syntax**

caneditpropertyvalue = [objectreference].CanEditProperty

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

The document must be in the middle of the route at a preordained stop for this property to function.

**Word Pro: CanEmbed property**

{button ,AL('H\_BASETABLE\_CLASS;H\_CONTENT\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FORMULA\_CLASS;H\_GLOSSARY\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS;H\_PARALLEL\_COLUMNS\_CLASS;H\_SUPERTABLE\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CANEMBED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not another object can be embedded into the current object.

**Data Type**

Integer

**Syntax**

canembedvalue = [objectreference].CanEmbed

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you may use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

A graphic object, for instance, will always contain a value of 0 in this property.

### **Word Pro: CanWePrint property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CANWEPRINT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

canweprintvalue = [objectreference].CanWePrint

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: Caption property

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_MENUITEM\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CAPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[MenuItem]

The name of a menu item that displays on a Word Pro menu.

[Window]

Text that displays on the title bar of a window.

[ApplicationWindow]

The name of the application window that displays in the title bar.

## Data Type

[String](#)

## Syntax

captionvalue = [objectreference].Caption

[objectreference].Caption = captionvalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

This property is not valid for IconBarManager or StatusBar.

[MenuItem]

Use this property to alter the name of a menu item that displays on a Word Pro menu. For example, you can write a script that changes a menu item's caption when a user triggers an event in Word Pro.

To display an accelerator key stroke caption, such as the Ctrl+S accelerator key stroke for Save, you must manually create a string. You can right-align the accelerator key stroke caption in the menu with a special tab character that you create by using this right alignment variable: `Spacer$ = Chr$(8)`.

You can see how to use this variable in the following example that shows how to write a Word Pro Save menu caption and its accelerator key stroke caption (Ctrl+S):

```
MyMenuName$ = "&Save" & Spacer$ & "Ctrl+S"
```

[ApplicationWindow]

Use this property to set or get the caption of the active application window. This property is not implemented for this class in Word Pro 97.

## **Word Pro: Case property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CASE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

Case

### **Syntax**

casevalue = [objectreference].Case

[objectreference].Case = casevalue

### **Legal values**

\$LtsCaseLower (1056964842)

\$LtsCaseMixed (1056964843)

\$LtsCaseUpper (1056964841)

\$LwpCaseDontcare (114)

\$LwpCaseInitcaps (115)

\$LwpCaseStyle (116)

### **Usage**

## **Word Pro: CellStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CELLSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

cellstylevalue = [objectreference].CellStyleName

[objectreference].CellStyleName = cellstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: CenteredHorz property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CENTEREDHORZ\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) For internal use only.

### Data Type

Integer

### Syntax

centeredhorzvalue = [objectreference].CenteredHorz

[objectreference].CenteredHorz = centeredhorzvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: CenteredVert property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_CENTEREDVERT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not the content of a layout is centered vertically.

## Data Type

[Integer](#)

## Syntax

centeredvertvalue = [objectreference].CenteredVert

[objectreference].CenteredVert = centeredvertvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

In Word Pro, this property is represented by the "Vertical alignment" setting, located on the Miscellaneous panel of the InfoBox for certain layout objects.



### **Word Pro: Changed property**

{button ,AL('H\_DIVISIONINFO\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHANGED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

changedvalue = [objectreference].Changed

[objectreference].Changed = changedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: CharacterBorderName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARACTERBORDERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

characterbordernamevalue = [objectreference].CharacterBorderName

[objectreference].CharacterBorderName = characterbordernamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: CharacterSetName property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARACTERSETNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

charactersetNamevalue = [objectreference].CharacterSetName

[objectreference].CharacterSetName = charactersetNamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: CharacterStyleName property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARACTERSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the character style at the insertion point.

**Data Type**

String

**Syntax**

characterstylevalue = [objectreference].CharacterStyleName

[objectreference].CharacterStyleName = characterstylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: CharSet property

{button ,AL('H\_CHARACTERSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHARSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Variant (Enumerated)

CharSet

### Syntax

charsetvalue = [objectreference].CharSet

[objectreference].CharSet = charsetvalue

### Legal values

\$LwpCharSetAnsi (119)

\$LwpCharSetAnsi1250 (120)

\$LwpCharSetAnsi1251 (121)

\$LwpCharSetAnsi1252 (122)

\$LwpCharSetAnsi1253 (123)

\$LwpCharSetAnsi1254 (124)

\$LwpCharSetArabic (134)

\$LwpCharSetAscii (118)

\$LwpCharSetBig5 (126)

\$LwpCharSetCp850 (139)

\$LwpCharSetCp932 (125)

\$LwpCharSetCp949 (127)

\$LwpCharSetCyrillic (133)

\$LwpCharSetGreek (135)

\$LwpCharSetHebrew (136)

\$LwpCharSetIsolatin1 (129)

\$LwpCharSetIsolatin2 (130)

\$LwpCharSetIsolatin3 (131)

\$LwpCharSetIsolatin4 (132)

\$LwpCharSetIsolatin5 (137)

\$LwpCharSetIsolatin6 (138)

\$LwpCharSetUnicode (128)

### Usage

## **Word Pro: Checked property**

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHECKED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if a menu item is checked or not checked.

### **Data Type**

[Integer](#)

### **Syntax**

checkedvalue = [objectreference].Checked

[objectreference].Checked = checkedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

To check a menu item, set the Checked property value to True. To uncheck a menu item, set the Checked property value to False.

Setting the Checked property is available only for custom created menu items. You cannot set the Checked property for predefined Word Pro menu items, because Word Pro dynamically sets the Checked property for predefined Word Pro menu items based on the current context.

**Word Pro: CheckForScriptEnumError property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHECKFORSCRIPTENUMERROR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is not implemented. Indicates if the Script Editor is set to check enumeration values.

**Data Type**

[Integer \(Bool\)](#)

**Syntax**

checkforscriptenumerrorvalue = [objectreference].CheckForScriptEnumError

[objectreference].CheckForScriptEnumError = checkforscriptenumerrorvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Provides a script writer with an additional level of error checking while developing a script. Default is False (0), which causes the Script Editor to check the validity of property values beyond checking for data type. If the value for the property is changed to True (-1), the Script Editor generates an error if the value of a property does not match one of the enumerated legal values defined for that property. Enumerated legal values always begin with "\$" and always have a numeric equivalent.

## **Word Pro: CheckForScriptPropertyError property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CHECKFORSCRIPTPROPERTYERROR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if the Script Editor is set to check property values.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

checkforscriptpropertyerrorvalue = [objectreference].CheckForScriptPropertyError

[objectreference].CheckForScriptPropertyError = checkforscriptpropertyerrorvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Provides a script writer with an additional level of error checking while developing a script. Default is False (0), which causes Word Pro not to check the value of a property after it is set. If the value for the property is changed to True (-1), the Script Editor checks the value of a property after the property is set. It does this in order to confirm that the property value matches the value called for in the script. If, for some reason the value of the property does not match the value in the script statement, the Script Editor raises the error, 7009 LWPERR SetPropFailed. You can include an OnError routine in your script to handle this error, or simply use this extra level of checking as a debugging tool.



## **Word Pro: CityState property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CITYSTATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

citystatevalue = [objectreference].CityState

[objectreference].CityState = citystatevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: ClassName property

```
{button ,AL(^H_BASSETABLE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_CONTENT_CLASS;H_DIVISIONINFO_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNOTETABLE_CLASS;H_FORMULA_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GLOSSARY_CLASS;H_GRAPHIC_CLASS;H_GRAPHICOLEBJECT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_OLEOBJECT_CLASS;H_PAGELAYOUT_CLASS;H_PARALLELCOLUMNS_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLE_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_CLASSNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The type of layout or content object from which you called this property.

## Data Type

String

## Syntax

classnamevariable = [objectreference].ClassName

## Legal values

## Usage

You can check this property to see what type of object you are currently accessing. For example, the ClassName property of a frame layout object would contain a string value of "Frame."

### **Word Pro: CleanScreenMode property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEANSCREENMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the value to indicate if the current window is in Clean Screen mode.

### **Data Type**

[Integer\(Bool\)](#)

### **Syntax**

cleanscreenmodevalue = [objectreference].CleanScreenMode

[objectreference].CleanScreenMode = cleanscreenmodevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

If the legal value for this property is False, Word Pro is not in Clean Screen mode.

## **Word Pro: CleanScreenOnStartUp property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLEANSCREENONSTARTUP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if Clean Screen mode is enabled when starting Word Pro.

### **Data Type**

[Integer\(Bool\)](#)

### **Syntax**

cleanscreenonstartupvalue = [objectreference].CleanScreenOnStartUp

[objectreference].CleanScreenOnStartUp = cleanscreenonstartupvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the "Load in clean screen view" option on the General panel of the Word Pro Preferences dialog box. If the legal value for this property is False, Word Pro does not automatically load in clean screen view.

**Word Pro: ClickHerePrompts property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLICKHEREPROMPTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

clickherepromptsvalue = [objectreference].ClickHerePrompts

[objectreference].ClickHerePrompts = clickherepromptsvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: Action property**

```
{button ,AL(##H_CLICKHERE_CLASS;H_MENUITEM_CLASS;H_CLICKHERE_CLASS;H_MENUITEM_CLASS',0)}  
  See list of classes
```

```
{button ,AL('H_CLICKHERE_ACTION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates how a ClickHere block behaves.

### **Data Type**

Integer

### **Syntax**

actionvalue = [objectreference].Action

[objectreference].Action = actionvalue

### **Legal values**

Must be an integer from 1 to 10.

- 1 = Text - plain text block
- 2 = Table - create table dialog
- 3 = Picture - Import Picture dialog
- 4 = OleObject - Insert OLE Object dialog
- 5 = Chart - creates a chart
- 6 = Drawing - creates a drawing
- 7 = File - Insert File dialog
- 8 = Glossary - Insert glossary Item dialog
- 9 = Equation - creates an equation frame
- 10 = InternetLink - Stores a URL which is opened when the user clicks on the ClickHere block

### **Usage**

## Word Pro: ClientHeight property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLIENTHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the height of the content area within the margins of a container.

## Data Type

Long

## Syntax

clientheightvalue = [objectreference].ClientHeight

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

## Word Pro: ClientWidth property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLIENTWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the width of the content area within the margins of a container.

## Data Type

Long

## Syntax

clientwidthvalue = [objectreference].ClientWidth

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage



## **Word Pro: ClientWndHeight property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLIENTWNDHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

clientwndheightvalue = [objectreference].ClientWndHeight

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: ClientWndWidth property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLIENTWNDWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Long](#)

### **Syntax**

clientwndwidthvalue = [objectreference].ClientWndWidth

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: CloseDocMacroName property**

{button ,AL('H\_AUTORUNMACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CLOSEDOCMACRONAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

closedocmacrovalue = [objectreference].CloseDocMacroName

[objectreference].CloseDocMacroName = closedocmacrovalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: CodePage property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CODEPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

codepagevalue = [objectreference].CodePage

[objectreference].CodePage = codepagevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: Collapsible property**

{button ,AL('H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COLLAPSIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

collapsiblevalue = [objectreference].Collapsible

[objectreference].Collapsible = collapsiblevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: Collate property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COLLATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prints an entire copy of the document before printing the next copy. You can only use this property if the document contains multiple pages.

### **Data Type**

Integer

### **Syntax**

collatevalue = [objectreference].Collate

[objectreference].Collate = collatevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This property determines whether or not a the print job for a document is collated.

**Word Pro: ColorOverride property**

{button ,AL('H\_NUMERICFORMATSUBSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COLOROVERRIDE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

coloroverridevalue = [objectreference].ColorOverride

[objectreference].ColorOverride = coloroverridevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: ColumnBalance property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_COLUMNBALANCE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-Write) Indicates whether or not text is balanced in each column of certain layout objects.

## Data Type

[Integer](#)

## Syntax

columnbalancevalue = [objectreference].ColumnBalance

[objectreference].ColumnBalance = columnbalancevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

In Word Pro, this property is represented by the "Column balance" setting on the Columns panel of the InfoBox for certain layout objects.



## Word Pro: ColumnGap property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN_CLASS;H_COLUMNGROUP  
LAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLA  
SS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAM  
ELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAY  
OUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLA  
YOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADIN  
GLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_COLUMNGAP_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The distance between one newspaper column and the next newspaper column to the right.

## Data Type

Long

## Syntax

columngapvalue = [objectreference].ColumnGap

[objectreference].ColumnGap = columngapvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

Equivalent to the "Space between columns" setting, located on the Columns panel of the InfoBox for certain layout objects.

### **Word Pro: ColumnNumber property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COLUMNNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The number of the page column in which the insertion point is located.

### **Data Type**

[Integer](#)

### **Syntax**

columnnumbervariable = [objectreference].ColumnNumber

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

This property does not refer to table columns or parallel columns. Instead, it refers to the standard columns, such as those found in a page layout.

**Word Pro: ColumnWidth property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COLUMNWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The width of the column in which the insertion point is located. Does not refer to a parallel or table column.

**Data Type**

Long

**Syntax**

columnwidthvalue = [objectreference].ColumnWidth

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

## **Word Pro: Company property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COMPANY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

companyvalue = [objectreference].Company

[objectreference].Company = companyvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: ConditionType property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_CONDITIONTYPE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines when a specific layout object is used.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

[objectreference].ConditionType = conditiontypevalue

conditiontypevalue = [objectreference].ConditionType

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpConditionTypeAllbutspecificpage (156)	Setting this value inserts a certain layout object in all pages, except a specific page in a document.
\$LwpConditionTypeAllpages (154)	Setting this value inserts a certain layout object in all pages of a document.
\$LwpConditionTypeOnlyevenpages (157)	Setting this value inserts a certain layout object in even pages of a document.
\$LwpConditionTypeOnlyoddpages (158)	Setting this value inserts a certain layout object in odd pages of a document.
\$LwpConditionTypeOnlyspecificpage (155)	Setting this value inserts a certain layout object in specific pages of a document.
\$LwpConditionTypeStartatpage (159)	Setting this value specifies on which page to insert a certain layout object.

## Usage

Use this property in conjunction with other properties, such as PageToUseLayoutOn. If the parent layout is not a page, then a layout uses the parent's ConditionType.

## **Word Pro: Condition property**

{button ,AL('H\_USEWHEN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONDITION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

ConditionType

### **Syntax**

conditionvalue = [objectreference].Condition

[objectreference].Condition = conditionvalue

### **Legal values**

\$LwpConditionTypeAllbutspecificpage (156)

\$LwpConditionTypeAllpages (154)

\$LwpConditionTypeOnlyevenpages (157)

\$LwpConditionTypeOnlyoddpages (158)

\$LwpConditionTypeOnlyspecificpage (155)

\$LwpConditionTypeStartatpage (159)

### **Usage**

**Word Pro: ContactUponPermissionDenied property**

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTACTUPONPERMISSIONDENIED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the person to contact if access to a document is denied due to TeamSecurity protection.

**Data Type**

[String](#)

**Syntax**

contactuponpermissiondeniedvalue = [objectreference].ContactUponPermissionDenied

**Legal values****Usage**

This property contains the user name found on the Access panel of the TeamSecurity dialog box, under the section "Who can open this dialog box and change access, editing rights, and other protection options." If the user selects "Only," the user name selected in the list box appears in this property.

Use this property to track the name of the person who should be contacted in the event a document cannot be opened due to TeamSecurity protection.

## Word Pro: ContentHeight property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTENTHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the height of the content based on its rotation within a container.

## Data Type

Long

## Syntax

contentheightvalue = [objectreference].ContentHeight

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage



'Example: BaseLineOffset property

'This example script has not yet been created.

'Example: BaseTable property

'This example script has not yet been created.

```
'Example: BeginChange method
' This example creates a table with 5 columns and 5 rows. Several background
' table cell properties are changed all at one using the BeginChange and
' EndChange methods.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ExampleTable As Table
Dim ExampleCell As CellLayout

.CreateTable False, "Default Table", 5,5
Set ExampleTable = .Table
ExampleTable.CellLayout(1,0).GotoLayout

Set ExampleCell = ExampleTable.CellLayout(1,0)
```

**.BeginChange**

```
ExampleCell.Content.InsertText "Hello"
With ExampleCell.BackGround
    .Pattern = $LtsFillSolid
    .Color.Red = 255
    .Color.Blue = 194
    .Color.Green = 255
    .Color.Override = $LwpColorOverrideRgb
    .BackColor.Red = 65
    .BackColor.Blue = 176
    .BackColor.Green = 0
End With

.EndChange
```

'Example: BeginCustomLines method

'This example script has not yet been created.

'Example: BinNames property

'This example script has not yet been created.

'Example: BinName property

'This example script has not yet been created.

'Example: Bisection method

'This example script has not yet been created.

'Example: BlockPaint property

'This example script has not yet been created.



'Example: Blue property

'This example script has not yet been created.

'Example: BodyOnly property

'This example script has not yet been created.

'Example: Bold method

' This example first inserts sample text in the current document and selects  
' the paragraph. The script then uses the Bold method to toggle the bold  
' attribute.

' RUNTIME DEPENDENCIES: You must have a document open with selected text  
' for this script to work.

.Text.InsertText "This is some sample text."  
.SelectParagraph

**.Bold**

'Example: Bold property

'This example script has not yet been created.

'Example: BookmarkManager property  
'This example script has not yet been created.

'Example: BookmarksByMarkerName property

'This example script has not yet been created.

'Example: Bookmarks property

'This example script has not yet been created.

'Example: BorderLines property

'This example script has not yet been created.



'Example: BorderRadius property

'This example script has not yet been created.

'Example: BorderStyleName property  
'This example script has not yet been created.

'Example: BottomBorder property

'This example script has not yet been created.

'Example: BottomExternalMargin property

'This example script has not yet been created.

'Example: BottomIntArea property

'This example script has not yet been created.

'Example: BreakLink method

'This example script has not yet been created.

'Example: BreaksStyleName property  
'This example script has not yet been created.

'Example: Breaks property

'This example script has not yet been created.



```
'Example: BringFrameToFrontOne method
' This example creates two frames and changes the order of the layering
' for the two frames.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewFrame 3285, 1200, 1575, 1830
.Frame.Layout.Background.Color.Red = 82
.Frame.Layout.Background.Color.Blue = 239
.Frame.Layout.Background.Color.Green = 145
.Frame.Layout.Background.Override = $LwpColorOverrideRgb
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent

.Deselect

.NewFrame 5285, 2200, 1575, 1830
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent
.Frame.Layout.Background.Color.Red = 182
.Frame.Layout.Background.Color.Blue = 139
.Frame.Layout.Background.Color.Green = 45
.Frame.Layout.Background.Override = $LwpColorOverrideRgb

MessageBox "Click OK to send frame to back. ",MB_OK,"Example Script"
.SendFrameToBack

MessageBox "Click OK to bring frame to front. ",MB_OK,"Example Script"
.BringFrameToFrontOne
```

```
'Example: BringFrameToFront method
' This example creates two frames and changes the order of the layering
' for the two frames.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewFrame 3285, 1200, 1575, 1830
.Frame.Layout.Background.Color.Red = 82
.Frame.Layout.Background.Color.Blue = 239
.Frame.Layout.Background.Color.Green = 145
.Frame.Layout.Background.Color.Override = $LwpColorOverrideRgb
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent

.Deselect

.NewFrame 5285, 2200, 1575, 1830
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent
.Frame.Layout.Background.Color.Red = 182
.Frame.Layout.Background.Color.Blue = 139
.Frame.Layout.Background.Color.Green = 45
.Frame.Layout.Background.Color.Override = $LwpColorOverrideRgb

MessageBox "Click OK to send frame to back. ",MB_OK,"Example Script"
.SendFrameToBack

MessageBox "Click OK to bring frame to front. ",MB_OK,"Example Script"
.BringFrameToFront
```

'Example: BulletFonts property

'This example script has not yet been created.

'Example: BulletStyleName property  
'This example script has not yet been created.

'Example: Bullet property

'This example script has not yet been created.

'Example: CalcSmartLevels method

'This example script has not yet been created.

'Example: CalculateSmartLevels method

' This example updates the SmartLevels for the currently active division.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CalculateSmartLevels

'Example: Cancel method

'This example script has not yet been created.



'Example: CanEditProperty property

'This example script has not yet been created.

'Example: CanEmbed property

'This example script has not yet been created.

```
'Example: CanHaveFootnotes method
' This example determines whether the current table cell can contain
' footnotes.
' The result is printed to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: A table cell must be selected for this script to work.

Print .Cell.CanHaveFootNotes
```

'Example: CanWePrint property

'This example script has not yet been created.

```
'Example: Caption property
Dim FileMenu As MenuItem
Set FileMenu = .ApplicationWindow.LWPMenuBar.Items("&File")
' Set a counter
Count% = 1

' Cycle through all options on the File menu
Forall Item In FileMenu.Items

' Display some info on each menu option
    Print Count% , "Name -", Item.Caption
    Print , "Action -", Item.Action
    Count% = Count% + 1
End Forall
```

```
'Example: CascadeWindow method
' This example creates two new documents based on the 'DEFAULT.MWP'
' SmartMaster.
' The script then prompts you to cascade the new windows.

.NewDocument , , "DEFAULT.MWP", ,
.NewDocument , , "DEFAULT.MWP", ,
MessageBox "Click OK to cascade the new windows.", MB_OK, "Example Script"

.CascadeWindow
```

```
'Example: Cascade method
' This example cascades any document windows.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
' Restore will restore the main Word Pro window
' Cascade the open document windows within Word Pro

.ApplicationWindow.Cascade
```

'Example: Case property

'This example script has not yet been created.



'Example: CellEngines property

'This example script has not yet been created.

'Example: CellEngine property

'This example script has not yet been created.

'Example: CellLayoutStyles property  
'This example script has not yet been created.

'Example: CellLayouts property

'This example script has not yet been created.

```
'Example: CellLayout method
' This example creates a table with 5 columns and 5 rows then inserts text and
' changes the background color for row 1, column 0.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim ExampleTable As Table
Dim ExampleCell As CellLayout

.CreateTable False, "Default Table", 5,5
Set ExampleTable = .Table
ExampleTable.CellLayout(1,0).GotoLayout

Set ExampleCell = ExampleTable.CellLayout(1,0)

ExampleCell.Content.InsertText "Hello"
With ExampleCell.BackGround
    .Pattern = $ItsFillSolid
    .Color.Red = 255
    .Color.Blue = 194
    .Color.Green = 255
    .Color.Override = $LwpColorOverrideRgb
    .BackColor.Red = 65
    .BackColor.Blue = 176
    .BackColor.Green = 0
End With
```

'Example: CellLayout property

'This example script has not yet been created.

```
'Example: CellRevert method
' This example creates a table with 5 columns and 5 rows then inserts text and
' changes the background color for row 1, column 0. The background color is
' then reverted after the message box is closed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ExampleTable As Table
Dim ExampleCell As CellLayout
```

```
.CreateTable False, "Default Table", 5,5
Set ExampleTable = .Table
ExampleTable.CellLayout(1,0).GotoLayout
```

```
Set ExampleCell = ExampleTable.CellLayout(1,0)
```

```
ExampleCell.Content.InsertText "Hello"
With ExampleCell.BackGround
    .Pattern = $LtsFillSolid
    .Color.Red = 255
    .Color.Blue = 194
    .Color.Green = 255
    .Color.Override = $LwpColorOverrideRgb
    .BackColor.Red = 65
    .BackColor.Blue = 176
    .BackColor.Green = 0
End With
```

```
Messagebox "Click OK to revert cell color change." ,MB_OK,"Example Script"
```

```
.CellRevert
```

'Example: CellStyleName property

'This example script has not yet been created.



'Example: Cell property

'This example script has not yet been created.

'Example: CenteredHorz property

'This example script has not yet been created.

'Example: CenteredVert property

'This example script has not yet been created.

```
'Example: ChangeAllEditsToEditor method
' This example changes all document edits and editing rights to the current
' editor.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim CurrentEditor As String
CurrentEditor = .Application.Preferences.UserName

.ActiveDocument.ChangeAllEditsToEditor CurrentEditor

'restrict editing of this document to the current editor
.ActiveDocument.DocControl.DocControlRestrictedToEditor = CurrentEditor
.ActiveDocument.DocControl.FileProtectionType = $LwpFileProtectTypeOrigAuthor
```

'Example: Changed property

'This example script has not yet been created.

'Example: ChangeSmartMaster method

' This example changes the current division's SmartMaster to BUSPLAN.MWP.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

SmartMasterPath = .ApplicationWindow.UserInterfacePrefs.StylePath

SmartMaster = SmartMasterPath & "\" & "BUSPLAN.MWP"

.ChangeSmartMaster SmartMaster, "Lotus Word Pro", ""

'Example: CharacterBorderName property

'This example script has not yet been created.

'Example: CharacterBorder property

'This example script has not yet been created.



'Example: CharacterSetName property  
'This example script has not yet been created.

'Example: CharacterSet property

'This example script has not yet been created.

'Example: CharacterStyleName property

'This example script has not yet been created.

'Example: CharacterStyles property

'This example script has not yet been created.

'Example: CharacterStyle property

'This example script has not yet been created.

'Example: CharSet property

'This example script has not yet been created.

```
'Example: Checked property
' This example toggles whether or not 'Example Menu' item is checked
' or unchecked. This script run from the 'NewItem' example and is not
' intended to be run stand-alone.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim NewMenu As MenuItem
Dim MenuName as String
```

```
MenuName = "&Example Menu"
Set NewMenu =.ApplicationWindow.LwpMenuBar.Items.Item("&File")
```

```
If NewMenu.Items(MenuName).Checked Then
    NewMenu.Items(MenuName).Checked = False
Else
    NewMenu.Items(MenuName).Checked = True
End If
```

'Example: CheckForScriptEnumError property  
'This example script has not yet been created.



'Example: CheckForScriptPropertyError property  
'This example script has not yet been created.

'Example: ChgLineStyle method

'This example script has not yet been created.

'Example: ChildLayouts property

'This example script has not yet been created.

'Example: CityState property

'This example script has not yet been created.

'Example: ClassName property

'This example script has not yet been created.

'Example: CleanScreenMode property

'This example script has not yet been created.

'Example: CleanScreenOnStartUp property

'This example script has not yet been created.

'Example: ClearAll method

' This example inserts 10 right aligned tabs with leader dots which are spaced  
' one half inch apart. After the message box is closed all of the tabs are  
' removed.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim TabNumber As Integer

For TabNumber = 1 To 10

    stat = .Text.TabRack.InsertOne(720 \* TabNumber, \$LwpTabTypeLeft, \$LwpTabLeaderDot,  
    \$LwpTabRelativeLeft, 32)

    .Type TabNumber & "[Tab]"

Next

.Text.TabRack.ClearAll



'Example: ClearDivisionList method

'This example script has not yet been created.

'Example: ClearInternalSpellInfo method

'This example script has not yet been created.

'Example: ClearParaRevisionTags method

' This example removes paragraph revision tags from the active document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ClearParaRevisionTags

'Example: ClearPopupData method

'This example script has not yet been created.

```
'Example: ClearSplits method
' This example sets a split view. It displays the same document in two
' windows. In the top window it displays the document at page width.
' In the bottom window, it displays the document in a multi-page view showing
' the first seven pages.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

' clear any current splits
.ActiveDocWindow.WinViewPrefs.ClearSplits

' the first window will display seven pages left to right
.ActiveDocWindow.WinViewPrefs.NumCols = 7
.ActiveDocWindow.WinViewPrefs.IsInDraft = False
.ActiveDocWindow.WinViewPrefs.ViewType = &H40

' now set a new window that will take 66% of the doc window
.ApplicationWindow.UserInterfacePrefs.VerticalSplitWindow = True
.ApplicationWindow.UserInterfacePrefs.SplitPercentage = 66

' open the new window
.NewWindow

' change the new window to display page width
.ApplicationWindow.UserInterfacePrefs.VerticalSplitWindow = False
.ActiveDocWindow.WinViewPrefs.IsInDraft = False
.ActiveDocWindow.WinViewPrefs.ViewType = &H100
```

'Example: ClearUpdate method

'This example script has not yet been created.

```
'Example: Clear method
' This example insert some text into the current document. After the message
' box is closed all text is cleared.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim x as Integer
For x = 1 To 5
    .Text.InsertText "Some Text " & x
Next
MessageBox "Click OK to remove all text.",MB_OK,"Example Script"
.Text.Clear $LwpClearWhatDefault
```

'Example: ClickHerePrompts property  
'This example script has not yet been created.



'Example: ClickHeres property

'This example script has not yet been created.

'Example: ClientHeight property

'This example script has not yet been created.

'Example: ClientWidth property

'This example script has not yet been created.

'Example: ClientWndHeight property

'This example script has not yet been created.

'Example: ClientWndWidth property

'This example script has not yet been created.

'Example: CloseAll method

' This example closes all open files including untitled files.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CloseAll \$LwpCloseFileIfLastdocOpenUntitled

'Example: CloseDocMacroName property

'This example script has not yet been created.

'Example: CloseDocWindow method

' This example creates a new document window based upon the current document  
' and displays a message box prompting you to close the window.  
' After you click OK, the script closes the new document.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewWindow

MessageBox "Click OK to close the current window.",MB\_OK,"Example Script"

**.CloseDocWindow**



'Example: CloseMergeDataFile method  
' This example merges data for the current merge document.  
' RUNTIME DEPENDENCIES: You must have a document open which has been assigned  
' to a merge data file has inserted merge fields for this script to work.

Dim stat As Integer

' Set up to merge and view  
.ApplicationWindow.ActiveDocument.MergeOptions.Options = &H2  
.ApplicationWindow.ActiveDocument.MergeOptions.MergeStepNumber =  
    \$LwpMergeStepNumber3  
.MergeStart

Do  
    stat = .Merge(\$LwpMergeActionNextRecord)  
    .Merge \$LwpMergeActionMergeOne  
Loop Until stat = False

.Merge \$LwpMergeActionClose

**.CloseMergeDataFile**

'Example: CloseObject method

'This example script has not yet been created.

'Example: Close method

' This example completely shuts down Word Pro while asking to save any changes.

.ApplicationWindow.Close

'Example: CodePage property

'This example script has not yet been created.

'Example: Collapsible property

'This example script has not yet been created.

'Example: Collate property

'This example script has not yet been created.

A special data construct which corresponds to a specific type of item in an application and defines what aspects of that item you can access through LotusScript and what you can do with that item. For example, the TextDocument class in Word Pro corresponds to a Word Pro document. The class definition includes variables for storing the name of the document and other information. The class definition also includes methods for performing specific tasks with a document. In short, the TextDocument class defines exactly how much of a Word Pro document you can access and what you can do with it.

When you open a document in Word Pro, Word Pro creates an instance of the TextDocument class. That instance of the TextDocument class is an object. You use that object to control the document you opened.

## **By Category**

Classes

Properties

Methods

Events



## Overview: Division names in LotusScript

In LotusScript, each division has two names: an external name and an internal name.

### External division names

The external name is the name you see displayed in the divider tab on your screen. This external name is read-write and does not have to be unique. While the external name is usually for display purposes only, you can use the external name to retrieve the internal name as described below. The external name is stored in the Name property on the DivisionInfo object.

This statement returns the external name for the current division:

```
.Division.DivisionInfo.Name
```

### Internal division names

The internal name is read-only and is unique for each division. It allows you to uniquely identify a division when you use methods to access and control that division. The internal name is stored in the Name property on each Division object.

This statement returns the internal name for the current division:

```
.Division.Name
```

### Getting the internal name of a division

Every method which allows you to control a division requires that you identify a division by its unique internal name. Getting the internal name for the currently active division is done with the following statement:

```
.Division.Name
```

However, to get the internal name of a division when you are not in that division, you first need to gain access to that division. You can gain access to any division by assigning that division to a variable, using the Bind statement.

The Bind statement allows you to assign an object to a variable using that object's name. In the case of a division, the Bind statement allows you to use the division's external name as shown in the following statements:

```
Dim mydivision As Division
Dim divisionname As String
Set mydivision = Bind("!Body")
divisionname = mydivision.Name
```

This example declares two variables (mydivision and divisionname). It then binds the division named "Body" to the mydivision variable. Then it assigns the value of the Name property on mydivision to the other variable (divisionname). You can then use the divisionname variable as the value for any parameter which refers to the Body division. For example, if you use the CombineDivisions method, you could use the divisionname variable as the value for either the StartName or EndName parameters.

An action, defined as part of a class, which returns specific information about an object. An event can be a keystroke, entering a layout, or any other action which Word Pro can detect. When an event occurs in Word Pro, we say the event was raised and the object involved in that event is said to have raised the event.

For example, the EnterLayout event is raised each time you move the insertion point to a different layout. You can write a script that will run each time this event is raised. If more than one object has a layout, you must attach the script to the EnterLayout event for each object.

A function, defined as part of a class, which performs a specific task with any object created from that class. Once an object is instantiated from a class, you can use that class' methods to access and manipulate the object.

For example, you can use the AddDivision method in a TextDocument object to add a division to a Word Pro document.

Acts as a link between your script and a specific part of an application or a document. All objects are created from class descriptions. The class acts as a blueprint or template for the object itself. You can create more than one object from the same class. If you create more than one object from the same class, you can access and control them independently of one another. Each object created from a class is said to be an instance of that class. When an object is created, we say the object has been "instantiated."

A kind of variable which is defined as part of a class. Once an object is instantiated from a class, these variables store information about the object, such as the object's name or location. Properties can also contain other objects instantiated from other classes.

For example, the `TextDocument` class defines a `FullName` property and a `LineNumberOptions` property. When you open a Word Pro document, Word Pro creates an object based on the `TextDocument` class. In addition to filling some properties with strings and numbers, Word Pro must create all the objects which are parts of the `TextDocument` object, such as `LineNumberOptions`, `PrintSettings`, and `SortOptions`. Each object is stored in one of the `TextDocument` object's properties, just as if it were a string or a number.

## Overview: The Word Pro LotusScript Object Model

Word Pro provides its own set of LotusScript language tools called classes. These classes define which parts of Word Pro you can access with LotusScript and what you can do with them. Each Lotus application that uses LotusScript has its own set of these "product-defined classes."

Each Word Pro class corresponds to a type of object in Word Pro, such as windows, frames, and pieces of text. Each type of object has its own Word Pro class. The class itself is like a blueprint for one type of object. The class only defines the parts of an object (called the properties), and the tools for manipulating or interacting with that type of object (called methods and events). When you run Word Pro, Word Pro automatically keeps track of all its active objects. Many objects, such as frames, are created from the same class. But every object has unique qualities, such as name and location, which allow you to control it independently of other objects created from the same class.

When you look at all the LotusScript classes defined in Word Pro, you see a kind of model of the Word Pro application and all its parts. Everything from the application window, to the individual documents, to the divisions and pages of those documents and the information they contain, is represented in LotusScript by an object. Every object has a corresponding class which defines what the object is and how you can interact with it. We call these classes the Word Pro object model because, when seen as a whole, the classes form a model of the objects which comprise Word Pro, its features and documents. When you write a script, you can access and control any part of Word Pro that is represented by a class in the Word Pro object model.

---

{button ,AL('H\_WP\_THE\_LOTUSSCRIPT\_LANGUAGE\_OVER;H\_USING\_LOTUSSCRIPT\_IN\_WORD\_PRO\_OVER'  
.0)} [See related topics](#)

## **Overview: Using LotusScript in Word Pro**

The [LotusScript language](#) works the same way in all Lotus applications. However, Word Pro gives you its own access to LotusScript, dialog boxes, and a script recorder. The following topics explain some of the basic concepts of how to use LotusScript effectively in Word Pro:

[Overview: Word Pro scripts](#)

[Overview: Startup scripts](#)

[Overview: Recording a script](#)

[Playing a script](#)

[Playing an Ami Pro 3.x macro](#)

[Displaying the Script Editor](#)

[Attaching a script to an icon](#)

[Attaching a script to a ClickHere block](#)

[Creating custom dialog boxes](#)

While using the Word Pro Reference Help, you may encounter terminology with which you are not familiar. For your convenience, many of these terms are defined and explained in the following topics:

[Overview: Word Pro LotusScript Object Model](#)

[Overview: Word Pro LotusScript Object References](#)

[Overview: Word Pro LotusScript Units Of Measurement](#)

[Overview: Word Pro LotusScript Collection Classes](#)

[Overview: Word Pro LotusScript Enumerated Values](#)

[Overview: Word Pro LotusScript Abstract Classes](#)

[Overview: Word Pro LotusScript Object Containment](#)

[Overview: Word Pro LotusScript Class Hierarchy and Inheritance](#)

## **Overview: Word Pro LotusScript Abstract Classes**

Word Pro uses a number of classes, such as BaseObject, Application, and Layout, as a starting point for a group of similar classes, or as a means of passing class members onto an entire group of classes. For example, the BaseObject class serves only one purpose in the Word Pro object model. It provides a set of six basic properties which are then inherited by every single class in the Word Pro object model. The Application class is an abstract class which provides a basic set of properties, methods, and events, which are shared by all Lotus applications. The Layout class provides a basic set of class members which is shared by all its derived classes, including PageLayout, FooterLayout, HeaderLayout, TableLayout, and FrameLayout.

The concept of using one class as the basis for a group of other classes is known as inheritance. When a class gets a set of class members from another class, it is said to inherit those class members. You can tell the origin of class members by looking at the Base Classes and Derived Classes headings in a class definition.



## **Overview: Word Pro LotusScript Class Hierarchy and Inheritance**

Classes based on other classes are said to be derived from the original class.

## **Overview: Word Pro LotusScript Collection Classes**

Word Pro helps you keep track of objects by grouping those objects together in collection objects. Each collection object in Word Pro corresponds to one of the Word Pro object classes. Each object in a collection is said to be an item of that collection. For example, the TextDocumentCollection object contains all objects instantiated from the TextDocument class. Each TextDocument object is said to be an item of the TextDocumentCollection object. Grouping objects together in this fashion makes it easier to locate and access one or more objects of a particular class.

### **The Scope of Collections**

Each collection object has a fixed scope which determines where the collection object derives its items. Most collections are limited to a particular division. For example, the FrameLayoutCollection object contains all the FrameLayout objects in a particular division in a document. Thus, if the document has three divisions, Word Pro maintains one FrameLayoutCollection for each division that contains a FrameLayout object.

To access an object through its collection, you must know which collection object contains the object(s) for which you are looking.

### **Accessing objects in a collection**

You can access objects in a collection in one of two ways:

- Iteration - Using the ForAll statement to access every object in turn by stepping through the entire collection.

- Indexing - Using the Item method or the indexing syntax to access one specific object in a collection.

In both iteration and indexing, you access the object(s) through the corresponding collection object.

## Overview: Word Pro LotusScript Enumerated Values

Many properties and methods in Word Pro make use of enumerated lists of values. These lists of values serve two purposes:

- They define the legal values for a property or method parameter.
- They provide textual and numeric constants for those legal values.

For example, the `AlignmentType` property has only nine legal values. They are represented in LotusScript by the following constants:

```
$LtsAlignmentSmart  
$LtsAlignmentLeft  
$LtsAlignmentRight  
$LtsAlignmentHorizCenter  
$LtsAlignmentJustify  
$LwpAlignmentTypeJustifyall  
$LwpAlignmentTypeNumericleft  
$LwpAlignmentTypeNumericright  
$LwpAlignmentTypeAlignRevert
```

Each constant corresponds to a different alignment setting for the `AlignmentType` property. However, these constants can only be used within the confines of LotusScript and they require more keystrokes than their numeric equivalents (which are seen in parentheses below):

```
$LtsAlignmentSmart (1056964612)  
$LtsAlignmentLeft (1056964609)  
$LtsAlignmentRight (1056964610)  
$LtsAlignmentHorizCenter (1056964611)  
$LtsAlignmentJustify (1056964613)  
$LwpAlignmentTypeJustifyall (5)  
$LwpAlignmentTypeNumericleft (6)  
$LwpAlignmentTypeNumericright (7)  
$LwpAlignmentTypeAlignRevert (8)
```

Depending on your needs and your preference, you can use either the textual constant or its numeric equivalent. If you call the property or method from a non-Lotus application, you must use the numeric constant.

The data type for a property or parameter which has an enumerated list of values is always `Variant`. This allows you to use either the textual constant or its numeric equivalent.

**Note** Many properties, method parameters, and method return values in the Word Pro LotusScript object model are limited to -1 or 0, and will accept the LotusScript constants, `True` and `False`, in place of the integers, -1 and 0. These are not enumerated values. They are Boolean expressions that have a data type of `Integer`. The constants, `True` and `False`, can be used anywhere to take the place of -1 and 0. In contrast, enumerated constants can only be used as values in those properties and parameters which list them as legal values.

## Bitmasks

Word Pro also employs a second kind of enumerated constant, called a "bitmask." Like the enumerated value described above, a bitmask limits the number of legal values for a property or a parameter. A bitmask also provides both a textual and a numeric constant that you can use interchangeably. However, in a bitmask, you can often combine the constants as a means of achieving a combined result in the property or parameter.

For example, a property might use a bitmask which defined four constants as seen below:

```
LwpStartOnNewPage (&H0)  
LwpStartOnCurrentPage (&H1)  
LwpIncludePrevHeader (&H20)  
LwpIncludePrevFooter (&H10)
```

You could use one of these constants to achieve one of the effects, or you could combine constants to get a combined result. For example, the following combination would start something on a new page and would include the previous header and footer:

LwpStartOnNewPage OR LwpIncludePrevHeader OR LwpIncludePrevFooter

We use the OR operator to combine bitmask constants. This combination of the hexadecimal equivalents can do the same thing:

&H0 OR &H20 OR &H10

Some bitmasks have constants which are mutually exclusive. This means they cannot be combined. In the example above, the first two constants would be mutually exclusive because the first constant starts something on a new page, while the second constant starts the same thing on the current page.

You cannot use the textual constants unless you include the contents of the file named "WPBITMSK.LSS" in the Declarations section of your script. WPBITMSK.LSS provides the link between the hexadecimal bitmask constant and its textual equivalent. Even if you include this file in the Declarations section, you cannot use the textual constant from another application through OLE automation. OLE automation always requires the use of the hexadecimal constant. Word Pro always returns the hexadecimal constant from properties that employ bitmasks.

To include the wpbitmask.lss file, place the following statement in the Declarations script of the !Globals object.

```
%include "wpbitmask.lss"
```

## Overview: Word Pro LotusScript Object Containment

In Word Pro, you will encounter many objects that are contained in the properties of other objects. For example, when you create a frame in Word Pro, that frame is represented in LotusScript by a FrameLayout object. That FrameLayout object has a number of properties. Some of those properties contain strings, or integers. Thus, when you want to set the name of the frame, you can use the following statement:

```
.Frame.Layout.Name = "BlueFrame"
```

But other properties on that FrameLayout object contain smaller objects which are part of the frame you created. For example, the Background property on your FrameLayout object contains a Background object. When you want to manipulate the background of your frame, you use that Background object. Thus, you can set the fill pattern for the background of the frame to a horizontal bar pattern, as follows:

```
.Frame.Layout.Background.Pattern = "$LtsFillHorizBar"
```

But the chain of containment doesn't end there. The Background object also has the Color and BackColor properties which contain Color objects. The Color object in the Color property lets you control the color of the fill pattern in the background of the frame. The Color object in the BackColor property lets you control the color of the null space behind the fill pattern. Thus, if you wanted to set the color of the horizontal bars to red and the space behind them to white, you can use the following code:

```
.Frame.Layout.Background.Color.SetRGB(255,0,0)
.Frame.Layout.Background.BackColor.SetRGB(0,0,0)
```

The Word Pro object model is structured so that many objects are comprised of smaller objects that are contained in the properties of the larger object. This containment of objects gives you much finer control over Word Pro and all its parts.

### The focus in contained objects

When an object has the focus, the object (or objects) which contain that object share the focus with that object. For example, when the insertion point is in a table cell, the cell has the focus. But the cell shares the focus with the table that contains the cell, and the WordPro object shares the focus with the Table object. This can affect which properties and methods will be available to you at any given time. Thus, you must bear in mind both the focus and the containment of an object when using leading dot notation to reference an object.

For example, if you use the leading dot to call the Parent property while the CellLayout object has the focus, you get the CellContainer object for that CellLayout. However, if you changed the focus by selecting the table and used the same statement, the leading dot would return the WPApplication object because the Parent property for the Table contains a reference to WPApplication.

## Overview: Word Pro LotusScript Object References

Each time Word Pro creates a LotusScript object, it uses that object's class to define exactly how much access you have to an object. How much of an object you can see and what you can do with that object is all determined by the class definition. The class defines properties to give you access to an object's attributes, methods to give you special tools for manipulating the object, and events which allow you to run certain scripts when an event occurs in a specific object. When you use LotusScript to access and manipulate the object, your actions are passed on to the part of the application represented by that object.

The syntax descriptions for properties and methods include "[objectreference.]" at the beginning of the syntax. When you use a property or method in a script statement, you must replace "[objectreference.]" with a reference to the object that contains or defines the property or method you are using.

For example, the syntax for the AlignmentType property is described as follows:

```
[objectreference].AlignmentType
```

The AlignmentType property is defined as part of the Alignment class. Therefore, the AlignmentType property is always part of an Alignment object. When you use the AlignmentType property in a statement, you must include a reference to its Alignment object as seen in the following statement.

```
.Text.Alignment.AlignmentType
```

The Alignment object is always contained by another object (such as Text). Therefore, you must include the containing object as part of your object reference. The amount of information required in an object reference depends on two factors:

- the object or objects which have focus
- the number of objects within the focus which match the object reference

### How the focus affects the object reference

The focus is usually defined as the place in an application which is currently active. For example, when your cursor is in a Word Pro document and that document is active, we say the document has the focus. When you save the document and the Save dialog box opens, we say the dialog box has the focus. There are even more subtle differences in focus. For example, when you move your cursor from the page of your document into a frame, you again change the focus. When you move from the frame to a table, you change the focus again. In each case, you may notice that the menus, SmartIcons, status bar, and even the cursor itself change, depending on what part of Word Pro has the focus. The same holds true when you are running a script.

If you try to use a Text object method while a picture has the focus, you may get an error. If your focus is on a table and you try to check a property on a frame, you will get an error. That is why you must always provide enough of an object reference to specify exactly what object you are trying to access. While it is difficult to define exactly how much of an object reference is required in every situation, there are a few simple rules to keep in mind when referencing an object in a script.

- If the object you are referencing will have the focus when you run the script, you can simply precede the property or method name with a dot (.). Word Pro always interprets a leading dot as the object with the current focus. For example, when a Text object has the focus, you can select the word which has the focus by using the Select method, without explicitly referencing the text object itself:

```
.Select (1749)
```

You can use a property using the same reference:

```
.SelectionHidden = True
```

- If you want to access an object that is contained as a part of another object which has the focus (such as a Font object on a Text object), you must provide the name of the property which contains the object before calling the property or method. For example, while a Text object has the focus, you can reference the Font object in the Font property in this way:

```
.Font.FontName = "Helvetica"
```

Once again, Word Pro sees the leading dot and uses the focus to determine the object reference. Note that the focus in an application can only be placed in certain areas. You cannot place the focus on a Font object; thus you must provide an object reference to the Font object as seen in the example above.

**Note** When an object is stored in the property of another object, we say the first object is contained by the second object. For more information about object containment, see [Overview: Word Pro LotusScript Object Containment](#).

- Another means of accessing objects which don't have the focus is through the WordPro object (created from WPAApplication class). The WordPro object provides direct or indirect access to nearly every part of the Word Pro application. Even when your focus is on a Text object, you can use the Application property on that Text

object to access the WordPro object, and thus the rest of the objects in Word Pro. For example, you can get to the background color of a frame even while the focus remains in a Text object, using the following syntax.

```
.Application.Divisions(itemreference).Foundry.Frames(itemreference).Background.BackColor.SetRGB (255,255,255)
```

Note the use of the properties, Divisions and Frames, in the example above. These properties contain collection objects, each of which serves as a storage area for a particular class of objects. Divisions has a data type of DivisionCollection and contains all the Division class objects in the active document. Frames has a data type of FrameLayoutCollection and contains all the FrameLayout class objects per division. See [Overview: Word Pro LotusScript Collection Classes](#) for more information on collections.

## **Overview: Word Pro LotusScript Units Of Measurement**

Word Pro uses a number of different units of measurement in its LotusScript object model. Some properties and methods accept inches, while others may accept points or twips or units.

The following list of equivalent values may be useful in determining the appropriate value to use.

### **Inch equivalents**

1 inch = 72 points

1 inch = 1440 twips

1 inch = 4718592 units

### **Point equivalents**

1 point = 1/72"

1 point = 20 twips

1 point = 65536 units

### **Twip equivalents**

1 twip = 1/1440"

1 twip = 1/20 point

1 twip = 3276.8 units

### **Unit equivalents**

1 unit = 1/4718592"

1 unit = 1/65536 point

1 unit = 1/3276.8 twips



## Overview: Word Pro Menu Command IDs

Below is the list of constants for the Word Pro menu command IDs. You can use either the text constant or the hexadecimal constant to identify a specific Word Pro menu command.

LwpMenuFilemenu	&H64	LwpMenuMIFieldauto	&HE0
LwpMenuMfNew	&H65	LwpMenuMIFieldshowrtl	&HE1
LwpMenuMfOpen	&H66	LwpMenuMIFielddoauto	&HE2
LwpMenuMfSave	&H67	LwpMenuMIFieldremove	&HE3
LwpMenuMfSaveas	&H68	LwpMenuMIExechand	&H23E
LwpMenuMfRevert	&H69	LwpMenuMIIndexopts	&HE4
LwpMenuMfAppendtext	&H6A	LwpMenuHelpmenu	&H226
LwpMenuMfImport	&H6B	LwpMenuMhIdxhelp	&H227
LwpMenuMfFilemanage	&H6C	LwpMenuMhAbout	&H228
LwpMenuMfDocdesc	&H6D	LwpMenuMhMacrohelp	&H229
LwpMenuMfCreatedatafile	&H6E	LwpMenuMhCompatible	&H22B
LwpMenuMfPrint	&H6F	LwpMenuMhUsingHelp	&H33E
LwpMenuMfChgprinter	&H70	LwpMenuMhKeyboard	&H33F
LwpMenuMfExit	&H71	LwpMenuMhHowDoI	&H340
LwpMenuMfPrintreset	&H72	LwpMenuMhDoHelp	&H341
LwpMenuMfDde	&H73	LwpMenuMhForUpgraders	&H342
LwpMenuMfMergeaction	&H74	LwpMenuMhTutorial	&H345
LwpMenuMfJustprint	&H75	LwpMenuMhSearch	&H36F
LwpMenuMfStandalonep	&H147	LwpMenuMsControl	&H259
LwpMenuMfAllfldsrename	&H76	LwpMenuMsFileman	&H25A
LwpMenuMfPrintwrongpaper	&H77	LwpMenuMtTblayout	&H272
LwpMenuMfPrintcancel	&H78	LwpMenuMtInscolorw	&H273
LwpMenuMfDummymrgsel	&H79	LwpMenuMtDelcolorw	&H274
LwpMenuMfDummymrgext	&H7A	LwpMenuMtEditfrm1a	&H275
LwpMenuMfAsciiopts	&H7B	LwpMenuMtTbladd	&H276
LwpMenuMfCanprint	&H7C	LwpMenuMtHeading	&H277
LwpMenuMfFldrename	&H7D	LwpMenuMtTableinfox	&H279
LwpMenuMfPrintopt	&H7E	LwpMenuMtSizecolorw	&H27A
LwpMenuMfSaveasnw	&H7F	LwpMenuMtConnect	&H27B
LwpMenuMfImportnw	&H80	LwpMenuMtSavefrm1a	&H27D
LwpMenuMfDocdescnw	&H81	LwpMenuMtQuickaddrow	&H283
LwpMenuMfMasternw	&H82	LwpMenuMtQuickaddcol	&H284
LwpMenuMfNwimport	&H83	LwpMenuMtDelcolumn	&H285
LwpMenuMfMergeviewprint	&H84	LwpMenuMtDelrow	&H286
LwpMenuMfLastopen1	&H85	LwpMenuMtInsertrow	&H287
LwpMenuMfLastopen2	&H86	LwpMenuMtInsertcol	&H288
LwpMenuMfLastopen3	&H87	LwpMenuMtInsertaccel	&H289
LwpMenuMfLastopen4	&H88	LwpMenuMtDeleteaccel	&H28A
LwpMenuMfLastopen5	&H89	LwpMenuMtDeltab1e	&H28B
LwpMenuMfOpennostyle	&H8A	LwpMenuMtSelectcolumn	&H28C
LwpMenuMfMergelabels	&H8B	LwpMenuMtSelectrow	&H28D
LwpMenuMfOtherflds	&H8C	LwpMenuMtSelecttable	&H28E
LwpMenuMfImportpicture	&H8D	LwpMenuMtChgicons1ze	&H28F

LwpMenuMfExport	&H8E	LwpMenuMtDisconnect	&H290
LwpMenuMfAttribnw	&H8F	LwpMenuMtSplitcells	&H291
LwpMenuMfCanmergeprint	&H92	LwpMenuMtSplittable	&H292
LwpMenuMfImporttext	&H98	LwpMenuMtPcolinscolrow	&H293
LwpMenuMfMergeprintopts	&H235	LwpMenuMtPcolinsertrow	&H294
LwpMenuMfClose	&H90	LwpMenuMtPcolinsertcol	&H295
LwpMenuMfCloseall	&H2DB	LwpMenuMtPcoldeltable	&H296
LwpMenuMfCloseallreplacelast	&H2DC	LwpMenuMtPcoldelcolumn	&H297
LwpMenuMfPassword	&H91	LwpMenuMtPcoldelrow	&H298
LwpMenuMfPassword2	&H122	LwpMenuMtPcolconnect	&H299
LwpMenuMfMasterdocument	&H2E0	LwpMenuMtPcoldisconnect	&H29A
LwpMenuMfClosefile	&H93	LwpMenuMtPcolsplitcells	&H29B
LwpMenuMfLock	&H94	LwpMenuMtPcolselectcolumn	&H29C
LwpMenuMfNextmdiwindow	&H2D7	LwpMenuMtPcolselectrow	&H29D
LwpMenuMfMail	&HE5	LwpMenuMtPcolselecttable	&H29E
LwpMenuMfMailNew	&H2E1	LwpMenuMtPcoldelcolrow	&H29F
LwpMenuMfMailread	&HE6	LwpMenuMtPcolheading	&H2A0
LwpMenuMfQuickopen	&HE7	LwpMenuMtCellinfoebox	&H2A1
LwpMenuMfFilesep	&HEB	LwpMenuMtShowtablelineinfoebox	&H2A2
LwpMenuMfSavecopyas	&H124	LwpMenuMtShowtablesizeinfoebox	&H2A3
LwpMenuMfUpdateobject	&H125	LwpMenuMtCreatetablewithgrid	&H2A4
LwpMenuMfEditorgreet	&H2DD	LwpMenuMtShowpcollineinfoebox	&H2A5
LwpMenuMfNewdivision	&H2DE	LwpMenuMtShowpcolsizeinfoebox	&H2A6
LwpMenuMfOpendivision	&H2DF	LwpMenuMtSplitpcol	&H2A7
LwpMenuMfSaContinueRoute	&H2E2	LwpMenuMtPcolselectentiretable	&H2A8
LwpMenuMfSaRoute	&H2E3	LwpMenuMtSelectentiretable	&H2A9
LwpMenuMfSeldatafile	&H30C	LwpMenuMtSelectentirecellrange	&H2AA
LwpMenuMfMrgdelimit	&H30C	LwpMenuMtSelectentirepcolcellrange	&H2AB
LwpMenuMfMrgletter	&H30D	LwpMenuMtTabletopalign	&H2AC
LwpMenuMfMrgenvelope	&H312	LwpMenuMtTablecenteralign	&H2AD
LwpMenuMfMrggotorecord	&H30E	LwpMenuMtTablebottomalign	&H2AE
LwpMenuMfCreatedesc	&H30F	LwpMenuMtConnectrow	&H2AF
LwpMenuMfMrgsort	&H310	LwpMenuMtPcolconnectrow	&H2B0
LwpMenuMfCreatemerge	&H311	LwpMenuMtGotonextblock	&H2B1
LwpMenuMfEditdatafile	&H313	LwpMenuMtCellinfoeboxmouse	&H2B2
LwpMenuMfUsecurasmrgdoc	&H314	LwpMenuMtTableinfoeboxmouse	&H2B3
LwpMenuMfOpenandeditdatafile	&H315	LwpMenuMxRunrexx	&H2BB
LwpMenuMfOpenpreview	&H316	LwpMenuMxRunontime	&H2BC
LwpMenuMfNewpreview	&H317	LwpMenuMxHourglass	&H2D0
LwpMenuMxMergemacro	&H318	LwpMenuMxMacpause	&H2D1
LwpMenuMfMergeprintall	&H319	LwpMenuMxMacresume	&H2D2
LwpMenuMfMergeopen	&H31A	LwpMenuMxTimedelay	&H2D3
LwpMenuMIsRunscript	&H31B	LwpMenuMxTestforcancel	&H2D4
LwpMenuMIsCreatescript	&H35F	LwpMenuMxMacmessages	&H2D5
LwpMenuMIsCreatedialog	&H360	LwpMenuMxPreanswermmsgbox	&H2D6

LwpMenuMIsCompile	&H365	LwpMenuMxMaccancel	&H2D8
LwpMenuMIsAutoapp	&H369	LwpMenuMxMaccontinue	&H2D9
LwpMenuMIsIdeStepIn	&H3C0	LwpMenuMxMacrovariables	&H2DA
LwpMenuMIsIdeStepOver	&H3C1	LwpMenuMxRunautonew	&H31E
LwpMenuMIsIdeStepExit	&H3C2	LwpMenuMxRunautoopen	&H31F
LwpMenuMIsIdeContinue	&H3C3	LwpMenuOutlinemenu	&H2EE
LwpMenuMIsIdeStop	&H3C4	LwpMenuMolPromote	&H2EF
LwpMenuMIsIdeSetBp	&H3C5	LwpMenuMolDemote	&H2F0
LwpMenuMIsIdeDisBp	&H3C6	LwpMenuMolMoveup	&H2F1
LwpMenuMIsIdeClearAll	&H3C7	LwpMenuMolMovedown	&H2F2
LwpMenuMIsIdeDisAll	&H3C8	LwpMenuMolExpand	&H2F3
LwpMenuMIsIdeBreakpoints	&H3C9	LwpMenuMolContract	&H2F4
LwpMenuMIsIdeBrowser	&H3CA	LwpMenuMolShowlevels	&H2F5
LwpMenuMIsIdeOutput	&H3CB	LwpMenuMolExpandsingle	&H2F6
LwpMenuMIsIdeVariables	&H3CC	LwpMenuMolContractsingle	&H2F7
LwpMenuMIsIdeCheckScript	&H3CD	LwpMenuMolUseoutlinestyle	&H2F8
LwpMenuMIsIdeNewSub	&H3CE	LwpMenuMolSetshowlevel1	&H2F9
LwpMenuMIsIdeNewFun	&H3CF	LwpMenuMolSetshowlevel2	&H2FA
LwpMenuMIsIdeDebug	&H3D0	LwpMenuMolSetshowlevel3	&H2FB
LwpMenuMIsIdeRedo	&H3D1	LwpMenuMolSetshowlevel4	&H2FC
LwpMenuMIsShowide	&H3D2	LwpMenuMolSetshowlevel5	&H2FD
LwpMenuMIsIdeImport	&H3D3	LwpMenuMolSetshowlevel6	&H2FE
LwpMenuMIsIdeExport	&H3D4	LwpMenuMolSetshowlevel7	&H2FF
LwpMenuMIsIdeScriptPref	&H3D5	LwpMenuMolSetshowlevel8	&H300
LwpMenuMIsIdeCloseWin	&H3D6	LwpMenuMolSetshowlevel9	&H301
LwpMenuMIsIdeToggle	&H3D7	LwpMenuMolSetshowlevel10	&H302
LwpMenuMIsIdePrevSub	&H3D8	LwpMenuMolOutstylesequence	&H2ED
LwpMenuMIsIdeNextSub	&H3D9	LwpMenuMtTabkey	&H320
LwpMenuMIsIdeTogglebp	&H3DA	LwpMenuMeDefpathsnw	&H324
LwpMenuMIsIdeCodeWindow	&H3DB	LwpMenuMvHidepicts	&H325
LwpMenuMIsIdeCheckAll	&H3DC	LwpMenuMvHidemarks	&H326
LwpMenuMIsIdeSaveLso	&H3DD	LwpMenuMvHidetabrack	&H327
LwpMenuMIsIdeShowicon	&H3E0	LwpMenuMvColguide	&H328
LwpMenuMIsIdeRuncursub	&H3E1	LwpMenuMvShowcleanscreen	&H348
LwpMenuMIsIdeHelpLs	&H3E2	LwpMenuMvFixediconbar	&H34A
LwpMenuMIsIdeHelpWp	&H3E3	LwpMenuMvViewlevel	&H34B
LwpMenuMIsIdeHelpSe	&H3E4	LwpMenuMtbChange	&H32A
LwpMenuMIsIdeHelpAbout	&H3E5	LwpMenuMtbTabbox	&H32B
LwpMenuMIsIdeInsertLso	&H3E6	LwpMenuMvVertruler	&H32E
LwpMenuMIRenumnotes	&H95	LwpMenuMvShownotes	&H330
LwpMenuEditmenu	&H96	LwpMenuMvShowtabsandreturns	&H332
LwpMenuMeUndo	&H97	LwpMenuMvOut2lay	&H334
LwpMenuMeCut	&H99	LwpMenuMfImportdraw	&H335
LwpMenuMeCopy	&H9A	LwpMenuMIMacroendrecord	&H336
LwpMenuMePaste	&H9B	LwpMenuMvLay2out	&H337

LwpMenuMeUndoredo	&H9C	LwpMenuMvSetpreferences	&H338
LwpMenuMePastelinks	&H9D	LwpMenuMrCreatelastframe	&H339
LwpMenuMeInsvariable	&H9E	LwpMenuMIQuickendrecord	&H33A
LwpMenuMeFootnotes	&HA0	LwpMenuMrUngroupframes	&H33B
LwpMenuMeNotes	&HA1	LwpMenuMIRevmarkingoff	&H33C
LwpMenuMeNotes20	&H11A	LwpMenuMIFieldhide	&H33D
LwpMenuMeDefaults	&HA2	LwpMenuMcExtracton	&H346
LwpMenuMeTypeover	&HA3	LwpMenuMdivDivisionproperties	&H34C
LwpMenuMeNbDefaults	&HA4	LwpMenuMdivNewdivision	&H34D
LwpMenuMeInsertdate	&HA5	LwpMenuMdivCombinedivisions	&H34E
LwpMenuMeAddmergevar	&HA6	LwpMenuMdivDeletedivision	&H34F
LwpMenuMeAdddocdescvar	&HA7	LwpMenuMdivExpanddivision	&H350
LwpMenuMxMdiswitch	&HA8	LwpMenuMdivGroupdivision	&H356
LwpMenuMeSeclevaction	&HA9	LwpMenuMsecSectionproperties	&H351
LwpMenuMeDefpaths	&HAA	LwpMenuMsecNewsection	&H352
LwpMenuMeFootopts	&HAB	LwpMenuMsecCombinesections	&H353
LwpMenuMeRedo	&HAC	LwpMenuMsecSelectsection	&H354
LwpMenuMeDocoptions	&HAD	LwpMenuMsecExpandsection	&H355
LwpMenuMeDefpath	&HAE	LwpMenuMtabSettabs	&H357
LwpMenuMeDefsty	&HAF	LwpMenuMtabCleartabs	&H358
LwpMenuMeDefback	&HB0	LwpMenuMtabLefttab	&H359
LwpMenuMeFootnoteinsert	&HB1	LwpMenuMtabRighttab	&H35A
LwpMenuMeInsertnote	&HB2	LwpMenuMtabCentertab	&H35B
LwpMenuMeDelete	&HB3	LwpMenuMtabNumerictab	&H35E
LwpMenuMeIndexentry	&HB4	LwpMenuMtabDefaulttabs	&H361
LwpMenuMeAddnote	&HB5	LwpMenuMtabRemoveruler	&H362
LwpMenuMeMergevar	&HB6	LwpMenuMtabHideruler	&H363
LwpMenuMeDescvar	&HB7	LwpMenuMtabQuickright	&H3DE
LwpMenuMeFormatcheckboxbar	&HB8	LwpMenuMtabQuickcenter	&H3DF
LwpMenuMePageup	&HB9	LwpMenuMrSetmarginsinfolbox	&H366
LwpMenuMePagedown	&HBA	LwpMenuMfToggleinfolbox	&H367
LwpMenuMeRevinsertion	&HBB	LwpMenuMfCreateversion	&H368
LwpMenuMeProtected	&HBC	LwpMenuMfCreateversionDivision	&H389
LwpMenuMeNohyphen	&HBD	LwpMenuMvViewversion	&H370
LwpMenuMeGetglosname	&HC6	LwpMenuMafidHelp	&H384
LwpMenuMeMacpath	&H11E	LwpMenuMafidNotfound	&H385
LwpMenuMeTocentry	&H343	LwpMenuMfMailNextstop	&H386
LwpMenuMeBullet	&H344	LwpMenuMfMailEditroute	&H387
LwpMenuMeMergedatafile	&H232	LwpMenuMergeScriptNotify	&H38E
LwpMenuMeMergeextdesc	&H234	LwpMenuFirstpopupid	&H39D
LwpMenuMeMergevarsdatafile	&H236	LwpMenuRenderformat	&H35C
LwpMenuMePastespecial	&H24C	LwpMenuRenderallformats	&H35D
LwpMenuMeInsertobject	&H24D	LwpMenuMcCommenttoolsOn	&H379
LwpMenuMeChangelink	&H24E	LwpMenuMnDeletenote	&H37A
LwpMenuMeUseworkpath	&H24F	LwpMenuMnClosenote	&H37B

LwpMenuMeSelectword	&H253	LwpMenuMnNextnote	&H37C
LwpMenuMeSelectsentence	&H254	LwpMenuMnPrevnote	&H37D
LwpMenuMeSelectparagraph	&H255	LwpMenuMcCommenttools	&H37E
LwpMenuMeSelectfile	&H256	LwpMenuMcHighnote	&H37F
LwpMenuMeRevisionmode	&H257	LwpMenuMnCloseall	&H380
LwpMenuMIDoccontrol	&H258	LwpMenuMnOpenall	&H381
LwpMenuMfSharedoccontrol	&H25A	LwpMenuMnDeleteall	&H382
LwpMenuMIDoccontrolEditorpanel	&H25B	LwpMenuMnShowinitials	&H383
LwpMenuMIDoccontrolExternal	&H25C	LwpMenuMnNumbernotes	&H384
LwpMenuMvViewversionExternal	&H25D	LwpMenuMfQuickprint	&H3B2
LwpMenuMfDocdescExternal	&H25E	LwpMenuMdDebugger	&H3E8
LwpMenuMfMrglabel	&H25F	LwpMenuMdDebugDoc	&H3E9
LwpMenuMfMrgletterssetup	&H260	LwpMenuMdDebugPres	&H3EA
LwpMenuMfMrgenvelopesetup	&H261	LwpMenuMdDebugDivinfo	&H3EB
LwpMenuMfMrgviewprint	&H264	LwpMenuMdDebugCont	&H3EC
LwpMenuNwobjects	&HBE	LwpMenuMdDebugCol	&H3ED
LwpMenuMoOpennw	&HBF	LwpMenuMdDebugLine	&H3EE
LwpMenuMoCreatenw	&HC0	LwpMenuMdDebugVobj	&H3EF
LwpMenuMoListnw	&HC1	LwpMenuMdDebugHeap	&H3F0
LwpMenuMoLinksnw	&HC2	LwpMenuMdDebugMain	&H3F1
LwpMenuMoAttribnw	&HC3	LwpMenuMdDebugProf	&H3F2
LwpMenuViewmenu	&HC8	LwpMenuMdDebugVcnt	&H3F3
LwpMenuMvFit	&HC9	LwpMenuMdDebugDivision	&H3F4
LwpMenuMvWorking	&HCA	LwpMenuMdDebugLay	&H3F5
LwpMenuMv75	&H31C	LwpMenuMdDebugVermgr	&H3F6
LwpMenuMv150	&H31D	LwpMenuMdDebugVerctrl	&H3F7
LwpMenuMv100	&HCB	LwpMenuMdDebugDoclay	&H3F8
LwpMenuMv200	&HCC	LwpMenuMdDebugPara	&H3FA
LwpMenuMvSwitchiconstate	&HCD	LwpMenuMdDebugInfo	&H3FB
LwpMenuMvPagesort	&H303	LwpMenuMdDebugAtom	&H3FC
LwpMenuMvZoomoptions	&H304	LwpMenuMdDebugWrdarg	&H3FD
LwpMenuMvComment	&H306	LwpMenuMdDebugDwrdarg	&H3FE
LwpMenuMvShowheadfoot	&H307	LwpMenuMdDebugStrarg	&H3FF
LwpMenuMvCustom	&H305	LwpMenuMdDebugSline	&H400
LwpMenuMvFacingpages	&H308	LwpMenuMdDebugPhint	&H401
LwpMenuMvMultiplepages	&H309	LwpMenuMdDebugChint	&H402
LwpMenuMvPagemargin	&H30A	LwpMenuMdDebugMsgtrace	&H403
LwpMenuMvPagewidth	&H30B	LwpMenuMdDebugFoundry	&H404
LwpMenuMvSpecialview	&H323	LwpMenuMdDebugCounts	&H405
LwpMenuMvClearspecialview	&H371	LwpMenuMdDebugPresinfo	&H406
LwpMenuMvSynchronizeviews	&H364	LwpMenuMdDebugMeter	&H407
LwpMenuMvClearhighlight	&H36A	LwpMenuMdDebugStack	&H408
LwpMenuMvPreferences	&HCE	LwpMenuMdDebugPtrholder	&H409
LwpMenuMvTabrack	&HCF	LwpMenuMdDebugMemory	&H40A
LwpMenuMvFunctionbar	&HD0	LwpMenuMdDebugMemorydump	&H40B

LwpMenuMvChgsidebar	&HD1	LwpMenuMdDebugContent	&H40C
LwpMenuMvLayout	&HD2	LwpMenuMdDebugSilver	&H40D
LwpMenuMvDraft	&HD3	LwpMenuMdDebugVernode	&H40E
LwpMenuMvFacing	&HD4	LwpMenuMdDebugCheckpoint	&H40F
LwpMenuMvFacecancel	&HD6	LwpMenuMdDebugSocket	&H410
LwpMenuMvShowpicts	&HD7	LwpMenuMdDebugSfb	&H411
LwpMenuMvShowmarks	&HD8	LwpMenuMdDebugDemandload	&H412
LwpMenuMvPreferdefaults	&HD9	LwpMenuMdDebugNumberhint	&H413
LwpMenuMvOutline	&HDA	LwpMenuMdDebugDialog	&H414
LwpMenuMvFloatingicon	&HDB	LwpMenuMdDebugBack	&H415
LwpMenuMvLefticon	&HDC	LwpMenuMdDebugHeadtailholder	&H416
LwpMenuMvRighticon	&HDD	LwpMenuMdDebugNotifypersistent	&H417
LwpMenuMvTopicon	&HDE	LwpMenuMdDebugObjectholder	&H418
LwpMenuMvBottomicon	&HDF	LwpMenuMdDebugEnumobjhead	&H419
LwpMenuMvSmarticons	&HE8	LwpMenuMdDebugNotify	&H41A
LwpMenuMvFullscreen	&HE9	LwpMenuMdDebugNamedobjectholde r	&H41B
LwpMenuMvFullscreenoptions	&HEA	LwpMenuMdDebugSection	&H41C
LwpMenuMvEditicon	&HEC	LwpMenuMdDebugFontmgr	&H41D
LwpMenuMfPrintenvelope	&HED	LwpMenuMdDebugSanity	&H41E
LwpMenuMvNexticonset	&HEF	LwpMenuMdDebugAtomWatch	&H41F
LwpMenuMvSaveicon	&HF0	LwpMenuMdDebugAtomCount	&H420
LwpMenuMvCreateicon	&HF1	LwpMenuMdDebugPieceCount	&H421
LwpMenuMvFullscreenstartup	&HF2	LwpMenuMdDebugObjectSizes	&H422
LwpMenuMvSectiontabs	&HF4	LwpMenuMdDebugTextstyle	&H423
LwpMenuCharmenu	&HFA	LwpMenuMdDebugEnd	&H44B
LwpMenuMcTextinfobox	&H11B	LwpMenuMkPageup	&H44C
LwpMenuMcTextinfoboxmouse	&H262	LwpMenuMkPagedown	&H44D
LwpMenuMcDrawinfoboxmouse	&H263	LwpMenuMcDooledragdrop	&H44E
LwpMenuMcFont	&HFB	LwpMenuMsMplMargins	&H514
LwpMenuMcNormal	&HFD	LwpMenuMsMplPage	&H515
LwpMenuMcBold	&HFE	LwpMenuMsMplFooter	&H516
LwpMenuMcItalic	&HFF	LwpMenuMsMplHeader	&H517
LwpMenuMcUnderline	&H100	LwpMenuMsMplLines	&H518
LwpMenuMcWordunder	&H101	LwpMenuMrMflType	&H578
LwpMenuMcLeft	&H103	LwpMenuMrMflSizepos	&H579
LwpMenuMcCenter	&H104	LwpMenuMrMflLines	&H57A
LwpMenuMcRight	&H105	LwpMenuMrMflColumnstabs	&H57B
LwpMenuMcJustify	&H106	LwpMenuMISetmacrofile	&H5DC
LwpMenuMcAlignment	&H107	LwpMenuMICompilemacro	&H5DD
LwpMenuMcCaps	&H108	LwpMenuMIMacrorun	&H5DE
LwpMenuMcIndentall	&HF7	LwpMenuMIMacroappend	&H5DF
LwpMenuMcIndentfirst	&HF8	LwpMenuMIPrototype	&H5E0
LwpMenuMcIndentrest	&HF9	LwpMenuMeSmartcorrect	&H5E1
LwpMenuMcExtract	&HF6	LwpMenuMrChart	&H5E2
LwpMenuMcMacrofastformat	&HEE	LwpMenuMrCharttodraw	&H5E3

LwpMenuMcFontvert	&HF3	LwpMenuMmCascades	&H640
LwpMenuMcEnlargetext	&HFC	LwpMenuSubmenuInsert	&H640
LwpMenuMcReducetext	&HF5	LwpMenuMmFields	&H641
LwpMenuMcIndent	&H102	LwpMenuSubmenuMarktextas	&H642
LwpMenuMcOutdent	&H11C	LwpMenuSubmenuAlignment	&H643
LwpMenuMcBulletpanel	&H11D	LwpMenuMmCaps	&H644
LwpMenuMcNextfacename	&H3B6	LwpMenuMmMacros	&H647
LwpMenuMcNextalignment	&H3B7	LwpMenuSubmenuInsertsmartsum	&H649
LwpMenuMcNextattribute	&H3B8	LwpMenuMMailmemo	&H64A
LwpMenuMcNextbullet	&H3B9	LwpMenuMPrinttofile	&H64B
LwpMenuMcNextnumbering	&H3BA	LwpMenuSubmenuUsersetup	&H64C
LwpMenuMcNextstyle	&H3BB	LwpMenuSubmenuScripts	&H64D
LwpMenuMcDoubleunder	&H10E	LwpMenuMmShowoptions	&H64E
LwpMenuMcSuperscript	&H10F	LwpMenuSubmenuQuickdemo	&H64F
LwpMenuMcSubscript	&H110	LwpMenuSubmenuOtherhelpstuff	&H650
LwpMenuMcAllcaps	&H111	LwpMenuMmOtherframe	&H651
LwpMenuMfEnvelopeprint	&H112	LwpMenuMmOthertable	&H652
LwpMenuMcAllower	&H11F	LwpMenuSubmenuOthertextstuff	&H653
LwpMenuMcInitscaps	&H120	LwpMenuSubmenuPriority	&H654
LwpMenuMcSmallcaps	&H121	LwpMenuMmOtherstuff	&H655
LwpMenuMcJustfont	&H116	LwpMenuSubmenuSelect	&H656
LwpMenuMcJustpointsize	&H117	LwpMenuSubmenuOtherdocpart	&H657
LwpMenuMcStrikethru	&H118	LwpMenuSubmenuAttributes	&H658
LwpMenuMcFastformat	&H123	LwpMenuSubmenuOutline	&H659
LwpMenuMcHighlight	&H126	LwpMenuSubmenuCollapse	&H65A
LwpMenuStylemenu	&H12C	LwpMenuSubmenuEdit	&H65B
LwpMenuMsCreate	&H12D	LwpMenuSubmenuDebug	&H65C
LwpMenuMsStylemanage	&H12F	LwpMenuSubmenuDocprops	&H65D
LwpMenuMsChgstyle	&H130	LwpMenuSubmenuZoomto	&H65E
LwpMenuMsSavestyle	&H131	LwpMenuSubmenuShowhide	&H65F
LwpMenuMsSelectstyle	&H132	LwpMenuSubmenuOtherviewstuff	&H660
LwpMenuMsOutlinestyle	&H140	LwpMenuSubmenuNamedstyles	&H661
LwpMenuMsStylemanageinit	&H141	LwpMenuSubmenuDelete	&H662
LwpMenuMsStylemanageaction	&H142	LwpMenuSubmenuFile	&H663
LwpMenuMsChangestylename	&H143	LwpMenuSubmenuView	&H664
LwpMenuMsDefinestyle	&H144	LwpMenuSubmenuTeammail	&H665
LwpMenuMsModcreatestyle	&H145	LwpMenuSubmenuEditmodemenu	&H666
LwpMenuMsStylemanagefinish	&H146	LwpMenuSubmenuDebugmodemenu	&H667
LwpMenuMsRedefinestyle	&H150	LwpMenuSubmenuHelp	&H668
LwpMenuMsMargins	&H148	LwpMenuCreatemenu	&H672
LwpMenuMsPagesettings	&H149	LwpMenuParallelstylemenu	&H67A
LwpMenuMsApply	&H14A	LwpMenuEnvAddress	&H67D
LwpMenuMsStylemru1	&H14B	LwpMenuEnvSave	&H681
LwpMenuMsStylemru2	&H14C	LwpMenuEnvPrint	&H682
LwpMenuMsStylemru3	&H14D	LwpMenuMfCreatetabdivision	&H686

LwpMenuMsStylemru4	&H14E	LwpMenuMfCreatedivision	&H683
LwpMenuMrManuatable	&H14F	LwpMenuMeTocassist	&H684
LwpMenuMrFrameanchorinfo	&H187	LwpMenuMpNewsection	&H685
LwpMenuMrFramelineinfo	&H188	LwpMenuMtInsertclickhere	&H687
LwpMenuMrFramebackinfo	&H189	LwpMenuMcTextlinepanel	&H688
LwpMenuMrFramegfxinfo	&H18A	LwpMenuMvOtherviewstuff	&H689
LwpMenuMrFrametopalign	&H18B	LwpMenuMoApplystyle	&H68A
LwpMenuMrFramecenteralign	&H18C	LwpMenuMsFramecreate	&H68B
LwpMenuMrFramebottomalign	&H18D	LwpMenuMsFramerefinestyle	&H68C
LwpMenuMrDeleteframe	&H18F	LwpMenuMsTablecreate	&H68D
LwpMenuPagemenu	&H15E	LwpMenuMsTablerefinestyle	&H68E
LwpMenuMpCreatelayout	&H15F	LwpMenuMsTablecellcreate	&H68F
LwpMenuMpModifylayout	&H160	LwpMenuMsTablecellrefinestyle	&H690
LwpMenuMpTabs	&H163	LwpMenuMvShowmisspelled	&H691
LwpMenuMpPagenum	&H164	LwpMenuMtDelrowacc	&H692
LwpMenuMpPagebreak	&H166	LwpMenuMtCtrlretacc	&H693
LwpMenuMpLinenum	&H167	LwpMenuMsQuickrefinestyle	&H694
LwpMenuMpHeadfoot	&H168	LwpMenuMtInsertrowacc	&H695
LwpMenuMpModlayoutinit	&H169	LwpMenuMrOleconvert	&H696
LwpMenuMpModlaypagesettings	&H16A	LwpMenuMrOleconvertandactivate	&H697
LwpMenuMpModlayrightfacepage	&H16B	LwpMenuMrFrameleft	&H698
LwpMenuMpModlayrightheadfacepage	&H16C	LwpMenuMrFramecenter	&H699
LwpMenuMpModlayrightfooterfacepage	&H16D	LwpMenuMrFrameright	&H69A
LwpMenuMpModlayrightlines	&H16E	LwpMenuMrFramehorzspan	&H69B
LwpMenuMpModlayleftfacepage	&H16F	LwpMenuMrFramevertspan	&H69C
LwpMenuMpModlayleftheaderfacepage	&H170	LwpMenuMfNewanytab	&H69D
LwpMenuMpModlayleftfooterfacepage	&H171	LwpMenuMfOdmaopen	&H69E
LwpMenuMpModlayleftlines	&H172	LwpMenuMfOdmaininsert	&H69F
LwpMenuMpDellayout	&H175	LwpMenuMfOdmaconvert	&H6A0
LwpMenuMpRevertlayout	&H176	LwpMenuMfOdmasaveas	&H6A1
LwpMenuMpInslayout	&H177	LwpMenuMfOdmaexternalize	&H6A2
LwpMenuMpModlayoutfinish	&H178	LwpMenuMfOdmaexport	&H6A3
LwpMenuMpHeaderfootergoto	&H179	LwpMenuMrOleverb	&H6A4
LwpMenuMpMacinslayout	&H17A	LwpMenuMrOleverbmax	&H707
LwpMenuMpParallel	&H17B	LwpMenuMrOledivtabverb	&H708
LwpMenuMpPcolinfo	&H17C	LwpMenuMrOledivtabverbmax	&H76B
LwpMenuMpPageproperties	&H17D	LwpMenuMtSelectcell	&H76C
LwpMenuMpCreate	&H17E	LwpMenuMtPcolselectcell	&H76D
LwpMenuMpNewpagestyle	&H17F	LwpMenuMpColumnbreak	&H76E
LwpMenuMpHeaderproperties	&H180	LwpMenuMeViewtablegrid	&H76F
LwpMenuMpFooterproperties	&H181	LwpMenuMeViewtableheading	&H770
LwpMenuMpPagepropertiesmouse	&H182	LwpMenuMeViewpcolguides	&H771
LwpMenuMpFooterpropertiesmouse	&H183	LwpMenuMeViewmarginguides	&H772
LwpMenuMpHeaderpropertiesmouse	&H184	LwpMenuMeViewpagegauge	&H773
LwpMenuMpPcolinfo	&H185	LwpMenuMeViewshowallmarks	&H774



LwpMenuFramemenu	&H190	LwpMenuMeViewhideallmarks	&H775
LwpMenuMrFrametext	&H192	LwpMenuMtTableleft	&H776
LwpMenuMrScale	&H193	LwpMenuMtTablecenter	&H777
LwpMenuMrGroupframes	&H194	LwpMenuMtTableright	&H778
LwpMenuMrFrameinfobox	&H195	LwpMenuMtTablehorzspan	&H779
LwpMenuMrBringtofront	&H197	LwpMenuMtGotoinfobox	&H77A
LwpMenuMrSendtoback	&H198	LwpMenuMhTour	&H77B
LwpMenuMrAdd	&H199	LwpMenuMfCreatedivision	&H77D
LwpMenuMrSendbackone	&H19A	LwpMenuMfToggleclickhereprompts	&H77E
LwpMenuMrDrawingicon	&H19B	LwpMenuMcToggleskipbullet	&H77F
LwpMenuMrChartingicon	&H19C	LwpMenuMlSkipmisspelledword	&H780
LwpMenuMrCreateframe	&H19D	LwpMenuMlSkipallmisspelledword	&H781
LwpMenuMrAddframewithdlg	&H19E	LwpMenuMlAddtouserdictionary	&H782
LwpMenuMrFramedefaults	&H19F	LwpMenuMlTogglebubblehelp	&H783
LwpMenuMrFramelines	&H1A0	LwpMenuMfPlainnew	&H784
LwpMenuMrFramestype	&H1A1	LwpMenuMvToggleshowanchors	&H785
LwpMenuMrFrameborders	&H1A2	LwpMenuMvNextnewestversion	&H786
LwpMenuMrFramecolumns	&H1A3	LwpMenuMvMostrecentversion	&H787
LwpMenuMrModframereflow	&H1A4	LwpMenuMvOldestversion	&H788
LwpMenuMrModframeinit	&H1A5	LwpMenuMcJustifyall	&H789
LwpMenuMrBringfrontone	&H186	LwpMenuMcClosegraphicole	&H78A
LwpMenuMrSelectframe	&H18E	LwpMenuMfFramerevert	&H78B
LwpMenuMrEquationsicon	&H1A7	LwpMenuMtCellrevert	&H78C
LwpMenuMrImageprocessing	&H1A8	LwpMenuMfOpentabdivision	&H78D
LwpMenuMrFrameinfoboxmouse	&H26C	LwpMenuMfNewtabdivision	&H78E
LwpMenuMrFramelineinfoboxmouse	&H26D	LwpMenuMtInsertclickherelink	&H78F
LwpMenuMrFramebackinfoboxmouse	&H26E	LwpMenuMtEditclickherelink	&H790
LwpMenuMrFramegfxinfoboxmouse	&H26F	LwpMenuMsecDeletesection	&H79D
LwpMenuMrFrameanchorinfoboxmouse	&H270	LwpMenuMfExternalfileexit	&H79E
LwpMenuMrExpert	&H38F	LwpMenuMfExternalfilesavedivision	&H79F
LwpMenuMrExpertmax	&H39C	LwpMenuMfFtpopen	&H7A0
LwpMenuMrWppack	&H15D	LwpMenuMfFtpinsert	&H7A1
LwpMenuMrAfid	&H1A9	LwpMenuMfFtpconvert	&H7A2
LwpMenuMrAfidmax	&H1C1	LwpMenuMfFtpsaveas	&H7A3
LwpMenuToolmenu	&H1C2	LwpMenuMfFtpexternalize	&H7A4
LwpMenuMlSearch	&H1C3	LwpMenuSubmenuInternet	&H7A5
LwpMenuMlGoto	&H1C4	LwpMenuMfFtpoptions	&H7A6
LwpMenuMlSpellopts	&H1C5	LwpMenuMfOdmashowattributes	&H7A7
LwpMenuMlUserdict	&H1C6	LwpMenuMrRuby	&H7A8
LwpMenuMlThesaurus	&H1C7	LwpMenuMfFtpopendivision	&H7A9
LwpMenuMlSort	&H1C8	LwpMenuMfFtpopenurl	&H7AA
LwpMenuMlTables	&H1C9	LwpMenuMlShowfrommenu	&H7AB
LwpMenuMlGotolast	&H1CA	LwpMenuMoHideinfobox	&H7AC
LwpMenuMlSpellopts2	&H1CB	LwpMenuMfWelcome	&H7AD
LwpMenuMlRevmarkoptions	&H1CD	LwpMenuWp5Expert	&H7AE

LwpMenuMIToc	&H1CE	LwpMenuWp6Expert	&H7AF
LwpMenuMIAssignmacro	&H1CF	LwpMenuWpWelcome	&H7B0
LwpMenuMIMacros	&H1D0	LwpMenuMsQuickredefineframestyle	&H7B1
LwpMenuMIMacroaccel	&H1A6	LwpMenuMsQuickredefinetablestyle	&H7B2
LwpMenuMIGeneratetoc	&H1D1	LwpMenuMsQuickredefinecellstyle	&H7B3
LwpMenuMIIndexnextfile	&H119	LwpMenuMsQuickredefinestyle	&H7B4
LwpMenuMIBookmark	&H1D2	LwpMenuMeInsertsymbol	&H7B5
LwpMenuMIGlossary	&H1D3	LwpMenuMeWordcount	&H7B6
LwpMenuMIGlosset	&H1D4	LwpMenuMeDropcaps	&H7B7
LwpMenuMIMacpause	&H1D5	LwpMenuMfOdmaintportpicture	&H7B8
LwpMenuMIGenerateindex	&H1D6	LwpMenuMrOleobjectinabox	&H7B9
LwpMenuMISpelladdict	&H1D7	LwpMenuMcInternettoolson	&H7BA
LwpMenuMISpellskipall	&H1D8	LwpMenuMcInternettools	&H7BB
LwpMenuMISpellreplace	&H1DA	LwpMenuMhLotushomepage	&H7BC
LwpMenuMISpell	&H1DB	LwpMenuMhLotussupport	&H7BD
LwpMenuMISpellinit	&H1DC	LwpMenuMhLotusftpsite	&H7BE
LwpMenuMISpellcheck	&H1DD	LwpMenuMrLinkframes	&H7BF
LwpMenuMISpellcancel	&H1DE	LwpMenuMrUnlinkframes	&H7C0
LwpMenuMISpellskip	&H1DF	LwpMenuMfMrgwelcome	&H7C1
LwpMenuMIReplace	&H1E0	LwpMenuMfHtmloptions	&H7C2
LwpMenuMIChglang	&H1E1	LwpMenuMcUpdateindex	&H7C3
LwpMenuMIRenumseq	&H1E2	LwpMenuMcUpdatetoc	&H7C4
LwpMenuMISrattr	&H1E5	LwpMenuMvDesignmode	&H7C5
LwpMenuMIRepopts	&H1E6	LwpMenuMfNosopen	&H7C6
LwpMenuMISropts	&H1E7	LwpMenuMfNossaveas	&H7C7
LwpMenuMISearchonly	&H1E8	LwpMenuMfMailSendmessage	&H7C8
LwpMenuMIMacplay	&H1E9	LwpMenuMcInsertdefnumber	&H7C9
LwpMenuMIAssign	&H1EA	LwpMenuMcInsertdefbullet	&H7CA
LwpMenuMIACdde	&H1EB	LwpMenuMeTogglesmartselect	&H7CB
LwpMenuMIACtype	&H1EC	LwpMenuMcCreateobjectviewer	&H7CC
LwpMenuMICall	&H1ED	LwpMenuMcCreateobjectcomment	&H7CD
LwpMenuMISammy2	&H1EE	LwpMenuMcCreateobjectdraw	&H7CE
LwpMenuMITocopts	&H1EF	LwpMenuMcCreateobjectschedule	&H7CF
LwpMenuMISrfindmatch	&H1F0	LwpMenuMcCreateobjectsheetspreadsheet	&H7D0
LwpMenuMISrreplacethenfind	&H1F1	LwpMenuMcCreateobjectchart	&H7D1
LwpMenuMISrcancel	&H1F2	LwpMenuMhInternetsearch	&H7D2
LwpMenuMISrinit	&H1F3	LwpMenuMhLotusSMARTsuite	&H7D3
LwpMenuMIDoccompare	&H1F4	LwpMenuMcClickherekeyword	&H7D4
LwpMenuMIACredit	&H1F5	LwpMenuMfPublishweb	&H7D5
LwpMenuMIACplay	&H1F6	LwpMenuMICrossreference	&H7D6
LwpMenuMIACrecord	&H1F7	LwpMenuMtabHidevertruler	&H7D7
LwpMenuMIACoptions	&H1F8	LwpMenuMpUpdateheaders	&H7D8
LwpMenuMIACquickrec	&H1F9	LwpMenuMpUpdatefooters	&H7D9
LwpMenuMIACquickplay	&H1FA	LwpMenuMrRubyAbove	&HBB8
LwpMenuMIACgrammar	&H1FB	LwpMenuMrRubyBelow	&HBB9

LwpMenuMIGrammaropts	&H1FC	LwpMenuMrRubyDelete	&HBBA
LwpMenuMIGrammarstart	&H1FD	LwpMenuMxMinmacromenu	&H2328
LwpMenuMIGrammarinit	&H1FE	LwpMenuMxMaxmacromenu	&H238B
LwpMenuMIModcreatorule	&H1FF	LwpMenuMrAfidmenu	&H2710
LwpMenuMIGrammarnext	&H200	LwpMenuMrAfidmenumax	&H3E80
LwpMenuMIGrammarcancel	&H201	LwpMenuMaNotesflow	&H3E81
LwpMenuMIGrammarsugtext	&H202	LwpMenuMaNotesflowmax	&H3F48
LwpMenuMIGrammaroptionstext	&H203	LwpMenuFcsTextmenu	&H0
LwpMenuMIGrammarchange	&H204	LwpMenuFcsFramemenu	&H1
LwpMenuMIGrammarstats	&H205	LwpMenuFcsFrametextmenu	&H2
LwpMenuMIIconpath	&H206	LwpMenuFcsTabletextmenu	&H3
LwpMenuMICyclekeysetup	&H207	LwpMenuFcsFramegraphicmenu	&H4
LwpMenuMISmartfill	&H208	LwpMenuFcsPowerfieldmenu	&H5
LwpMenuMINewspell	&H209	LwpMenuFcsParallelcoltextmenu	&H6
LwpMenuMIRevisionbar	&H20A	LwpMenuFcsDivisionmenu	&H7
LwpMenuMIToa	&H20B	LwpMenuFcsSectionmenu	&H8
LwpMenuWindowmenu	&H4	LwpMenuFcsTablemenu	&H9
LwpMenuMwNewwindow	&H20D	LwpMenuFcsParallelcolmenu	&HA
LwpMenuMwTilewindow	&H20E	LwpMenuFcsTablegraphicmenu	&HB
LwpMenuMwCascadewindow	&H20F	LwpMenuFcsParallelcolgraphicmenu	&HC
LwpMenuMwStartoffiles	&H210	LwpMenuFcsHeadermenu	&HD
LwpMenuMwTilewindowhorz	&H246	LwpMenuFcsFootermenu	&HE
LwpMenuMwSplitvertwindow	&H251	LwpMenuFcsRulermenu	&HF
LwpMenuMwSplithorzwindow	&H252	LwpMenuFcsNotemenu	&H10
LwpMenuMwEndoffiles	&H224	LwpMenuFcsOutlinemenu	&H11
LwpMenuMwFilesseparator	&H225	LwpMenuFcsOledivisionmenu	&H12
LwpMenuMIIMacresume	&H22A	LwpMenuFcsFramechartmenu	&H13
LwpMenuMIReviewrevs	&H22D	LwpMenuFcsTablechartmenu	&H14
LwpMenuMIRevacceptall	&H36B	LwpMenuFcsParallelcolchartmenu	&H15
LwpMenuMIRevcancelall	&H36C	LwpMenuFcsTablecornermenu	&H16
LwpMenuMIRevaccept	&H36D	LwpMenuFcsTablerowmenu	&H17
LwpMenuMIRevcancel	&H36E	LwpMenuFcsTablecolumnmenu	&H18
LwpMenuMIField	&H238	LwpMenuFcsVertrulermenu	&H19
LwpMenuMIFieldnext	&H239	LwpMenuFcsRubymenu	&H1A
LwpMenuMIFieldprev	&H23A	LwpMenuFcsFootnotetextmenu	&H28
LwpMenuMIFieldeval	&H23B	LwpMenuFcsIndextextmenu	&H29
LwpMenuMIFieldadd	&H23C	LwpMenuFcsToctextmenu	&H2A
LwpMenuMIFieldupdate	&H23D	LwpMenuFcsTablecornermenu	&H2B
LwpMenuMIFieldtog	&H10A	LwpMenuFcsTabletextmenu	&H2C
LwpMenuMIFieldupall	&H10B	LwpMenuFcsFrametextmenudropcap	&H2D
LwpMenuMIFieldlock	&H10C	LwpMenuFcsFramemenudropcap	&H2E
LwpMenuMIFieldsave	&H10D		

## Overview: Word Pro Text Subobjects

In Word Pro, text can appear in many places and in many forms including the text which flows from page to page or the text within a table cell or a frame. Each of these text streams is seen as a separate Text object. However, a Text object can be comprised of much more than simple ASCII text characters. A Text object may contain smaller items known as text subobjects.

such as paragraph markers, tabs, frames, tables, Comment Notes and much more. In fact, Word Pro can discern up to 25 distinct types of Text subobjects which are not classified as ASCII text. The total number of Text subobjects including ASCII text is 26.

You can detect the presence of any of these text subobjects when you call one of the following methods from a Text object:

### *GetText*

Returns the Subobject Name for the subobject located at the cursor.

### *Next*

Moves the caret to the next instance of the specified type of subobject.

### *Previous*

Moves the caret to the previous instance of the specified type of subobject.

## The Different Types of ASCII Text

While most of the Text subobjects represent special characters or markers, there is one subobject which represents ASCII text characters. This subobject is called Text and Word Pro uses the name Text to refer to any group of adjacent ASCII text characters which share the same attributes.

For example, the following sentence is a single Text object.

My summer vacation:[TAB]What a trip![EOP]

However, it is comprised of four distinct Text subobjects as seen in this table:

<u>Subobject</u>	<u>Subobject Type</u>
"My summer vacation:"	Text (plain)
[TAB]	Tab
"What a trip!"	Text (plain)
[EOP]	EOP (or End Of Paragraph)

Note the ASCII characters in the phrase "My summer vacation:". All the characters share the same attributes so Word Pro sees them as a single Text subobject of type Text. Word Pro is capable of detecting even the most subtle differences in ASCII text attributes. For example, if you apply the Bold attribute to a word, Word Pro sees that word as a distinct Text subobject which is separate from any adjacent subobjects. Therefore, the same sentence with different attributes would yield an entirely different result. For example:

**My** summer vacation:[TAB]*What* a trip![EOP]

This sentence would yield two more subobjects of type Text as seen in this table:

<u>Subobject</u>	<u>Subobject Type</u>
<b>"My"</b>	Text (bold)
" summer vacation:"	Text (plain)
[TAB]	Tab
<i>"What"</i>	Text (italic)
" a trip!"	Text (plain)
[EOP]	EOP (or End Of Paragraph)

In this example, Word Pro sees the words "My" and "What" as separate Text subobjects because one is Bold and the other is Italic. But that is only the beginning. There are many attributes which cannot be seen on the page. For example, if Word Pro does not recognize a word, it will mark that word as misspelled. That mark would cause Word Pro to see the word as a separate subobject of type Text.

The table below lists the many attributes which can be applied to ASCII text characters. When any combination of these attributes is applied to a group of adjacent ASCII text characters, Word Pro sees those characters as a separate Text subobject of type Text.

<u>Attribute</u>	<u>Description</u>
------------------	--------------------

skipped  
 misspelled  
 hidden  
 grammar error  
 protected  
 no hyphen  
 double-word  
 highlighted  
 revision  
 language  
 font size  
 font face  
 color  
 character style  
 attributes  
 code page

The table below provides descriptions of each of these Text subobjects.

<b>Subobject Name</b>	<b>Description</b>
AnchoredFrame	A Frame which is placed "In Text", "In Text - Vertical", or "With Paragraph Above".
Bookmark	The Marker for either the Start or End of a Bookmark range.
ClickHereBlock	The Marker for either the Start or End of a ClickHereBlock.
ColumnBreak	A hard column break within a multi-column page layout. This is created by choosing Text - Insert Other and then Column Break.
DDE	The Marker for either the Start or End of a DDE link.
DocVariable	A DocInfo field
EOP	The marker at the <b>End Of a Paragraph</b> created by pressing ENTER.
FootnoteMark	The Marker for either the Start or End of a Footnote.
HardSpace	A non-breaking space created by pressing CTRL+Spacebar.
HKatakana	A special Asian language character. Seen in Asian language versions only.
Kanji	A special Asian language character. Seen in Asian language versions only.
LineBreak	A soft line break created by pressing SHIFT+ENTER.

Note	A Comment Note created by choosing Create - Comment Note.
PageBreak	A hard page break created by pressing CTRL+ENTER.
PageNumber	A page number created by choosing Text - Insert Page Number.
ParaNumber	The outline style sequence position number for a paragraph.
PowerField	The Marker for either the Start or End of a PowerField.
RubyFrame*	Available only in the Asian language version of Word Pro.
RubyMarker*	Available only in the Asian language version of Word Pro.
Section	A Section marker which defines the separation of two sections within a Word Pro document.
SoftHyphen	A hyphen which does not appear unless the hyphenated word flows to the end of a line where it would break naturally.
SpecialTab	SpecialTabs contain their own tab stop and alignment information. Normal Tab markers derive their tab stop and alignment information from the Ruler.  SpecialTab markers are created in two ways: the InsertSpecialTab method or by importing a document from a file format which uses SpecialTabs (such as WordPerfect).
Tab	A normal Tab marker created by pressing TAB.
Table	A Table which is placed "In Text", "In Text - Vertical", or "With Paragraph Above".
Text	ASCII text characters. For details, see The Different Types of ASCII Text below.
Tombstone	A revision marker which displays the initials of the author of a particular revision.
UniCode	A special foreign-language type of character used in multi-language documents.

While most text subobjects are not represented in the Word Pro object model, there are eleven subobjects which have their own corresponding LotusScript classes. The table below lists these eleven subobjects and names the LotusScript classes which represent these subobjects:

<u>Subobject Name</u>	<u>Class Name of Corresponding LotusScript Object</u>
Table	TableLayout, SuperTableLayout
AnchoredFrame	FrameLayout
FootnoteMark	FootnoteLayout
Section	Section
Note	NoteLayout
Bookmark	BookMark
DDE	DdeLink
PowerField	PowerField
ClickHereBlock	ClickHere
RubyMarker	RubyMarker (Asian language versions only)
RubyFrame	RubyLayout (Asian language versions only)

only)

If a subobject is represented by its own LotusScript class, you can control that subobject directly without going through the Text object. However, if the Text subobject does not have its own LotusScript class, the only way to manipulate the subobject is to go through the Text object which contains that subobject.

For example, In a sentence which contains ASCII text, a soft hyphen, and a Bookmark, you would use the Bookmark class to manipulate the Bookmark. However, there is no class for the soft hyphen so you would manipulate the soft hyphen by going through the Text object which contains that soft hyphen.

## Overview: Word Pro Scalar Data Types

LotusScript recognizes the following scalar (numeric and string) data types:

<b>Data type</b>	<b>Suffi x</b>	<b>Value range</b>	<b>Size</b>
Integer	%	-32,768 to 32,767Initial value: 0	2 bytes
Long	&	-2,147,483,648 to 2,147,483,647Initial value: 0	4 bytes
Single	!	-3.402823E+38 to 3.402823E+38Initial value: 0	4 bytes
Double	#	-1.7976931348623158+308 to 1.7976931348623158+308Initial value: 0	8 bytes
Currency	@	-922,337,203,685,477.5807 to 922,337,203,685,477.5807Initial value: 0	8 bytes
String	\$	(String length ranges from 0 to 32K characters)Initial value: "" (empty string)	(2 bytes/character)

Besides these scalar data types, LotusScript supports the following additional data types and data structures:

<b>Data type or structure</b>	<b>Description</b>	<b>Size</b>
Array	An aggregate set of elements having the same data type. An array can comprise up to 8 dimensions whose subscript bounds can range from -32768 to 32767. Initial value: Each element in a fixed array has an initial value appropriate to its data type.	Up to 64K bytes
List	A one-dimensional aggregate set whose elements have the same data type and are referred to by name rather than by subscript.	Up to 64K bytes
Variant	A special data type that can contain any scalar value, array, list, or object reference. Initial value: EMPTY	16 bytes
User-defined data type	An aggregate set of elements of possibly disparate data types. Comparable to a record in Pascal or a structure in C. Initial value: Member variables have initial values appropriate to their data type.	Up to 64K bytes
Class	An aggregate set of elements of possibly disparate data types together with procedures that operate on them. Initial value: When you create an instance of a class, LotusScript initializes its member variables to values appropriate to their data types, and generates an	



Object reference	object reference to it. A pointer to an OLE Automation object or an instance of a product class or user-defined class. Initial value: NOTHING.	4 bytes
------------------	---	---------

In each of the preceding tables, the specified storage size is platform-independent.

## StrField Function

{button ,AL(' ',1)} [See example](#)

Extracts a string from within another string. Use this function to extract a string from a source string that contains two or more strings separated by a common character, such as a comma or a space.

### Syntax

StrField(SourceString, FieldNum, Separator)

### Elements

#### *SourceString*

A String containing two or more strings separated by the same character. The string "Seattle, WA;Atlanta, GA;Cambridge, MA;USA, All cities" contains four or more strings, depending on which separator character you use.

#### *FieldNum*

An Integer indicating the position of the string to be extracted. String positions are determined by the separator character specified in the Separator element. In the string "Seattle, WA;Atlanta, GA;Cambridge, MA;USA, All cities," a semicolon separator puts "Atlanta, GA" at position 2. Alternatively, a comma separator puts " WA;Atlanta" at position 2.

#### *Separator*

The character which separates the strings in the SourceString. In the string "Seattle, WA;Atlanta, GA;Cambridge, MA;USA, All cities," a semi-colon separator creates four strings and a comma separator creates five strings.

### Return value

StrField returns a string.

## Overview: The LotusScript Language

LotusScript is an object-oriented programming language which is shared by most Lotus applications, including Lotus Word Pro. You can use LotusScript as you would a macro language, to automate tasks, gather information, and change the appearance and functionality of Word Pro. What makes LotusScript a better tool than a macro language is the fact that you can use it with most LotusSuite applications, including the latest releases of Word Pro, Lotus 1-2-3, Notes, Approach, and Freelance Graphics.

The LotusScript language is comprised of two kinds of language elements. To access and control a Lotus application and its documents or files, you must use both of these language elements:

- Common elements which are shared by all Lotus applications:

These elements form the core of the LotusScript language. They provide the basic tools for constructing scripts, such as variables, syntax, statements, keywords, and standard data types. For more information on the core LotusScript language elements, see the LotusScript Language Reference or the [complete on-line listing](#) of core LotusScript language elements.

- Product-defined [classes](#) which are specific to one Lotus application:

These classes are defined as part of a Lotus application. For more information on Lotus Word Pro's product-defined classes, see [Overview: The Word Pro LotusScript Object Model](#).

---

{button ,AL('H\_THE\_WORD\_PRO\_LOTUSSCRIPT\_OBJECT\_MODEL\_OVER;H\_USING\_LOTUSSCRIPT\_IN\_WOR  
D\_PRO\_OVER',0)} [See related topics](#)

'Example: ColorOverride property

'This example script has not yet been created.

'Example: Color property

'This example script has not yet been created.

'Example: ColumnBalance property

'This example script has not yet been created.

'Example: ColumnGap property

'This example script has not yet been created.

'Example: ColumnLayouts property

'This example script has not yet been created.



'Example: ColumnNumber property

'This example script has not yet been created.

'Example: ColumnWidth property

'This example script has not yet been created.

```
'Example: CombineDivisions method
' This example combines two divisions names 'Body' and 'Division' into one
' division.
' RUNTIME DEPENDENCIES: You must have a document open containing these two
' divisions for this script to work.
```

```
Dim DivIdName As String
Dim DivInfoName As String
Dim DivId1 As String
Dim DivId2 As String
```

```
'Get the hexadecimal id for the 'Body' division
DivInfoName = "Body"
Gosub GetDivId
DivId1 = DivIdName
```

```
'Get the hexadecimal id for the 'Division' division
DivInfoName = "Division"
Gosub GetDivId
DivId2 = DivIdName
```

```
'combine these two divisions into one using their hexadecimal ids.
.CombineDivisions DivId1, DivId2
```

```
Exit Sub
```

```
GetDivId:
Forall Div In .ActiveDocument.Divisions
    If (Ucase$(Div.DivisionInfo.Name) = Ucase$(DivInfoName)) Then
        DivIdName = Div.Name
    End If
End Forall
Return
```

```
'Example: CombineSections method
' This example inserts two sections and then combines them into the into the
' current division.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
Forall Section In .Division.Foundry.Sections
    .CombineSections Section.Name
End Forall
```

'Example: CommentColor property

'This example script has not yet been created.

'Example: Company property

'This example script has not yet been created.

```
'Example: CompareFiles method
' This example compares the current file with the file named 'COMPARE.LWP'.
' RUNTIME DEPENDENCIES: You must have a document open and have a file named
' COMPARE.LWP located in the Word Pro default document directory.

FilePath = .ApplicationWindow.UserInterfacePrefs.DocPath & "\COMPARE.LWP"
Print FilePath
FileType = "Lotus Word Pro"
IsMultiDocs = False
IndexOfMultiDocToCompare = 0
.CompareFiles FilePath, FileType, IsMultiDocs, IndexOfMultiDocToCompare
```

'Example: ConditionType property

'This example script has not yet been created.



'Example: Condition property

'This example script has not yet been created.

'Example: Configure method

' This example displays the SmartIcon configuration dialog.

' RUNTIME DEPENDENCIES: None.

.ApplicationWindow.IconBarManager.Configure

'Example: ConnectCells method

' This example creates a table with 5 columns and 4 rows based on the

' Default Table style, selects the first column, and connects the cells.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4

.SelectColumn

**.ConnectCells**

'Example: ConnectContainer method

'This example script has not yet been created.

'Example: ConnectedLayouts property  
'This example script has not yet been created.

'Example: ConnectRows method

' This example creates a table with 4 rows and 5 columns, selects the first

' row, and connects the cells in that row.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4

.SelectRow

**.ConnectRows**

```
'Example: ConnectSectionTabs method
' This example insert two section into the current division and then connects
' the section tabs which creates a new division making the currently selected
' division the child of the new division.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True

.ApplicationWindow.SectionTabs.ConnectSectionTabs
```

```
'Example: Connect method
' This example creates a table with 4 rows and 5 columns, selects the entire
' table, and connects the cells.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.SelectTable
.Table.Connect
```



'Example: ContactUponPermissionDenied property  
'This example script has not yet been created.

'Example: Container property

'This example script has not yet been created.

'Example: ContentHeight property

'This example script has not yet been created.

'Example: ContentLayouts property

'This example script has not yet been created.

'Example: ContentName property

'This example script has not yet been created.

'Example: ContentStyleName property  
'This example script has not yet been created.

'Example: Contents property

'This example script has not yet been created.

'Example: ContentType property

'This example script has not yet been created.



'Example: ContentWidth property

'This example script has not yet been created.

'Example: Content property

'This example script has not yet been created.

'Example: ContextMenuOptions property

'This example script has not yet been created.

'Example: ContinuedFromAlignment property  
'This example script has not yet been created.

'Example: ContinuedFromMessage property  
'This example script has not yet been created.

'Example: ContinuedFromStory property

'This example script has not yet been created.

'Example: ContinuedOnAlignment property

'This example script has not yet been created.

'Example: ContinuedOnMessage property  
'This example script has not yet been created.



'Example: ContinuedOnStory property

'This example script has not yet been created.

'Example: ContractOutlineLevel method

'This example script has not yet been created.

```
'Example: Contract method
' This example creates two child divisions and then contracts and expands the
' divider tabs.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
.ActiveDocWindow.WinViewPrefs.IsViewSectionTabs = True
.ApplicationWindow.SectionTabs.ConnectSectionTabs
.ApplicationWindow.SectionTabs.ConnectSectionTabs
```

```
.ApplicationWindow.SectionTabs.Contract
.ApplicationWindow.SectionTabs.Expand
.ApplicationWindow.SectionTabs.Contract
```

'Example: ConvertOnNew property

'This example script has not yet been created.

'Example: ConvertToClass method

'This example script has not yet been created.

'Example: Copies property

'This example script has not yet been created.

```

'Example: CopyItem method
' This example adds a new menu item name 'New Edit' to the File menu just
' below the Save item. All items from the Edit menu are then copied to
' 'New Menu'
' RUNTIME DEPENDENCIES: You must have not deleted the Edit or File menus
' for this script to work.

Dim MenuName as String
Dim SourceMenu As MenuItem
Dim DestinationMenu As MenuItem
Dim MenuSpacer as String
MenuSpacer = Chr$(8)
MenuName = "&New Edit"

' Set DestinationMenu to the File menu
Set DestinationMenu=.Applicationwindow.LwpMenuBar.Items.Item("&File")

' Set SourceMenu to the Edit Menu
Set SourceMenu = .ApplicationWindow.LwpMenuBar.Items.Item("&Edit")

' Create a new Edit Menu
DestinationMenu.DeleteItem "My Edit"
DestinationMenu.NewItem MenuName,,0,"&Save" & MenuSpacer & "Ctrl+S"

' Copy all the items from the Edit Menu to My new Menu
Forall Items In SourceMenu.Items
    DestinationMenu.Items(MenuName).CopyItem Items, True, ,
End Forall

```

'Example: CopyMeaning method

'This example script has not yet been created.



'Example: CopySelection method

' This example copies the current selection to the clipboard.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ActiveDocument.CopySelection

```
'Example: Copy method  
' This example copies the current division.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim Div As Division  
Set Div = Bind("!Body")  
.Division.Copy Div.Name
```

'Example: CountBlankLines property  
'This example script has not yet been created.

'Example: Count property

'This example script has not yet been created.

```
'Example: CreateDataFile method
' This example creates a data file for the current document.  Two records are
' added and the Merge bar is opened so to insert fields for merging.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateDataFile "~|", "Name~Address~City~State~Zip|", False, "C:\mergedat.lwp"

.MergeAddDataRecord "Jane Doe~100 Main St.~ Atlanta~ GA~30319|"
.MergeAddDataRecord "John Doe~100 Main St.~ Atlanta~ GA~30319|"

.StartFieldInsert
```

```
'Example: CreateDivision method
' This example creates a new division based on the "DEFAULT.MWP" Smart Master.
' It is placed after the current division in the current document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim SmartMaster as String
Dim NewDivId as String
```

```
SmartMaster = .ApplicationWindow.UserInterfacePrefs.StylePath & "\DEFAULT.MWP"
.CreateDivision SmartMaster, "", $LWPDivLocInsertAfterCurrentDiv, "", ""
```

'Example: CreateDocument method

'This example script has not yet been created.

'Example: Created event

'This example script has not yet been created.



'Example: CreateExternalDivision method

' This example creates an external division based on the README95.LWP file.

' It is placed after the current division.

' RUNTIME DEPENDENCIES: The README95.LWP file must be located in the WordPro  
' documents path.

Dim ExternalFilename as String

ExternalFilename = "README95.LWP"

.CreateExternalDivision ExternalFilename, "", \$LwpDivLocInsertAfterCurrentdiv, "", ""

'Example: CreateFrame method

' This example inserts a 1 inch by 1 inch frame into the current document.

' The "Default Frame" style is used which places the upper left corner of the

' frame 1 inch down and 1 inch to the left of the page's upper left corner.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

**.CreateFrame False, "Default Frame",1440, 1440**

'Example: CreateFromBitmap method

'This example script has not yet been created.

'Example: CreateFromClipBrd method

'This example script has not yet been created.

'Example: CreateFromDataObject method

'This example script has not yet been created.

'Example: CreateFromMetafile method

'This example script has not yet been created.

```
'Example: CreateGlossaryEntry method
' This example inserts a glossary entry named NewGlossaryItem for the current
' selection in the specified Glossary file.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "Some stuff for the glossary."
.SelectSentence
.CreateGlossaryEntry "GLOSSARY.GLS", "NewGlossaryItem"
```

'Example: CreateGlossary method

' This example stores a file name in the variable GlossFileName, hides the  
' open documents, opens the default Word Pro glossary file, creates and saves  
' a glossary file named "GLOSTST.GLS" in the User Setup glossary directory,  
' then closes the glossary files and resets the default values user interface  
' preferences.

' RUNTIME DEPENDENCIES: You must have create file rights in the specified  
' glossary directory for this script to work.

Dim GlossFileName As String

GlossFileName = "GLOSTST.GLS"

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = False

.GlossaryOpen GlossFileName, "Lotus Word Pro"

**.CreateGlossary**

.SaveGlossary GlossFileName, "Lotus Word Pro", False

.Close

.ApplicationWindow.UserInterfacePrefs.IsReplacement = False

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = True

.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = False



'Example: CreateGraphic method

' This example creates a Word Pro Drawing frame.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateGraphic "WordProDraw", False

```
'Example: CreateNewButton method
' This example creates a new button to the status bar and then adds text to
' the button. The STATUSBARBUTTONOVERRIDE TEXT is then bound to the
' SetTheButtonText subroutine to set the button text during needs repainting.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ButtonName As String
Dim NewButton As StatusBarButton
With .ApplicationWindow.StatusBar
    ButtonName = .CreateNewbutton (0,0,100,&H1) 'create the new button
    Set NewButton = .StatusBarButtons(ButtonName)
    With .StatusBarButtons(ButtonName)
        .SetOverRideText("New Button...")
        Call .SetButtonText("Button",True)
        .InvalidateButton
        On Event STATUSBARBUTTONOVERRIDE TEXT From NewButton Call
            SetTheButtonText
    End With
    .InvalidateWholeBar ' Force the bar to repaint
End With
End Sub

Sub SetTheButtonText (Source As StatusBarButton, ButtonName As String)
    'Add the the button text each time the status bar needs repainting.
    Source.SetOverRideText("New Button...")
End 2
End Sub
```

'Example: CreateNew method

'This example script has not yet been created.

```
'Example: CreateOleEmbeddedFile method
' This example creates an embedded Word Pro OLE object from the file named
' 'README95.LWP'.
' RUNTIME DEPENDENCIES: You must have a document open and a file named
' 'README95.LWP' located in the default document directory for this script to
' work.
```

```
Dim FilePath As String
```

```
Dim ClassID As String
```

```
Dim IconMetaFilePictHandle As Integer
```

```
FilePath = .ApplicationWindow.UserInterfacePrefs.DocPath & "\README95.LWP"
```

```
ClassID = "{00000000-0000-0000-0000-000000000000}"
```

```
IconMetaFilePictHandle = 0
```

```
.CreateOleEmbeddedFile ClassID, FilePath, IconMetaFilePictHandle
```

```
'Example: CreateOleLinkedFile method
' This example creates an embedded Word Pro OLE object from the file named
' 'README95.LWP'.
' RUNTIME DEPENDENCIES: You must have a document open and a file named
' 'README95.LWP' located in the default document directory for this script to
' work.
```

```
Dim FilePath As String
```

```
Dim ClassID As String
```

```
Dim IconMetaFilePictHandle As Integer
```

```
FilePath = .ApplicationWindow.UserInterfacePrefs.DocPath & "\README95.LWP"
```

```
ClassID = "{00000000-0000-0000-0000-000000000000}"
```

```
IconMetaFilePictHandle = 0
```

```
.CreateOleLinkedFile FilePath, IconMetaFilePictHandle
```

```
'Example: CreateOleNew method
' This example creates a new Lotus Approach OLE object in the current
' document.
' RUNTIME DEPENDENCIES: You must have a document open and have Lotus Approach
' installed for this script to work.
```

```
Dim ClassID As String
```

```
ClassID = "{00028703-0000-0000-c000-000000000046}"
.CreateOleNew(ClassID)
```

'Example: CreateParallelColumns method

' This example creates a parallel column table with 3 columns.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

```
'Example: CreateRemark method
' This example inserts an editorial remark in a version of the current
' document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Print .Division.VersionManager.CurrentVersion.CreateRemark("Test
Remark",835302017,"LOT")
```



'Example: CreateTable method

' This example creates a table with 4 rows and 5 columns.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4

```
'Example: CreateVersion method
' This example creates a version for the current document then deletes the
' version.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Division.VersionManager.CreateVersion "NewVersion"
Forall Version In .ActiveDocument.VersionManager.Versions
    If Version.name = "NewVersion" Then
        .ActiveDocument.VersionManager.DeleteVersion Version.DocVersionId
    End If
End Forall
```

```
'Example: Create method  
' This example creates a new character style.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim StyleName as String  
StyleName = "My New Char Style"  
Style = .Division.Foundry.Create($LwpFoundryCreateTypeStyle, StyleName, 39)  
With .Division.Foundry.CharacterStyles(Style)  
    .Font.Underline = True  
    .Font.FontColor.Blue = 128  
    .Font.FontColor.Red = 128  
    .Font.FontColor.Green = 0  
End With
```

```
'Example: CreationDateString property
With .ActiveDocument
MsgTxt = "Current Word Pro Doc is " & .FullName
MsgTxt = MsgTxt & ", the author is " & .DocInfo.AuthorName
Msgbox MsgTxt,64,"Word Pro Information"
MsgTxt = "It was created on " & .DocInfo.CreationDateString & " at " & .DocInfo.CreationTimeString
Msgbox MsgTxt,64,"Word Pro Information"
End With
```

```
'Example: CreationTimeString property
'With .ActiveDocument
MsgTxt = "Current Word Pro Doc is " & .FullName
MsgTxt = MsgTxt & ", the author is " & .Docinfo.AuthorName
Msgbox MsgTxt,64,"Word Pro Information"
MsgTxt = "It was created on " & .DocInfo.CreationDateString & " at "
& .DocInfo.CreationTimeString
Msgbox MsgTxt,64,"Word Pro Information"
End With
```

'Example: CreationTimeValue property

'This example script has not yet been created.

'Example: Crop property

'This example script has not yet been created.

'Example: CurrentCell property

'This example script has not yet been created.



'Example: CurrentColumn property

'This example script has not yet been created.

'Example: CurrentEditor property

'This example script has not yet been created.

'Example: CurrentLanguage property

'This example script has not yet been created.

'Example: CurrentRow property

'This example script has not yet been created.

'Example: CurrentVersion property

'This example script has not yet been created.

'Example: CustomLength property

'This example script has not yet been created.

'Example: CustomViewLevel property  
'This example script has not yet been created.

```
'Example: CutSelection method
' This example inserts some text into the current document.  The text is then
' selected and cut to the clipboard.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "This is some sample text."
.Text.Select $LwpSelectObjectTypeParagraph
.CutSelection
```



```
'Example: DarkMode method
' This example turns on DarkMode which prevents the screen from updating.
' Some text is inserted, a table is created and then DarkMode is turned off.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.Darkmode True
For t =1 To 5
    .Type ("Hello world [Enter]")
Next
.CreateTable False, "Default Table", 4, 2
.ApplicationWindow.DarkMode False
```

'Example: DataFileFieldNames property  
'This example script has not yet been created.

'Example: DataFileName property

'This example script has not yet been created.

'Example: DataNames property

'This example script has not yet been created.

'Example: DataObjectGetDataHere method  
'This example script has not yet been created.

'Example: DataObjectGetData method

'This example script has not yet been created.

'Example: DateCreatedValue property

'This example script has not yet been created.

'Example: DateRevisedValue property

'This example script has not yet been created.



```
'Example: DbUnderline method
' This example toggles the double underline attribute of the selected text.
' RUNTIME DEPENDENCIES: You must have a document open and some text selected
' for this script to work.
```

```
.DbUnderline
MessageBox "Click OK undo double underline change.",MB_OK,"Example Script"
.DbUnderline
```

'Example: DdeLinkManager property

'This example script has not yet been created.

'Example: DdeLinksFromMarker property

'This example script has not yet been created.

'Example: DdeLinks property

'This example script has not yet been created.

'Example: DdeOutboundInfo property

'This example script has not yet been created.

## Word Pro: ContentName property

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CEL  
LAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISIONINFO_  
CLASS;H_DROPCAPCONTAINER_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_F  
OOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_FRAMEGROUPL  
AYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO  
UT_CLASS;H_NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELA  
YOUT_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGROUPLAYOU  
T_CLASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYLAYOUT_CLASS;H_SUBPAGECO  
NTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABL  
EGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEHEADIN  
GLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;H_TOCSUPERTABLELAYOUT_C  
LASS',0)} See list of classes
```

```
{button ,AL('H_CONTENTNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns the name of the content object in any container.

## Data Type

String

## Syntax

contentnamevalue = [objectreference].ContentName

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

## Word Pro: ContentStyleName property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_CONTENTSTYLENAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The name of the initial paragraph style used by certain layout objects.

### Data Type

String

### Syntax

contentstylevalue = [objectreference].ContentStyleName

[objectreference].ContentStyleName = contentstylevalue

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage

This property stores the name of the initial paragraph style used by a layout object. This is not the same as the paragraph style assigned to the currently active text object. To access the name of the paragraph style assigned to the current text object, check the text object's ParagraphStyleName property.

The ContentStyleName property is equivalent to the "Initial paragraph style" setting located in the Miscellaneous panel of the InfoBox for certain layout objects.

## Word Pro: Contents property

{button ,AL('H\_DOCINFOFIELD\_CLASS;H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[DocInfoField]

Assigns a string value to a DocInfo field.

[Foundry]

An object created from the ContentCollection class. This object provides access to Content objects.

## Data Type

[\[DocInfoField\]](#)

String

[Foundry]

ContentCollection

## Syntax

contentsvalue = [objectreference].Contents

[objectreference].Contents = contentsvalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

[DocInfoField]

When you add a DocInfo field, you assign a name and a value to it. The content is the value of a DocInfo field and the content is always a String value. For example, if you add a DocInfo field and assign the name "Client" and the value "Active," then the contents would be "Active." To change the value of the "Client" DocInfo field from "Active" to "Inactive," you would assign the string value "Inactive" to the Contents property.

[Foundry]

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Content objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the Content objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the Content objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the Content objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: WordPro LotusScript CollectionClasses](#).



## Word Pro: ContentType property

{button ,AL(^H\_BASSETABLE\_CLASS;H\_CONTENT\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FORMULA\_CLASS;H\_GLOSSARY\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS;H\_PARALLEL\_COLUMNS\_CLASS;H\_SUPERTABLE\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CONTENTTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates the type of content object in which the insertion point is located.

### Data Type

Variant

### Syntax

contenttypevariable = [objectreference].ContentType

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpContentTypeText (161)	Indicates that the content type is a Text object.
\$LwpContentTypeTableonly (162)	Indicates that the content type is a TableOnly object.
\$LwpContentTypeGraphic (163)	Indicates that the content type is a Graphic object.
\$LwpContentTypeOleobject (164)	Indicates that the content type is a OleObject.
\$LwpContentTypeParallelcolumns (165)	Indicates that the content type is a ParallelColumns object.
\$LwpContentTypeFootnotetable (166)	Indicates that the content type is a FootnoteTable object.
\$LwpContenttTypeFormula (167)	Indicates that the content type is a Formula object.
\$LwpContentTypeFormula (2056)	Indicates that the content type is a Formula object.

### Usage

Using this property allows you to verify a specific type of content object in a layout. For example, you can use this property to determine whether or not a certain frame contains a graphic object.

## Word Pro: ContentWidth property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTENTWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the width of the content based on its rotation within a container.

### Data Type

Long

### Syntax

contentwidthvalue = [objectreference].ContentWidth

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## **Word Pro: ContinuedFromAlignment property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTINUEDFROMALIGNMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

AlignmentType

### **Syntax**

continuedfromalignmentvalue = [objectreference].ContinuedFromAlignment

[objectreference].ContinuedFromAlignment = continuedfromalignmentvalue

### **Legal values**

\$LtsAlignmentHorizCenter (1056964611)

\$LtsAlignmentJustify (1056964613)

\$LtsAlignmentLeft (1056964609)

\$LtsAlignmentRight (1056964610)

\$LtsAlignmentSmart (1056964612)

\$LwpAlignmentTypeAlignRevert (8)

\$LwpAlignmentTypeJustifyall (5)

\$LwpAlignmentTypeNumericleft (6)

\$LwpAlignmentTypeNumericright (7)

### **Usage**

## **Word Pro: ContinuedFromMessage property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTINUEDFROMMESSAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

continuedfrommessagevalue = [objectreference].ContinuedFromMessage

[objectreference].ContinuedFromMessage = continuedfrommessagevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: ContinuedFromStory property

{button ,AL(`H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CONTINUEDFROMSTORY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the Text object that contains the "Continued from" message for a Division object.

### Data Type

[String](#)

### Syntax

continuedfromstoryvalue = [objectreference].ContinuedFromStory

### Legal values

### Usage

When a footnote is too large to fit in the footnote space provided, Word Pro automatically flows the footnote to the footnote space on the next page. Word Pro also adds a "Continued on" message to the first page and a "Continued from" message on the next page. These messages do not vary within a division, but can vary from one division to the next. Each "Continued from" message is a Text object that you can manipulate in a script. This ContinuedFromStory property stores the name of the Continued From Text object for the specified Division object.

Word Pro does not use this property in the WPAApplication.AppFoundry, WPAApplication.TempFoundry, or TextDocument.Foundry properties.

## **Word Pro: ContinuedOnAlignment property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTINUEDONALIGNMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

AlignmentType

### **Syntax**

continuedonalignmentvalue = [objectreference].ContinuedOnAlignment

[objectreference].ContinuedOnAlignment = continuedonalignmentvalue

### **Legal values**

\$LtsAlignmentHorizCenter (1056964611)

\$LtsAlignmentJustify (1056964613)

\$LtsAlignmentLeft (1056964609)

\$LtsAlignmentRight (1056964610)

\$LtsAlignmentSmart (1056964612)

\$LwpAlignmentTypeAlignRevert (8)

\$LwpAlignmentTypeJustifyall (5)

\$LwpAlignmentTypeNumericleft (6)

\$LwpAlignmentTypeNumericright (7)

### **Usage**

**Word Pro: ContinuedOnMessage property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CONTINUEDONMESSAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

continuedonmessagevalue = [objectreference].ContinuedOnMessage

[objectreference].ContinuedOnMessage = continuedonmessagevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: ContinuedOnStory property

{button ,AL(`H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CONTINUEDONSTORY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the Text object that contains the "Continued on" message for a Division object.

### Data Type

String

### Syntax

continuedonstoryvalue = [objectreference].ContinuedOnStory

### Legal values

### Usage

When a footnote is too large to fit in the footnote space provided, Word Pro automatically flows the footnote to the footnote space on the next page. Word Pro also adds a "Continued on" message to the first page and a "Continued from" message on the next page. These messages do not vary within a division, but can vary from one division to the next. Each "Continued on" message is a Text object that you can manipulate in a script. This ContinuedOnStory property stores the name of the Continued On Text object for the specified Division object.

Word Pro does not use this property in the WPAApplication.AppFoundry, WPAApplication.TempFoundry, or TextDocument.Foundry properties.



**Word Pro: ConvertOnNew property**

{button ,AL(^H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CONVERTONNEW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

()

**Data Type**

[String](#)

**Syntax****Legal values****Usage**

## **Word Pro: Copies property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COPIES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to set the number of copies you want to print.

### **Data Type**

[Integer](#)

### **Syntax**

copiesvalue = [objectreference].Copies

[objectreference].Copies = copiesvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing File - Print and selecting a number from the "Number of copies" box.

## **Word Pro: CountBlankLines property**

{button ,AL('H\_LINENUMBEROPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_COUNTBLANKLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

countblanklinesvalue = [objectreference].CountBlankLines

[objectreference].CountBlankLines = countblanklinesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: Count property

```
{button ,AL(^H_BAGCOLLECTION_CLASS;H_BASECOLLECTION_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_CELLCOLLECTION_CLASS;H_CELLLAYOUTCOLLECTION_CLASS;H_CHARACTERSTYLECOLLECTION_CLASS;H_CLICKHERECOLLECTION_CLASS;H_CONNECTEDLAYOUTCOLLECTION_CLASS;H_CONTENTCOLLECTION_CLASS;H_DDELINKCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCINFOFIELDCOLLECTION_CLASS;H_DOCUMENTS_CLASS;H_DOCWINDOWCOLLECTION_CLASS;H_DROPCAPLAYOUTCOLLECTION_CLASS;H_EDITORCOLLECTION_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECTION_CLASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTION_CLASS;H_FRAMELAYOUTCOLLECTION_CLASS;H_GLOSSARYCOLLECTION_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHICOLEBJECTCOLLECTION_CLASS;H_GROUPLAYOUTCOLLECTION_CLASS;H_HEADERLAYOUTCOLLECTION_CLASS;H_ICONBARCOLLECTION_CLASS;H_LAYOUTCOLLECTION_CLASSES;H_MARKERCOLLECTION_CLASS;H_MENUITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLASSES;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLINESEQCOLLECTION_CLASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELAYOUTCOLLECTION_CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARRALLELCOLSCOLLECTION_CLASS;H_POWERFIELDCOLLECTION_CLASS;H_ROWLAYOUTCOLLECTION_CLASSES;H_RUBYLAYOUTCOLLECTION_CLASS;H_SECTIONCOLLECTION_CLASS;H_SILVERBULLETCOLLECTION_CLASS;H_SMARTCORRECTCOLLECTION_CLASS;H_SMARTFILLCOLLECTION_CLASS;H_STATUSBARBUTTONCOLLECTION_CLASS;H_STRINGCOLLECTION_CLASS;H_SUPERTABLECOLLECTION_CLASS;H_SUPERTABLELAYOUTCOLLECTION_CLASS;H_TABLECOLLECTION_CLASS;H_TABLEHEADINGCOLLECTION_CLASSES;H_TABLEHEADINGLAYOUTCOLLECTION_CLASS;H_TABLELAYOUTCOLLECTION_CLASS;H_TABLEMARKERCOLLECTION_CLASS;H_TABLEONLYCOLLECTION_CLASS;H_TEXTCOLLECTION_CLASS;H_TEXTMARKERCOLLECTION_CLASS;H_TEXTSTYLECOLLECTION_CLASS;H_VERSIONCOLLECTION_CLASS;H_WPDATASETCOLLECTION_CLASS',0)} See list of classes
```

```
{button ,AL(^H_COUNT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only)

## Data Type

Long

## Syntax

countvalue = [objectreference].Count

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

## **Word Pro: CreationDateString property**

{button ,AL('H\_DOCINFO\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATIONDATESTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the date the document was created as a String value.

### **Data Type**

String

### **Syntax**

creationdatestringvalue = [objectreference].CreationDateString

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

The format of the CreationDateString property is determined by the Windows Short date style.

**Word Pro: CreationTimeString property**

{button ,AL('H\_DOCINFO\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATIONTIMESTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the time the document was created as a String value.

**Data Type**

[String](#)

**Syntax**

creationtimestringvalue = [objectreference].CreationTimeString

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The format of the CreationTimeString property is determined by the Windows Time style.

**Word Pro: CreationTimeValue property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATIONTIMEVALUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read only) Returns the time the document was created as a Long data type.

**Data Type**

Long

**Syntax**

creationtimevaluevalue = [objectreference].CreationTimeValue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The value that this property returns represents the number of seconds that have elapsed since midnight on January 1, 1970.

## **Word Pro: Crop property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CROP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prints fine lines that indicate the corners of the page.

### **Data Type**

Integer

### **Syntax**

cropvalue = [objectreference].Crop

[objectreference].Crop = cropvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This property enables you to crop any page you specify in a document. Crop marks print offset .5 inches from the top left corner of the physical page. Equivalent to choosing File - Print, clicking Options, and selecting "Crop marks."



## Word Pro: CurrentLanguage property

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CURRENTLANGUAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates the text's language at the insertion point. If text is selected and more than one language is assigned to the selected text, Word Pro returns the language for the text nearest the insertion point.

### Data Type

Variant (Enumerated)

### Syntax

currentlanguagevariable = [objectreference].CurrentLanguage

### Legal values

\$LwpLanguagesAfrikaans (474)

\$LwpLanguagesAlbanian (475)

\$LwpLanguagesAmerican (442)

\$LwpLanguagesArabic (1025)

\$LwpLanguagesArabicAlgeria (5121)

\$LwpLanguagesArabicBahrain (15361)

\$LwpLanguagesArabicEgypt (3073)

\$LwpLanguagesArabicIraq (2049)

\$LwpLanguagesArabicJordan (11265)

\$LwpLanguagesArabicKuwait (13313)

\$LwpLanguagesArabicLebanon (12289)

\$LwpLanguagesArabicLibya (4097)

\$LwpLanguagesArabicMorocco (6145)

\$LwpLanguagesArabicOman (8193)

\$LwpLanguagesArabicQatar (16385)

\$LwpLanguagesArabicSyria (10241)

\$LwpLanguagesArabicTunisia (7169)

\$LwpLanguagesArabicUAE (14337)

\$LwpLanguagesArabicYemen (9217)

\$LwpLanguagesAustralian (444)

\$LwpLanguagesBasque (1069)

\$LwpLanguagesBrazilian (468)

\$LwpLanguagesBritish (443)

\$LwpLanguagesBritishize (12297)

\$LwpLanguagesBritishmedize (13321)

\$LwpLanguagesBrmedical (11273)

\$LwpLanguagesBulgarian (478)

\$LwpLanguagesByelorussian (1059)

\$LwpLanguagesCatalan (436)

\$LwpLanguagesChineseHongKong (3076)

\$LwpLanguagesChinesePRChina (481)

\$LwpLanguagesChineseSingapore (4100)

\$LwpLanguagesChineseTraditional (479)

\$LwpLanguagesCroatian (1050)

\$LwpLanguagesCroatianCyrillic (2074)

\$LwpLanguagesCroatianLatin (1050)  
\$LwpLanguagesCroatianSerbian (3098)  
\$LwpLanguagesCzech (437)  
\$LwpLanguagesDanish (438)  
\$LwpLanguagesDutch (439)  
\$LwpLanguagesDutchBelgian (440)  
\$LwpLanguagesEnglishCanadian (445)  
\$LwpLanguagesEnglishCaribbean (9225)  
\$LwpLanguagesEnglishIreland (447)  
\$LwpLanguagesEnglishJamaica (8201)  
\$LwpLanguagesEnglishNewzealand (446)  
\$LwpLanguagesEnglishSAfrica (7177)  
\$LwpLanguagesEstonian (1061)  
\$LwpLanguagesFaeroese (1080)  
\$LwpLanguagesFarsi (1081)  
\$LwpLanguagesFinnish (452)  
\$LwpLanguagesFrench (453)  
\$LwpLanguagesFrenchBelgian (454)  
\$LwpLanguagesFrenchCanadian (455)  
\$LwpLanguagesFrenchLuxembourg (5132)  
\$LwpLanguagesFrenchSwiss (456)  
\$LwpLanguagesGerman (457)  
\$LwpLanguagesGermanAustrian (459)  
\$LwpLanguagesGermanLiechtenstein (5127)  
\$LwpLanguagesGermanLuxembourg (4103)  
\$LwpLanguagesGermanSwiss (458)  
\$LwpLanguagesGreek (460)  
\$LwpLanguagesHebrew (483)  
\$LwpLanguagesHungarian (461)  
\$LwpLanguagesIcelandic (484)  
\$LwpLanguagesIndonesian (1057)  
\$LwpLanguagesItalian (462)  
\$LwpLanguagesItalianSwiss (463)  
\$LwpLanguagesJapanese (485)  
\$LwpLanguagesKorean (486)  
\$LwpLanguagesKoreanJohab (2066)  
\$LwpLanguagesLatvian (1062)  
\$LwpLanguagesLithuanian (1063)  
\$LwpLanguagesMedical (448)  
\$LwpLanguagesNorwegian (464)  
\$LwpLanguagesNynorsk (465)  
\$LwpLanguagesPolish (466)  
\$LwpLanguagesPortuguese (467)  
\$LwpLanguagesRhaetoRoman (487)  
\$LwpLanguagesRomanian (488)  
\$LwpLanguagesRussian (469)  
\$LwpLanguagesRussianio (470)

\$LwpLanguagesSlovak (492)  
\$LwpLanguagesSlovene (493)  
\$LwpLanguagesSorbian (1070)  
\$LwpLanguagesSpanish (471)  
\$LwpLanguagesSpanishArgentina (11274)  
\$LwpLanguagesSpanishBolivia (16394)  
\$LwpLanguagesSpanishChile (13222)  
\$LwpLanguagesSpanishColombia (9226)  
\$LwpLanguagesSpanishCostaRica (5130)  
\$LwpLanguagesSpanishDominican (7178)  
\$LwpLanguagesSpanishEcuador (12298)  
\$LwpLanguagesSpanishGuatemala (4106)  
\$LwpLanguagesSpanishMexican (2058)  
\$LwpLanguagesSpanishPanama (6154)  
\$LwpLanguagesSpanishParaguay (15370)  
\$LwpLanguagesSpanishPeru (10250)  
\$LwpLanguagesSpanishUruguay (14346)  
\$LwpLanguagesSpanishVenezuela (8202)  
\$LwpLanguagesSwedish (473)  
\$LwpLanguagesSystem (434)  
\$LwpLanguagesThai (494)  
\$LwpLanguagesTurkish (495)  
\$LwpLanguagesUkrainian (496)  
\$LwpLanguagesUniversal (435)  
\$LwpLanguagesUrdu (497)  
\$LwpLanguagesVoorkeur (441)

## **Usage**

## Word Pro: CustomLength property

{button ,AL(^H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_CUSTOMLENGTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Long

### Syntax

customlengthvalue = [objectreference].CustomLength

[objectreference].CustomLength = customlengthvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

**Word Pro: CustomViewLevel property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CUSTOMVIEWLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

customviewlevelvalue = [objectreference].CustomViewLevel

[objectreference].CustomViewLevel = customviewlevelvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: DataFileFieldNames property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATAFILEFIELDNAMES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The names of merge fields in the Merge data file.

**Data Type**

[String](#)

**Syntax**

datafilefieldnamesvalue = [objectreference].DataFileFieldNames

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to the field names on a new or existing Merge data file.

**Word Pro: DataFileName property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATAFILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The path and name of the Merge data file.

**Data Type**

String

**Syntax**

datafilenamevalue = [objectreference].DataFileName

[objectreference].DataFileName = datafilenamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to the path and file name of a new or existing Merge data file.

**Word Pro: DateCreatedValue property**

{button ,AL('H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATECREATEDVALUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Long](#)

**Syntax**

datecreatedvaluevalue = [objectreference].DateCreatedValue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**



## **Word Pro: DateRevisedValue property**

{button ,AL('H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATEREVISIEDVALUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

daterevisedvaluevalue = [objectreference].DateRevisedValue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DdeOutboundInfo property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DDEOUTBOUNDINFO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

ddeoutboundinfovalue = [objectreference].DdeOutboundInfo

[objectreference].DdeOutboundInfo = ddeoutboundinfovalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: DebugVariable property**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEBUGVARIABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

debugvariablevalue = [objectreference].DebugVariable

[objectreference].DebugVariable = debugvariablevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: DefaultBinName property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTBINNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

defaultbinnamevalue = [objectreference].DefaultBinName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultCellStyleDescription property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTCELLSTYLEDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

defaultcellstyledescriptionvalue = [objectreference].DefaultCellStyleDescription

[objectreference].DefaultCellStyleDescription = defaultcellstyledescriptionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultColumnName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTCOLUMNSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

defaultcolumnstylevalue = [objectreference].DefaultColumnName

[objectreference].DefaultColumnName = defaultcolumnstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultFilePath property**

{button ,AL('H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTFILEPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the path for the current working directory for Word Pro. Identical to the Location property on WPAApplication.

### **Data Type**

String

### **Syntax**

defaultfilepathvalue = [objectreference].DefaultFilePath

### **Legal values**

The value of this property cannot be set by a script.

### **Usage**

If you use LotusScript to create a file without specifying a directory for that file, Word Pro will store that file in the current working directory.

Under Windows 3.1, you could see and specify a working directory in the Properties dialog box for a particular executable. Windows95 does not offer a user interface for working directories, but the concept of a working directory is still in use. In Word Pro, the working directory is always the same as the directory in which the Word Pro executable is installed.

## **Word Pro: DefaultFrameStyleDescription property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTFRAMESTYLEDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

defaultframestyledescriptionvalue = [objectreference].DefaultFrameStyleDescription

[objectreference].DefaultFrameStyleDescription = defaultframestyledescriptionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: DefaultLatinFont property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTLATINFONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

defaultlatinfontvalue = [objectreference].DefaultLatinFont

[objectreference].DefaultLatinFont = defaultlatinfontvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: DefaultLeftColumnName property**

{button ,AL('H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTLEFTCOLUMNSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the default style for the left column of a parallel column.

### **Data Type**

[String](#)

### **Syntax**

defaultleftcolumnstylevalue = [objectreference].DefaultLeftColumnName

[objectreference].DefaultLeftColumnName = defaultleftcolumnstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultNonLatinFont property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTNONLATINFONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

defaultnonlatinfontvalue = [objectreference].DefaultNonLatinFont

[objectreference].DefaultNonLatinFont = defaultnonlatinfontvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultPageHeight property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTPAGEHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

defaultpageheightvalue = [objectreference].DefaultPageHeight

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: DefaultPageStyleDescription property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTPAGESTYLEDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

defaultpagestyledescriptionvalue = [objectreference].DefaultPageStyleDescription

[objectreference].DefaultPageStyleDescription = defaultpagestyledescriptionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultPageWidth property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTPAGEWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

defaultpagewidthvalue = [objectreference].DefaultPageWidth

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: DefaultPitch property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTPITCH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

defaultpitchvalue = [objectreference].DefaultPitch

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultRightColumnName property**

{button ,AL('H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTRIGHTCOLUMNSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the default style for the right column of a parallel column.

### **Data Type**

[String](#)

### **Syntax**

defaultrightcolumnstylevalue = [objectreference].DefaultRightColumnName

[objectreference].DefaultRightColumnName = defaultrightcolumnstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: DefaultTableStyleDescription property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTTABLESTYLEDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

defaulttablestyledescriptionvalue = [objectreference].DefaultTableStyleDescription

[objectreference].DefaultTableStyleDescription = defaulttablestyledescriptionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DefaultTextStyleDescription property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFAULTTEXTSTYLEDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

defaulttextstyledescriptionvalue = [objectreference].DefaultTextStyleDescription

[objectreference].DefaultTextStyleDescription = defaulttextstyledescriptionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: DefCellStyleName property**

{button ,AL(`H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DEFCELLSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default style layout assigned to a cell object.

### **Data Type**

String

### **Syntax**

defcellstylevalue = [objectreference].DefCellStyleName

[objectreference].DefCellStyleName = defcellstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Allows you to specify the layout style of cell layout objects which are created after you set this property. For example, if you create a table, the "Default Cell" style will be assigned to all cells within the table. If you modify the DefCellStyleName property of the table, any new cell layout objects that are created will use the cell layout style that you specify.

### **Word Pro: DefColWidth property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DEFCOLWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default width of column objects inserted within a table object.

### **Data Type**

Long

### **Syntax**

defcolwidthvalue = [objectreference].DefColWidth

[objectreference].DefColWidth = defcolwidthvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

Use this property to determine the default width, or to reset the default width of a column object.

## Word Pro: Definition property

{button ,AL(^H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_CHARACTERSTYLE\_CLASS;H\_COLU  
MNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELA  
YOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLA  
SS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;  
H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_ROWGROUPLAYOUT  
\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SU  
PERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTA  
BLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DEFINITION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

CharacterStyle includes attributes, facename, size info, and so on (look in hierarchy box when creating a character style).

ParagraphStyle includes attributes and things that comprise the style (look in hierarchy box when creating a paragraph style).

[Layout]

This property indicates which style properties are being received from another style, based on the style hierarchy.

## Data Type

The data type for this parameter is [Long](#) which allows the value of this parameter to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

## Syntax

definitionvalue = [objectreference].Definition

## Legal values

[Layout]

LwpLayStyOverSize (&H1)

LwpLayStyOverPlacement (&H2)

LwpLayStyOverMargins (&H4)

LwpLayStyOverBorders (&H8)

LwpLayStyOverBackground (&H10)

LwpLayStyOverJoins (&H20)

LwpLayStyOverShadow (&H40)

LwpLayStyOverChildren (&H10000)

LwpLayStyOverColumns (&H400)

LwpLayStyOverContents (&H20000)

LwpLayStyOverLeaders (&H2000)

LwpLayStyOverMisc (&H8000)

LwpLayStyOverNumerics (&H200)

LwpLayStyOverOrientation (&H4000)

LwpLayStyOverRotation (&H1000)

LwpLayStyOverScaling (&H800)

LwpLayStyOverScript (&H100)

LwpLayStyOverSizeAndPlacement (&H3)

LwpLayStyOverTabs (&H80)

## Usage

[Layout]

Layout objects such as frames, table cells, and pages are based on styles. For example, when you create a frame in Word Pro, it is usually based on a style called "Default Frame." When you create a table in Word Pro, the table is

usually based on a style called "Default Table," and the table cells are usually based on a style called "Default Cell." When you create a style in Word Pro, it is always based on an existing style. This is known as style hierarchy. Style hierarchy is represented in Word Pro in the style panel of the InfoBox for layout objects. If you choose Create Style from the Style panel, then choose Hierarchy, you'll see the list of available style properties.

The Definition property of a layout object represents all of the style properties which come from a style that is higher up in the style hierarchy. In the Style Hierarchy Definition dialog, these are represented as unchecked style properties.

**Word Pro: DefRowHeight property**

{button ,AL('H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEFROWHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default height of row objects inserted within a table object.

**Data Type**

Long

**Syntax**

defrowheightvalue = [objectreference].DefRowHeight

[objectreference].DefRowHeight = defrowheightvalue

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

Use this property to determine the default height, or to reset the default height of a row object.

### **Word Pro: DemandLoad property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEMANDLOAD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

### **Data Type**

Variant (Enumerated)

ReservedParam

### **Syntax**

[objectreference].DemandLoad = demandloadvalue

### **Legal values**

\$LwpReservedParamDefault (1707)

### **Usage**



### **Word Pro: Descent property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DESCENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Points](#)

### **Syntax**

descentvalue = [objectreference].Descent

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DescriptionFileName property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DESCRIPTIONFILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) If you choose DescriptionFile when opening an external data file from another application, Word Pro stores the field names and separator of that file in a separate description file. The name of that file is stored in this property.

### **Data Type**

String

### **Syntax**

descriptionfilenamevalue = [objectreference].DescriptionFileName

[objectreference].DescriptionFileName = descriptionfilenamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Not required. Word Pro uses the DescriptionFile to get the separator and field names for an external data file.

## **Word Pro: Description property**

{button ,AL('H\_BASEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The description of an object. This property is defined in the BaseObject class and inherited by all Word Pro objects. It is not, however, used by all Word Pro objects. Objects that make use of this property usually use it to store a description of the object.

### **Data Type**

String

### **Syntax**

descriptionvalue = [objectreference].Description

[objectreference].Description = descriptionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

You will most likely use this property in a TextDocument object. When you call this property from a TextDocument object, it returns the value of the Description field for the document represented by that object. When you call this property from the WPAApplication object, it returns the value of the Description field for the currently active document. Most other objects do not use this description property.

## Word Pro: DirectionDown property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_DIRECTIONDOWN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Specifies whether a layout will autogrow or autosize in a downward direction.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

## Syntax

directiondownvalue = [objectreference].DirectionDown

[objectreference].DirectionDown = directiondownvalue

## Legal values

<u>Value</u>	<u>Effect</u>
LwpLayoutDirectionUseDirection (&H1)	Setting this value specifies that a layout object will adhere to the other bit values set for this property.
LwpLayoutDirectionAutoGrow (&H2)	Setting this value specifies that a layout object autogrows in a downward direction.
LwpLayoutDirectionAutoSize (&H4)	Setting this value specifies that a layout object autosizes in a downward direction.
LwpLayoutDirectionTocontainer (&H8)	Setting this value specifies that a layout object autogrows or autosizes to the bottom edge of its container.
LwpLayoutDirectionAllDirections (&HF)	Setting this value simultaneously autogrows or autosizes a layout object in all directions (down, up, left, and right).

## Usage

Make sure to set the &H1 bit value if you want to enable any of the grow or size options for a layout object. For example, if you want to specify that a layout object should automatically grow, it would not be sufficient to set the property value to &H2. You must also set the &H1 bit in order for the other bit settings to be valid. An example statement is shown below:

```
[objectreference].DirectionDown = &H1 OR &H2
```

## Word Pro: DirectionLeft property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_DIRECTIONLEFT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether a layout will autogrow or autosize toward the left.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

## Syntax

```
directionleftvalue = [objectreference].DirectionLeft
```

```
[objectreference].DirectionLeft = directionleftvalue
```

## Legal values

<u>Value</u>	<u>Effect</u>
LwpLayoutDirectionUseDirection (&H1)	Setting this value specifies that a layout object will adhere to the other bit values set for this property.
LwpLayoutDirectionAutoGrow (&H2)	Setting this value specifies that a layout object autogrows toward the left.
LwpLayoutDirectionAutoSize (&H4)	Setting this value specifies that a layout object autosizes toward the left.
LwpLayoutDirectionTocontainer (&H8)	Setting this value specifies that a layout object autogrows or autosizes to the left edge of its container.
LwpLayoutDirectionAllDirections (&HF)	Setting this value simultaneously autogrows or autosizes a layout object in all directions (down, up, left, and right).

## Usage

Make sure to set the &H1 bit value if you want to enable any of the grow or size options for a layout object. For example, if you want to specify that a layout object should automatically grow, it would not be sufficient to set the property value to &H2. You must also set the &H1 bit in order for the other bit settings to be valid. An example statement is shown below:

```
[objectreference].DirectionLeft = &H1 OR &H2
```

## Word Pro: DirectionRight property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_DIRECTIONRIGHT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether a layout will autogrow or autosize toward the right.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

## Syntax

```
directionrightvalue = [objectreference].DirectionRight
```

```
[objectreference].DirectionRight = directionrightvalue
```

## Legal values

<u>Value</u>	<u>Effect</u>
LwpLayoutDirectionUseDirection (&H1)	Setting this value specifies that a layout object will adhere to the other bit values set for this property.
LwpLayoutDirectionAutoGrow (&H2)	Setting this value specifies that a layout object autogrows toward the right.
LwpLayoutDirectionAutoSize (&H4)	Setting this value specifies that a layout object autosizes toward the right.
LwpLayoutDirectionTocontainer (&H8)	Setting this value specifies that a layout object autogrows or autosizes to the right edge of its container.
LwpLayoutDirectionAllDirections (&HF)	Setting this value simultaneously autogrows or autosizes a layout object in all directions (down, up, left, and right).

## Usage

Make sure to set the &H1 bit value if you want to enable any of the grow or size options for a layout object. For example, if you want to specify that a layout object should automatically grow, it would not be sufficient to set the property value to &H2. You must also set the &H1 bit in order for the other bit settings to be valid. An example statement is shown below:

```
[objectreference].DirectionRight = &H1 OR &H2
```

## Word Pro: DirectionUp property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERGROUPLAYOUT_CLASS;H_SUPERLAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERGROUPLAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_DIRECTIONUP_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether a layout will autogrow or autosize in an upward direction.

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

## Syntax

directionupvalue = [objectreference].DirectionUp

[objectreference].DirectionUp = directionupvalue

## Legal values

<u>Value</u>	<u>Effect</u>
LwpLayoutDirectionUseDirection (&H1)	Setting this value specifies that a layout object will adhere to the other bit values set for this property.
LwpLayoutDirectionAutoGrow (&H2)	Setting this value specifies that a layout object autogrows in a upward direction.
LwpLayoutDirectionAutoSize (&H4)	Setting this value specifies that a layout object autosizes in a upward direction.
LwpLayoutDirectionTocontainer (&H8)	Setting this value specifies that a layout object autogrows or autosizes to the top edge of its container.
LwpLayoutDirectionAllDirections (&HF)	Setting this value simultaneously autogrows or autosizes a layout object in all directions (down, up, left, and right).

## Usage

Make sure to set the &H1 bit value if you want to enable any of the grow or size options for a layout object. For example, if you want to specify that a layout object should automatically grow, it would not be sufficient to set the property value to &H2. You must also set the &H1 bit in order for the other bit settings to be valid. An example statement is shown below:

```
[objectreference].DirectionUp = &H1 OR &H2
```

## **Word Pro: DisableClickHeres property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DISABLECLICKHERES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to edit Click Here prompt text on-screen.

### **Data Type**

[Integer](#)

### **Syntax**

disableclickheresvalue = [objectreference].DisableClickHeres

[objectreference].DisableClickHeres = disableclickheresvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing File - TeamSecurity, and selecting "Edit Click Here Block prompts on-screen" on the Other Protection panel.



**Word Pro: DisableExportToNotes property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DISABLEEXPORTTONOTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prevents information in a document from being exported to Notes.

**Data Type**

[Integer](#)

**Syntax**

disableexporttonotesvalue = [objectreference].DisableExportToNotes

[objectreference].DisableExportToNotes = disableexporttonotesvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to choosing File - TeamSecurity, and selecting "Disable Notes/FX of TeamSecurity fields" on the Other Protection panel.

## **Word Pro: DisableVersionReview property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DISABLEVERSIONREVIEW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prevents a user from viewing versions other than the current version of the document.

### **Data Type**

[Integer](#)

### **Syntax**

disableversionreviewvalue = [objectreference].DisableVersionReview

[objectreference].DisableVersionReview = disableversionreviewvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

A Word Pro document can have multiple editors who can view each other's version of a document. However, you can set DisableVersionReview to restrict an editor to a specific version of the document.

Equivalent to choosing File - TeamSecurity, and selecting "Disable version review" on the Other Protection panel.

## **Word Pro: DistanceFromLeftMargin property**

{button ,AL('H\_LINENUMBEROPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DISTANCEFROMLEFTMARGIN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Long](#)

### **Syntax**

distancefromleftmarginvalue = [objectreference].DistanceFromLeftMargin

[objectreference].DistanceFromLeftMargin = distancefromleftmarginvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## Word Pro: DivisionName property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIVISIONNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the external user name for a division object.

## Data Type

String

## Syntax

divisionnamevalue = [objectreference].DivisionName

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

You use this property to return the name that displays on a division tab.

### **Word Pro: DivisionsRequired property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIVISIONSREQUIRED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

divisionsrequiredvalue = [objectreference].DivisionsRequired

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: Foundry property

{button ,AL(==H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS;H\_DIVISION\_CLASSES;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIVISION\_WPAPPLICATION\_FOUNDRY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The Foundry object which you use for accessing objects in a Division. In Word Pro, there is always one Foundry object for each Division object. The Foundry property on the WPAplication object (always named WordPro) always contains the Foundry object for the currently active division.

### Data Type

Foundry

### Syntax

foundryvalue = [objectreference].Foundry

### Legal values

### Usage

#### Division.Foundry

The Division Foundry provides access to all the objects in that division, including Layouts, Text, Graphics, Markers, Tables, Footnotes, and so on. You can access all the objects in a division through the appropriate collection in the Division Foundry.

#### WordPro.Foundry

The Foundry property on WordPro provides a shortcut to the currently active division's Foundry object. The Foundry object in WordPro.Foundry changes as the focus changes from one Division object to another. For example, if you had a document with one division named "Overview" and another division named "Summary," the contents of the Foundry property on WordPro would change as you moved the focus from Overview to Summary. While the focus is on the Overview division, this property contains the Foundry object for the Division object named Overview. When the focus is changed to the Summary division, the contents of this property changes to the Foundry object for the Division object named Summary.

## **Word Pro: DocControlRestrictedToEditor property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCCONTROLRESTRICTEDTOEDITOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Restricts a specific editor from setting or modifying document control options for a document.

### **Data Type**

[String](#)

### **Syntax**

doccontrolrestrictedtoeditorvalue = [objectreference].DocControlRestrictedToEditor

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing File - TeamSecurity, selecting "Only" and then the editor's name in the "Only" box on the Access panel.

## **Word Pro: DocPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the default path (drive and directory) where Word Pro looks for documents to open.

### **Data Type**

String

### **Syntax**

docpathvalue = [objectreference].DocPath

[objectreference].DocPath = docpathvalue

### **Legal values**

A valid path including drive and deirectory.

### **Usage**

Equivalent to the "Documents" field on the Locations panel of the Word Pro Preferences dialog box. In the Word Pro interface, the "Documents" field can contain multiple paths. You can use this property to clear all paths before setting the default or first document path, or you can use the property, DocumentPaths, to read multiple paths entered by the user.



**Word Pro: DocSize property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read only) Returns the size of a document in bytes.

**Data Type**

Long

**Syntax**

docsizevalue = [objectreference].DocSize

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: DocTextSize property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCTEXTSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

Not implemented.

### **Data Type**

Long

### **Syntax**

doctextsizevalue = [objectreference].DocTextSize

[objectreference].DocTextSize = doctextsizevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: DocumentLevel property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

documentlevelvalue = [objectreference].DocumentLevel

[objectreference].DocumentLevel = documentlevelvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: DocVersionID property**

{button ,AL('H\_VERSION\_CLASS;H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCVERSIONID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

docversionidvalue = [objectreference].DocVersionID

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: DoInitialCaps property**

{button ,AL('H\_SMARTCORRECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOINITIALCAPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

doinitialcapsvalue = [objectreference].DoInitialCaps

[objectreference].DoInitialCaps = doinitialcapsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: DoSmartQuotes property**

{button ,AL('H\_SMARTCORRECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOSMARTQUOTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

dosmartquotesvalue = [objectreference].DoSmartQuotes

[objectreference].DoSmartQuotes = dosmartquotesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: DoubleUnderline property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOUBLEUNDERLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

doubleunderlinevalue = [objectreference].DoubleUnderline

[objectreference].DoubleUnderline = doubleunderlinevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: DragDropOn property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DRAGDROPON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

dragdroponvalue = [objectreference].DragDropOn

[objectreference].DragDropOn = dragdroponvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



**Word Pro: DriverNameDriverName property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DRIVERNAMEDRIVERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

drivernamedrivernamevalue = [objectreference].DriverName

[objectreference].DriverName = drivernamedrivernamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: DuplexType property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DUPLExTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prints information on the front and back of the same page.

### **Data Type**

Variant (Enumerated)

DuplexType

### **Syntax**

duplextypevalue = [objectreference].DuplexType

[objectreference].DuplexType = duplextypevalue

### **Legal values**

\$LwpDuplexTypePrintHorzDuplex (219) Prints information across the width of the front and back of a printed page.

\$LwpDuplexTypePrintNoDuplex (217) Prevents information from being printed on the front and back of a page.

\$LwpDuplexTypePrintVertDuplex (218) ) Prints information across the length of the front and back of a printed page.

### **Usage**

## **Word Pro: Editable property**

{button ,AL('H\_BULLET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EDITABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

CommandState

### **Syntax**

editablevalue = [objectreference].Editable

[objectreference].Editable = editablevalue

### **Legal values**

\$LwpCommandStateOff (151)

\$LwpCommandStateOn (152)

\$LwpCommandStateStyle (153)

### **Usage**

**Word Pro: EditorAttemptingOpen property**

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EDITORATTEMPTINGOPEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores the name of the user attempting to open a document.

**Data Type**

[String](#)

**Syntax**

editorattemptingopenvalue = [objectreference].EditorAttemptingOpen

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Word Pro stores the name of every editor who attempts to open a document, regardless of whether or not the attempt was successful. Use this property to track the names of editors attempting to open a secured document.

## **Word Pro: EditorInitials property**

{button ,AL('H\_EDITOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EDITORINITIALS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the initials of the editor assigned to a document.

### **Data Type**

[String](#)

### **Syntax**

editorinitialsvalue = [objectreference].EditorInitials

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: EditorName property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_VERSION_CLASS',0)} See list of classes
```

```
{button ,AL('H_EDITORNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns the user name of the person who created a particular layout object.

## Data Type

[String](#)

## Syntax

editornamevalue = [objectreference].EditorName

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

This property stores the name of the person who created a layout object. In Word Pro, the editor name value can be accessed in the Personal panel of the Word Pro Preferences dialog box.

## Word Pro: EditorVerificationType property

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EDITORVERIFICATIONTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Verifies how an editor gains access to a document by using either E-mail login, operating system login, or user name.

### Data Type

Variant (Enumerated)

EditorVerifyType

### Syntax

editorverificationtypevalue = [objectreference].EditorVerificationType

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpEditorVerifyTypeByEmailLogin (236)	An editor must use an E-mail login to access a document.
\$LwpEditorVerifyTypeByOpsysLogin (235)	An editor must use an operating system login to access a document.
\$LwpEditorVerifyTypeByUserName (234)	An editor must use a user name to access a document.

### Usage

Equivalent to choosing File - TeamSecurity and selecting a verification type (E-mail login, operating system login, Word Pro user name) in the "Verify editors using" box on the Access panel.

### **Word Pro: EffectiveColumnWidth property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_EFFECTIVECOLUMNWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The column width less any indention for the paragraph in which the insertion point is located.

### **Data Type**

Long

### **Syntax**

effectivecolumnwidthvalue = [objectreference].EffectiveColumnWidth

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**



## **Word Pro: EMail property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EMAIL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

emailvalue = [objectreference].EMail

[objectreference].EMail = emailvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: Embedded property**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EMBEDDED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

embeddedvalue = [objectreference].Embedded

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: EmbedFonts property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EMBEDFONTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

embedfontsvvalue = [objectreference].EmbedFonts

[objectreference].EmbedFonts = embedfontsvvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: Enabled property**

{button ,AL(^H\_LWPTIMER\_CLASS;H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[MenuItem]

Indicates if a menu item is active (grayed) or not active (not grayed).

## **Data Type**

Integer

## **Syntax**

enabledvalue = [objectreference].Enabled

[objectreference].Enabled = enabledvalue

## **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## **Usage**

To gray a menu item, set the Checked property value to True. To ungray a menu item, set the Checked property value to False.

Setting this property is available only for custom created menu items. You cannot set the Enabled property for predefined Word Pro menu items, because Word Pro dynamically sets the Enabled property for predefined Word Pro menu items, based on the current context.

## **Word Pro: EncryptPassword2 property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENCRYPTPASSWORD2\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

encryptpassword2value = [objectreference].EncryptPassword2

[objectreference].EncryptPassword2 = encryptpassword2value

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: EncryptPassword property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENCRYPTPASSWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

encryptpasswordvalue = [objectreference].EncryptPassword

[objectreference].EncryptPassword = encryptpasswordvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: EndingColOfSelection property**

{button ,AL(^H\_BASetable\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ENDINGCOLOFSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of the last column included in a selection of table cells.

**Data Type**

Integer

**Syntax**

endingcolofselectionvalue = [objectreference].EndingColOfSelection

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The column ID is a zero based value, which means that the first column in a table has a row ID value of zero.

## **Word Pro: EndingRowOfSelection property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ENDINGROWOFSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of the last row included in a selection of table cells.

### **Data Type**

[Integer](#)

### **Syntax**

endingrowofselectionvalue = [objectreference].EndingRowOfSelection

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

The row ID is a zero based value, which means that the first row in a table has a row ID value of zero.



## Word Pro: Enumeration property

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENUMERATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Tells Word Pro which objects to include in the specified Foundry's collection objects.

### Data Type

Variant (Enumerated)

EnumScope

### Syntax

enumerationvalue = [objectreference].Enumeration

[objectreference].Enumeration = enumerationvalue

### Legal values

\$LwpEnumScopeAll (237) Include both types of objects in this Foundry's collection objects.

\$LwpEnumScopeLocal (238) Include only those objects which are not part of a SmartMaster.

\$LwpEnumScopeStyle (239) Include on those objects which are part of a SmartMaster.

Do not use quotation marks around the string values above. You can also use the numeric equivalents for these values (shown in parentheses).

### Usage

Each Word Pro document is first created from a SmartMaster. All the objects that comprise the SmartMaster are known as "style" objects. These objects might include ParagraphStyle, CharacterStyle, PageLayout, TableLayout, FrameLayout, and so on. When you add new objects to your document, such as a new paragraph of text, a new paragraph style or a new page layout that were not part of the original SmartMaster, Word Pro sees that object as a "local" object. Both "style" objects and "local" objects can be instantiated from the same class, but Word Pro sees them as either a part of the SmartMaster or not part of the SmartMaster.

For example, when you create a document whose SmartMaster contains a division with a paragraph style named "Body Text," you could create a new paragraph style called "Title." When you looked into the ParagraphStyleCollection object for that division, you would see two ParagraphStyle objects. Word Pro differentiates automatically between the "style" objects and the "local" objects. However, it treats them the same within the context of LotusScript.

You may find it useful to exclude the objects which originated with the SmartMaster from your collections. Setting the value of the Enumeration property to "\$LwpEnumScopeLocal" hides the SmartMaster objects from the Script Editor. By default, Word Pro stores both the "style" objects and the "local" objects in their appropriate collection objects.

## **Word Pro: Epoch property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EPOCH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

epochvalue = [objectreference].Epoch

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: EqnFontHeight property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EQNFONTHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Long](#)

**Syntax**

eqnfontheightvalue = [objectreference].EqnFontHeight

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: ExcludeRectBottom property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXCLUDERECTBOTTOM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

excluderectbottomvalue = [objectreference].ExcludeRectBottom

[objectreference].ExcludeRectBottom = excluderectbottomvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: ExcludeRectLeft property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXCLUDERECTLEFT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Long](#)

### **Syntax**

excluderectleftvalue = [objectreference].ExcludeRectLeft

[objectreference].ExcludeRectLeft = excluderectleftvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

**Word Pro: ExcludeRectRight property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXCLUDERECTRIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

excluderectrightvalue = [objectreference].ExcludeRectRight

[objectreference].ExcludeRectRight = excluderectrightvalue

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

## **Word Pro: ExcludeRectTop property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXCLUDERECTTOP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

excluderecttopvalue = [objectreference].ExcludeRectTop

[objectreference].ExcludeRectTop = excluderecttopvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: Expandable property**

{button ,AL('H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPANDABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

expandablevalue = [objectreference].Expandable

[objectreference].Expandable = expandablevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



**Word Pro: ExportToNotesFX property**

{button ,AL('H\_DOCINFOFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPORTTONOTESFX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not a DocInfo field should be exported to Notes.

**Data Type**

[Integer](#)

**Syntax**

exporttonotesFXvalue = [objectreference].ExportToNotesFX

[objectreference].ExportToNotesFX = exporttonotesFXvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

For example, if you have a DocInfo field named "Client" and you want to export that field to Notes, you can set the ExportToNotesFX property to True. Notes would then export the "Client" field with its contents from the document to Notes.

## **Word Pro: ExternalFileID property**

{button ,AL('H\_DIVISIONINFO\_CLASS;H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXTERNALFILEID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

externalfileidvalue = [objectreference].ExternalFileID

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: ExternalFileName property**

{button ,AL('H\_DIVISIONINFO\_CLASS;H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXTERNALFILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

String

**Syntax**

externalfilenamevalue = [objectreference].ExternalFileName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: Created event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATED\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

#### **Parameters**

Created(String StyleSheet)

### **Usage**

## **Word Pro: DocumentClosed event**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTCLOSED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised by Word Pro each time Word Pro closes a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro closed.

### **Usage**

You can write a script for this event which checks the name of the document and performs a set of actions after Word Pro has closed the document. This event is useful for cleaning up the application workspace.

## **Word Pro: DocumentClose event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTCLOSE\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised by Word Pro each time Word Pro receives an instruction to close an open document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to close.

### **Usage**

You can write a script for this event that checks the name of the document and performs a set of actions before allowing Word Pro to close the document. You can also use the object in the Source argument to access the WPAApplication object. This event is also useful for error-checking, prompting for saves, and cleaning up the application workspace.

## **Word Pro: DocumentCreated event**

{button ,AL(^H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTCREATED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro completes the creation of a new Word Pro document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *StyleSheet*

A String representing the name of the SmartMaster used in creating the new document.

### **Usage**

Use this event to detect the completion of the creation of a new document in Word Pro. You can write a script for this event which checks the name of the SmartMaster and performs a set of actions after Word Pro has created the document. This event is also useful for error-checking, prompting for optional actions, and preparing the application workspace.

## **Word Pro: DocumentCreate event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTCREATE\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised by Word Pro each time Word Pro receives an instruction to create a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *StyleSheet*

A String expression which specifies the name of the SmartMaster which Word Pro has been instructed to use in creating the new document.

### **Usage**

You can write a script for this event which checks the name of the SmartMaster and performs a set of actions before allowing Word Pro to create the document. This event is also useful for error-checking, prompting for optional actions, and preparing the application workspace.



## **Word Pro: DocumentExported event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTEXPORTED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro exports a document. Equivalent to choosing File - Import/Export and exporting a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro exported.

#### *DocType*

A String expression which specifies the file type of the exported document.

### **Usage**

## **Word Pro: DocumentExport event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTEXPORT\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to export a document. Equivalent to choosing File - Import/Export and exporting a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to export.

#### *DocType*

A String expression which specifies the file type which Word Pro will use for the exported document.

### **Usage**

## **Word Pro: DocumentImported event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTIMPORTED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro imports a document. Equivalent to choosing File - Import/Export and importing a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro imported.

#### *DocType*

A String expression which specifies the file type of the imported document.

### **Usage**

## **Word Pro: DocumentImport event**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DOCUMENTIMPORT\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to import a document. Equivalent to choosing File - Import/Export and importing a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to import.

#### *DocType*

A String expression which specifies the file type of the document which Word Pro is supposed to import.

### **Usage**

## **Word Pro: DocumentInserted event**

{button ,AL(^H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTINSERTED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro inserts a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has inserted.

### **Usage**

## **Word Pro: DocumentInsert event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTINSERT\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to insert a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to insert.

### **Usage**

## **Word Pro: DocumentOpened event**

{button ,AL(^H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTOPENED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro opens a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro opened.

### **Usage**

Use this event to detect when Word Pro opens a document. You can use the DocName argument to check the name of the document Word Pro opened.

## **Word Pro: DocumentOpen event**

{button ,AL(^H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTOPEN\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to open a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to open.

### **Usage**

Use this event to detect when Word Pro receives a command to open a document.



## **Word Pro: DocumentPrinted event**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTPRINTED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro prints a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro printed.

### **Usage**

## **Word Pro: DocumentPrint event**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DOCUMENTPRINT\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to print a document.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to print.

### **Usage**

## **Word Pro: DocumentSaveAs event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTSAVEAS\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to save a document as another type of document or under another name. Equivalent to choosing File - Save As.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to save as another type of document or under another name.

### **Usage**

## **Word Pro: DocumentSavedAs event**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTSAVEDAS\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro saves a document as another type of document or under another name. Equivalent to choosing File - Save As.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has saved in another file type or under another name.

### **Usage**

## **Word Pro: DocumentSaved event**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOCUMENTSAVED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro saves a document. Equivalent to choosing File - Save.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro saved.

### **Usage**

## **Word Pro: DocumentSave event**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DOCUMENTSAVE\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to save a document. Equivalent to choosing File - Save.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document which Word Pro has been instructed to save.

### **Usage**

## **Word Pro: EnterClickHere event**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ENTERCLICKHERE\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time the insertion point is moved from somewhere in the document into a ClickHere block.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *ClickHereName*

A String expression which specifies the name of the ClickHere block that was entered.

### **Usage**

## Word Pro: EnterLayout event

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_ENTERLAYOUT_EVENT_EXSCRIPT',1)} See example
```

Raised each time the insertion point is moved into a layout.

## Arguments

### *Source*

The object which raised this event. This could be WPAApplication or the layout itself. Use this argument to determine which object raised this event.

### *LayoutEntered*

A String expression which specifies the name of the layout which was entered.

## Usage

Each part of a document has its own layout. For example, a page, header, footer, and frame each has its own layout, even though they are all on the same page. When you move the insertion point from one of these layouts to another by clicking in the header, the footer, or some other part of the document, Word Pro raises an EnterLayout event. You can check the Source argument to determine which object is responsible for raising this event.



## **Word Pro: ExitClickHere event**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_EXITCLICKHERE\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time the insertion point is moved out of a ClickHere block.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *ClickHereName*

A String expression which specifies the name of the exited ClickHere block.

### **Usage**

## **Word Pro: ImportInserted event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IMPORTINSERTED\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro imports a document and inserts the contents at the current insertion point. Equivalent to choosing File - Import/Export and specifying the option, "Import at the current insertion point."

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document that Word Pro inserted.

#### *DocType*

A String expression which specifies the file type of the document that Word Pro inserted.

### **Usage**

## **Word Pro: ImportInsert event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IMPORTINSERT\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to import a document and insert the contents at the current insertion point. Equivalent to choosing File - Import/Export and specifying the option, "Import at the current insertion point."

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *DocName*

A String expression which specifies the name of the document that Word Pro has been instructed to import.

#### *DocType*

A String expression which specifies the file type of the document that Word Pro is supposed to import.

### **Usage**

## Word Pro: KeyStroke event

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_KEYSTROKE_EVENT_EXSCRIPT',1)} See example
```

Raised each time a user presses and releases a key.

## Arguments

### *Source*

The object which raised this event. Use this argument to determine which object raised this event. You can use Source to retrieve information about or change the properties of the object in which the keystroke occurred. When you write a script for a keystroke event in a Layout object, this argument returns the Layout object in which the keystroke event was raised. When you write a script for a keystroke event in the WPApplication object, this argument returns the WPApplication object.

### *Key*

An Integer which specifies the ASCII value of the key that was pressed. This code can be translated into a character, using the Chr function.

### *Modifier*

An Integer which specifies the ASCII value of the modifier key that was pressed.

### *ReceivingLayout*

A String expression representing the name of the layout object in which the Keystroke event was raised.

## Usage

Word Pro can detect a Keystroke event in four different types of objects: WPApplication, TextDocument, Division, and Layout. This means that you can write a script that will run every time a keystroke is detected anywhere within Word Pro (WPApplication), anywhere within a particular document (TextDocument), anywhere within a particular division (Division), or anywhere within a particular layout (Layout). By placing your script in the Keystroke event section for a particular object, you allow the script to run only when a Keystroke event is detected within that object. You can also use the End statement to limit the number of events that Word Pro will raise for a single keystroke.

## Word Pro: MouseDown event

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL(^H_MOUSESDOWN_EVENT_EXSCRIPT',1)} See example
```

Raised each time the user presses down on a mouse button.

## Arguments

### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

### *GeneralModifier*

The key or keys which were pressed when the mouse button was pressed down. Data type is Variant which allows the value of the argument to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

LwpGeneralModifierNone (&H0) No keys were down.

LwpGeneralModifierShift (&H1) The SHIFT key was down.

LwpGeneralModifierAlt (&H2) The ALT key was down.

LwpGeneralModifierCtrl (&H4) The CTRL key was down.

LwpGeneralModifierCommand (&H8) The COMMAND key was down.

LwpGeneralModifierOption (&H10) The OPTION key was down.

LwpGeneralModifierSys (&H20) The SYSRQ key was down.

LwpGeneralModifierCapslock (&H40) The CAPS LOCK key was down.

LwpGeneralModifierNumlock (&H80) The NUM LOCK key was down.

LwpGeneralModifierScrolllock (&H100) The SCROLL LOCK key was down.

LwpGeneralModifierHelp (&H200) The HELP key was down.

### *SpecificModifier*

Specifies which mouse button was pressed. Data type is Variant which allows the value of the argument to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

LwpSpecificModifierNone (&H0)

LwpSpecificModifierMbutton1 (&H1)

LwpSpecificModifierMbutton2 (&H2)

LwpSpecificModifierMbutton3 (&H4)

LwpSpecificModifierMbuttonmask (&H7)

LwpSpecificModifierMbuttondown (&H8)

LwpSpecificModifierMbuttonup (&H10)

LwpSpecificModifierDoublebyte (&H1)

LwpSpecificModifierGoingdown (&H2)

LwpSpecificModifierComingup (&H4)

LwpSpecificModifierVirtual (&H8)

LwpSpecificModifierClicked (&H2)

LwpSpecificModifierNomove (&H4)

LwpSpecificModifierNovolatile (&H8)

LwpSpecificModifierReset (&H10)

LwpPopcheckpoint (&H20)

*ReceivingLayout*

A String expression which specifies the name of the layout over which the mouse button was pressed.

**Usage**

## Word Pro: MouseUp event

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL(^H_MOUSEUP_EVENT_EXSCRIPT',1)} See example
```

Raised each time the user releases a mouse button after pressing it.

## Arguments

### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

### *GeneralModifier*

The key or keys which were pressed when the mouse button was released. Data type is Variant which allows the value of the argument to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

LwpGeneralModifierNone (&H0) No keys were down.

LwpGeneralModifierShift (&H1) The SHIFT key was down.

LwpGeneralModifierAlt (&H2) The ALT key was down.

LwpGeneralModifierCtrl (&H4) The CTRL key was down.

LwpGeneralModifierCommand (&H8) The COMMAND key was down.

LwpGeneralModifierOption (&H10) The OPTION key was down.

LwpGeneralModifierSys (&H20) The SYSRQ key was down.

LwpGeneralModifierCapslock (&H40) The CAPS LOCK key was down.

LwpGeneralModifierNumlock (&H80) The NUM LOCK key was down.

LwpGeneralModifierScrolllock (&H100) The SCROLL LOCK key was down.

LwpGeneralModifierHelp (&H200) The HELP key was down.

### *SpecificModifier*

Specifies which mouse button was released. Data type is Variant which allows the value of the argument to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these values when you want Word Pro to combine the features listed below. Use the OR operator to combine values.

LwpSpecificModifierNone (&H0)

LwpSpecificModifierMbutton1 (&H1)

LwpSpecificModifierMbutton2 (&H2)

LwpSpecificModifierMbutton3 (&H4)

LwpSpecificModifierMbuttonmask (&H7)

LwpSpecificModifierMbuttondown (&H8)

LwpSpecificModifierMbuttonup (&H10)

LwpSpecificModifierDoublebyte (&H1)

LwpSpecificModifierGoingdown (&H2)

LwpSpecificModifierComingup (&H4)

LwpSpecificModifierVirtual (&H8)

LwpSpecificModifierClicked (&H2)

LwpSpecificModifierNomove (&H4)

LwpSpecificModifierNovolatile (&H8)

LwpSpecificModifierReset (&H10)

LwpPopcheckpoint (&H20)

*ReceivingLayout*

A String expression which specifies the name of the layout over which the mouse button was released.

**Usage**



### **Word Pro: Moved event**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVED\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

Moved()

### **Usage**

This event has not been implemented for Word Pro '97.

## **Word Pro: Opened event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENED\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

#### **Parameters**

Opened(String DocName)

### **Usage**

## **Word Pro: PreClose event**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRECLOSE\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

PreClose(String DocName)

### **Usage**

## **Word Pro: PrePrint event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREPRINT\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

PrePrint(String DocName)

### **Usage**

## **Word Pro: Printed event**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTED\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

Printed(String DocName)

### **Usage**

## **Word Pro: Quit event**

{button ,AL(^H\_APPLICATION\_CLASS;H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_QUIT\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time Word Pro receives an instruction to exit the Word Pro application. The event handler in the Script Editor processes this event and any related scripts before Word Pro closes.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

### **Usage**

Use this event to run a script any time you exit Word Pro. The document that contains the Quit event script must be open when you exit Word Pro.

## **Word Pro: SaveAs event**

{button ,AL(^H\_DOCUMENT\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SAVEAS\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

SaveAs(String DocName)

### **Usage**

## **Word Pro: SavedAs event**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEDAS\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

SavedAs(String DocName)

### **Usage**



## **Word Pro: Saved event**

{button ,AL(^H\_DOCUMENT\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SAVED\_EVENT\_EXSCRIPT',1)} [See example](#)

## **Arguments**

### **Parameters**

Saved(String DocName)

*DocName*

A String representing the name of the saved document.

## **Usage**

## **Word Pro: Save event**

{button ,AL(^H\_DOCUMENT\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SAVE\_EVENT\_EXSCRIPT',1)} [See example](#)

### **Arguments**

### **Parameters**

Save(String DocName)

### **Usage**

## **Word Pro: StatusBarButtonClicked event**

{button ,AL('H\_STATUSBAR\_CLASS;H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBARBUTTONCLICKED\_EVENT\_EXSCRIPT',1)} [See example](#)

This event is emitted when a button is clicked on or when following the SimulateButtonClick method..

If you choose the status bar object in the LotusScript window, you will see the emitted events assigned to it. If you choose one event and write a routine within it, and, if any value other than False (0) is returned, that event will be handled by LotusScript, not Word Pro.

### **Arguments**

ButtonName = name of button that is clicked.

### **Parameters**

StatusBarButtonClicked(String ButtonName)

### **Usage**

This event applies only to buttons of type clickable. Popup type buttons receive the StatusBarButtonFillPopupList event instead.

In the script for this event, you should write the code that you want executed in response to the button click. You can determine which button was clicked by looking at the ButtonName parameter.

## **Word Pro: StatusBarButtonFillPopupList event**

{button ,AL('H\_STATUSBAR\_CLASS;H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBARBUTTONFILLPOPUPLIST\_EVENT\_EXSCRIPT',1)} [See example](#)

This event is emitted when a button of type popup is clicked. You can write instructions in this event to fill up the popup list.

If you choose the status bar object in the LotusScript window, you will see the emitted events assigned to it. If you choose one event and write a routine within it, and, if any value other than False (0) is returned, that event will be handled by LotusScript, not Word Pro.

### **Arguments**

### **Parameters**

StatusBarButtonFillPopupList(String ButtonName)

### **Usage**

The list is always empty when this event is emitted. There are several methods that you can call when responding to this event:

AddPopupGraphicItem - only used for graphic buttons; can be called multiple times, once for each graphic in the list.

AddPopupTextItem - only used for text popup buttons; can be called multiple times, once for each string in the list.

SetPopupAlignment - lets you align the button's contents.

SetPopupIndex - lets you select which item in the button's list will be highlighted by default.

SetPopupWidth - lets you set a width for the popup list.

SetPopupWidthType - lets you specify how the width of the popup list will be calculated.

If you want Word Pro to do its normal processing for a button property, this event should return False. To prevent Word Pro from doing its normal processing, this event should return True.

## **Word Pro: StatusBarButtonItemSelected event**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBARBUTTONITEMSELECTED\_EVENT\_EXSCRIPT',1)} [See example](#)

This event is emitted when an item is selected from a popup list in the button.

If you choose the status bar object in the LotusScript window, you will see the emitted events assigned to it. If you choose one event and write a routine within it, and, if any value other than False (0) is returned, that event will be handled by LotusScript, not Word Pro.

### **Arguments**

Button name - Name of the button selected.

Index (0 based) - Indicates the item number selected.

String - The text of the selected item.

### **Parameters**

StatusBarButtonItemSelected(String ButtonName, Integer Index, String SelectedItem)

### **Usage**

This event applies only to buttons of type clickable. Popup type buttons receive the StatusBarButtonFillPopupList event instead.

In the script for this event, you should write the code that you want executed in response to the button click. You can determine which button was clicked by looking at the ButtonName parameter.

## **Word Pro: StatusBarButtonOverrideGraphic event**

{button ,AL(`H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_STATUSBARBUTTONOVERRIDEGRAPHIC\_EVENT\_EXSCRIPT',1)} [See example](#)

This event is emitted when a graphic button must be refreshed.

If you choose the status bar object in the LotusScript window, you will see the emitted events assigned to it. If you choose one event and write a routine within it, and, if any value other than False (0) is returned, that event will be handled by LotusScript, not Word Pro.

### **Arguments**

### **Parameters**

StatusBarButtonOverrideGraphic(String ButtonName)

### **Usage**

You are responsible for providing the graphic in this event. Otherwise, the button will appear blank. The SetOverrideGraphic method must be called.

## **Word Pro: StatusBarButtonOverrideTextAndGraphic event**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBARBUTTONOVERRIDEDEXTANDGRAPHIC\_EVENT\_EXSCRIPT',1)} [See example](#)

This event is emitted when a text and graphic button must be updated.

If you choose the status bar object in the LotusScript window, you will see the emitted events assigned to it. If you choose one event and write a routine within it, and, if any value other than False (0) is returned, that event will be handled by LotusScript, not Word Pro.

### **Arguments**

### **Parameters**

StatusBarButtonOverrideTextAndGraphic(String ButtonName)

### **Usage**

You are responsible for providing the graphic and the text in this event. Otherwise, the button will appear blank. The SetOverrideGraphic and/or the SetOverrideText method must be called.

**Note** If the text on the status bar button is never going to change, you can use the LwpButtonNoTextFromHost (&H800) parameter when the button is created.

## **Word Pro: StatusBarButtonOverrideText event**

{button ,AL(^H\_STATUSBAR\_CLASS;H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_STATUSBARBUTTONOVERRIDE TEXT\_EVENT\_EXSCRIPT',1)} [See example](#)

This event is emitted when a text button must be updated.

If you choose the status bar object in the LotusScript window, you will see the emitted events assigned to it. If you choose one event and write a routine within it, and, if any value other than False (0) is returned, that event will be handled by LotusScript, not Word Pro.

### **Arguments**

### **Parameters**

StatusBarButtonOverrideText(StatusBarButton Source, String ButtonName)

### **Usage**

You are responsible for providing the text in this event. Otherwise, the button will appear blank. The SetOverrideText method must be called.

**Note** If the text on the status bar button is never going to change, you can use the LwpButtonNoTextFromHost (&H800) parameter when the button is created.



**Word Pro: TimerTick event**

{button ,AL('H\_LWPTIMER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TIMERTICK\_EVENT\_EXSCRIPT',1)} [See example](#)

**Arguments****Parameters**

TimerTick()

**Usage**

## **Word Pro: WMCommand event**

{button ,AL(^H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_WMCOMMAND\_EVENT\_EXSCRIPT',1)} [See example](#)

Raised each time a WMCommand is issued in Word Pro. This happens when a user chooses a menu or menu item and when a script calls the WMCommand method.

### **Arguments**

#### *Source*

The object which raised this event. Use this argument to determine which object raised this event.

#### *cmd*

An Integer expression which specifies which menu item or WMCommand was chosen or called. You can find the WMCommands and their corresponding integers in the file named WPBITMSK.LSS, which came with Word Pro.

### **Usage**

## Word Pro: CreateOleEmbeddedFile method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEOLEEMBEDDEDFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates an embedded OLE object from an existing file. Equivalent to choosing Create - Object, selecting "Create an object from a file" and specifying a file to embed.

### Syntax

[objectreference].CreateOleEmbeddedFile(sClassID,FilePath,IconMetaFilePictHandle)

### Parameters

#### *sClassID*

A String expression which specifies, in the form of a ClassID or a ProgID, the server application for the file you are embedding. For the purposes of this method, you can let Word Pro determine the proper file type by using the following value:

"{00000000-0000-0000-0000-000000000000}"

#### *FilePath*

A String expression indicating the path and name of the file you are embedding.

#### *IconMetaFilePictHandle*

An optional Numeric expression that allows you to specify which icon to use in representing the embedded object in the Word Pro file. Using any value other than 0 automatically tells Word Pro to display the embedded object as an icon. This value serves as a numeric handle (known as the HGLOBAL) to the metafile pict for an icon. You can get the HGLOBAL for a specific metafile pict by using the appropriate Windows API calls. This value must be of type Long. Default is 0, which indicates that you want the contents of the embedded OLE object to be displayed in the Word Pro document.

**Caution** If you record the process of choosing Create - Object and embedding a file to be displayed as an icon, Word Pro records a value for IconMetaFilePict which is valid only during the recording. When you play back the recorded script, the IconMetaFilePict value will be invalid and Word Pro will treat the value as if you passed a 0.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: CreateOleLinkedFile method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEOLELINKEDFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a linked OLE object from an external file. Equivalent to choosing Create - Object, selecting "Create an object from a file", selecting "Link to file", and specifying a file to link.

### Syntax

[objectreference].CreateOleLinkedFile(FilePath,[IconMetaFilePictHandle])

### Parameters

#### *FilePath*

A String expression which specifies the file path and name of the file you are linking.

#### *IconMetaFilePictHandle*

An optional Numeric expression which allows you to specify which icon to use in representing the linked object in the Word Pro file. Using any value other than 0 automatically tells Word Pro to display the linked object as an icon. This value serves as a numeric handle (known as the HGLOBAL) to the metafile pict for that icon. You can get the HGLOBAL for a specific metafile pict by using the appropriate Windows API calls. This value must be of type Long. Default is 0 which indicates that you want the contents of the linked OLE object to be displayed in the Word Pro document.

**Caution** If you record the process of choosing Create - Object and linking a file to be displayed as icon, Word Pro records a value for IconMetaFilePict which is valid only during the recording. When you play back the recorded script, the IconMetaFilePict value will be invalid and Word Pro will treat the value as if you passed a 0.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: CreateOleNew method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEOLENEW\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new OLE object and launches the server application. Equivalent to choosing Create - Object, selecting "Create a new object" and specifying type of object to create.

### Syntax

[objectreference].CreateOleNew(sClassID, IconMetaFilePictHandle)

### Parameters

*sClassID*

A String expression which specifies the type of object you are creating. The type of object is expressed as the ClassID or ProgID for the application which creates that type of object. (The application used to create an OLE object is often referred to as the server application.) For example, a Lotus Freelance 96 Presentation has a ClassID of "{CF746000-94FB-101B-8C12-02608C454BFF}" and a ProgID of "FLW3Presentation."

Here are the server application IDs for some other SmartSuite application objects:

#### 1-2-3 Worksheet

ClassID = {00045295-0000-0000-C000-000000000046}

ProgID = 123Worksheet

Launches 1-2-3 and opens an untitled worksheet.

#### Lotus Approach 96 Report

ClassID = {00028703-0000-0000-C000-000000000046}

ProgID = ApproachReport

Launches Approach 96 and prompts the user to select an existing database from which to create the report. Once the database is open, the Report Assistant opens and waits for the user to create the report.

#### Lotus Approach 96 Application

ClassID = {00028701-0000-0000-C000-000000000046}

ProgID = ApproachApplication

Launches Approach 96 and prompts the user to select an existing database.

#### Lotus Freelance 96 Presentation

ClassID = {CF746000-94FB-101B-8C12-02608C454BFF}

ProgID = FLW3Presentation

Launches Freelance 96 and prompts the user with the New Presentation dialog box.

#### Lotus Freelance 96 Drawing

ClassID = {CF746001-94FB-101B-8C12-02608C454BFF}

ProgID = FLW3Drawing

Launches Freelance 96 and opens a new presentation with one blank page.

#### Lotus ScreenCam Movie 2.1

ClassID = {00041920-0000-0000-C000-000000000046}

ProgID = ScreenCamMovie2

Launches ScreenCam 2.1 and displays the ScreenCam control panel for the user to start a recording.

You can find the ClassIDs and ProgIDs for other server applications in the Windows Registry for Windows 3.1 and Windows 95.

*IconMetaFilePictHandle*

An optional Numeric expression which allows you to specify which icon to use in representing the new OLE object. Using any value other than 0 automatically tells Word Pro to display the new object as an icon. This value serves as a numeric handle (known as the HGLOBAL) to the metafile pict for that icon. You can get the HGLOBAL for a specific metafile pict by using the appropriate Windows API calls. This value must be of type Long. Default is 0, which indicates that you want Word Pro to display the contents of the new OLE object.

**Caution** If you record the process of creating a new OLE object to be displayed as an icon, Word Pro records a value for IconMetaFilePict which is valid only during the recording. When you play back the recorded script, the IconMetaFilePict value will be invalid and Word Pro will treat the value as if you passed a 0.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## Word Pro: CreateParallelColumns method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEPARALLELCOLUMNS\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates parallel columns at the insertion point in the currently active document. Equivalent to choosing Create - Parallel Columns.

### Syntax

[objectreference].CreateParallelColumns(NumCols, [AlignmentType])

### Parameters

#### *NumCols*

An Integer value which specifies the number of parallel columns you want to create.

#### *AlignmentType*

Allows you to specify what type of alignment you want to use for the data in your parallel columns. Data type is Variant which allows the value of this parameter to be one of the nine string values listed below or their numeric equivalents (in parentheses). Default is `$LtsAlignmentHorizCenter`.

`$LtsAlignmentHorizCenter (1056964611)` Centers all data horizontally.

`$LtsAlignmentJustify (1056964613)` Justifies the alignment of all data with the left and right boundaries of the columns.

`$LtsAlignmentLeft (1056964609)` Aligns all data with the left side of the column.

`$LtsAlignmentRight (1056964610)` Aligns all data with the right side of the column.

`$LtsAlignmentSmart (1056964612)` Aligns data according to the type of data. Numbers are right-aligned, text is left-aligned, and so on.

`$LwpAlignmentTypeAlignRevert (8)` Reverts the alignment of all data to the alignment of provided by the paragraph style.

`$LwpAlignmentTypeJustifyall (5)` Justifies the alignment of all data with the left and right boundaries of the columns.

`$LwpAlignmentTypeNumericleft (6)` Aligns all data to the left-most decimal (or period if text is present).

`$LwpAlignmentTypeNumericright (7)` Aligns all data to the right-most decimal (or period if text is present).

### Return value

None

### Usage

The data in parallel columns flows from the top to the bottom of the left column and then into the next column to the right. Once the data reaches the bottom of the right parallel column on a page, it then flows to the top of the left parallel column on the next page.

When created, you can access and manipulate the currently active `ParallelColumns` object through the `ParallelColumns` property on `WPApplication`.

## **Word Pro: CreateRemark method**

{button ,AL('H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEREMARK\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to insert an editorial remark in a version of a document. Equivalent to choosing File - Versions, selecting the desired version, and clicking New Remark.

### **Syntax**

[objectreference].CreateRemark[(Remarks, Time, EditorInitials)]

### **Parameters**

*Remarks*

Data type is String.

*Time*

Data type is Long.

*EditorInitials*

Data type is String.

### **Return value**

### **Usage**



## Word Pro: CreateTable method

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_CREATETABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a table at the insertion point in the currently active document. Equivalent to choosing Create - Table.

### Syntax

[objectreference].CreateTable([UseDefault,][FrameStyle,][Columns,][Rows]

### Parameters

#### *UseDefault*

Allows you to use or bypass the default table style. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True which tells Word Pro to use the default table style.

#### *FrameStyle*

A String expression which specifies the name of the table style you want Word Pro to use in creating the new table. Use this parameter when you set the UseDefault parameter to False.

#### *Columns*

An Integer value indicating the number of columns you want to have in the new table. Data type is Long. You must provide a value for this parameter if you set UseDefault to False.

#### *Rows*

An Integer value indicating the number of rows you want to have in the new table. Data type is Long. You must provide a value for this parameter if you set UseDefault to False.

### Return value

None

### Usage

Issued without parameters, this method creates a table based on the default table style.

## **Word Pro: CreateVersion method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATEVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to create a version of a document for editing. Equivalent to File - Versions.

### **Syntax**

[objectreference].CreateVersion(Name)

### **Parameters**

*Name*

Data type is String.

### **Return value**

### **Usage**

## Word Pro: Create method

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CREATE\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates the specified type of object in the Foundry object from which you call this method.

## Syntax

[objectreference].Create(*CreateType* [,P2] [,P3])

## Parameters

### *CreateType*

Indicates which type of object you want to create. The need for parameters P2 and P3 is determined by the value of the *CreateType* parameter. Most of the values listed below do not require you to include the P2 or P3 parameters.

Choose one of the string values below or its numeric equivalent (indicated in parentheses):

\$LwpFoundryCreateTypeStyle (2028)

Creates an object from the one of the following classes: CellLayout, CharacterStyle, FrameLayout, PageLayout, ParagraphStyle, TableLayout. When you use this value for CreateType, you must also include values for P2 and P3.

\$LwpFoundryCreateTypeLayout (2029)

Creates a Layout object from one of the following classes: FrameLayout, NoteLayout, PageLayout, RubyLayout. When you use this value for CreateType, you must also include a value for P2 but no value is needed for P3.

\$LwpFoundryCreateTypeText (2030)

Creates a Text object from the Text class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeSupertable (2031)

Creates a SuperTable object from the SuperTable class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeToc (2032)

Creates a TOCSuperTable object from the TOCSuperTable class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeParallelcols (2033)

Creates a ParallelColumns object from the ParallelColumns class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeGraphic (2034)

Creates a Graphic object from the Graphic class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeOle (2035)

Creates an OLEObject object from the OLEObject class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeFootnote (2036)

Creates a Footnote object from the Footnote class. When you use this value for CreateType, you need a value for P2 but you do not need a value for P3.

\$LwpFoundryCreateTypeField (2037)

Creates a DocInfoField object from the DocInfoField class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeSilverbullet (2038)

Creates a SilverBullet object from the SilverBullet class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeSection (2039)

Creates a Section object from the Section class. When you use this value for CreateType, you do not need values for P2 and P3.

\$LwpFoundryCreateTypeIndexsection (2040)

Creates an IndexSection object from the IndexSection class. When you use this value for CreateType, you do not

need values for P2 and P3.

**\$LwpFoundryCreateTypeBag (2041)**

Creates a Bag object from the Bag class. When you use this value for CreateType, you do not need values for P2 and P3.

**\$LwpFoundryCreateTypeOutlineseq (2042)**

Creates an OutSeqItem object from the OutSeqItem class. When you use this value for CreateType, you do not need values for P2 and P3.

**\$LwpFoundryCreateTypeClickhere (2043)**

Creates a ClickHere object from the ClickHere class. When you use this value for CreateType, you do not need values for P2 and P3.

## P2

Data type for P2 is Variant which allows P2 to accept values of any data type. There is no default value for P2. Use P2 only when the CreateType parameter has one of the following three values:

### **\$LwpFoundryCreateTypeStyle (2028)**

If CreateType has a value of "\$LwpFoundryCreateTypeStyle" (or 2028), then the P2 parameter value must be a String which specifies the name of the new style object you are creating. You must also include a value for P3.

### **\$LwpFoundryCreateTypeLayout (2029)**

If CreateType has a value of "\$LwpFoundryCreateTypeLayout" (or 2029), you must use one of the following values for P2:

<u>Value</u>	<u>Layout Object Created</u>
7	PageLayout
17	NoteLayout
662	FrameLayout
628	RubyLayout

### **\$LwpFoundryCreateTypeFootnote (2036)**

If CreateType has a value of "\$LwpFoundryCreateTypeFootnote" (or 2036), the value of the P2 parameter must be one of the following strings or its numeric equivalent (indicated in parentheses):

\$LwpFnTypeAnyposition (289)

\$LwpFnTypeAtBottomOfPage (290)

\$LwpFnTypeAtEndOfDoc (291)

\$LwpFnTypeAtEndOfDocSepDiv (292)

\$LwpFnTypeAtEndOfDiv (293)

\$LwpFnTypeAtEndOfDivisionSepDiv (294)

\$LwpFnTypeAtEndOfDivisionGroup (295)

\$LwpFnTypeAtEndOfDivGroupSepDiv (296)

## P3

An Integer expression which specifies which type of style object you want to create. Use this parameter only when you set the value of the CreateType parameter to "\$LwpFoundryCreateTypeStyle" (or 2028). You must use one of the integers below:

<u>Value</u>	<u>Style Object Created</u>
7	Page
8	Frame
23	Table
25	Cell
35	Paragraph
39	Character

107 Header  
108 Footer  
676 DropCap

**Return value**

A string representing the name of the object created.

**Usage**

In order to create an object which you can see in your document, you must call this method from the Foundry object found in the Foundry property on WPApplication or a Division. DO NOT use the Foundry object stored in WPApplication.AppFoundry or TextDocument.Foundry.

### **Word Pro: CutSelection method**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_CUTSELECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes the current selection and places a copy in the Clipboard and the Foundry object located in the AppFoundry property on WPAApplication. Equivalent to choosing Edit - Cut.

#### **Syntax**

[objectreference].CutSelection()

#### **Parameters**

None

#### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

#### **Usage**

## **Word Pro: DarkMode method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DARKMODE\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is used to turn off the updating of the Word Pro screen during execution of a script.

### **Syntax**

[Objectreference].ApplicationWindow.DarkMode (Integer) Integer

### **Parameters**

*Integer*

Data type is Integer. The legal values for this parameter are -1 or 0 but you may use the LotusScript constants of True (-1) and False (0).

### **Return value**

The return values for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

When DarkMode is turned on, the screen display in the current MDI window is frozen. The screen is not updated until DarkMode is turned off. Make sure that you turn DarkMode off before the script finishes or the user will never see the screen repaint.

## **Word Pro: DataObjectGetDataHere method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATAOBJECTGETDATAHERE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].DataObjectGetDataHere(pFormatEtc,pStgMedium)

### **Parameters**

*pFormatEtc*

Data type is Long.

*pStgMedium*

Data type is Long.

### **Return value**

Integer

### **Usage**



## **Word Pro: DataObjectGetData method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DATAOBJECTGETDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].DataObjectGetData(pFormatEtc,pStgMedium)

### **Parameters**

*pFormatEtc*

Data type is Long.

*pStgMedium*

Data type is Long.

### **Return value**

Integer

### **Usage**

## **Word Pro: DbUnderline method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_DBLUNDERLINE\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the double underline attribute for selected text, or all following text if no text is selected. Acts as a toggle, turning the attribute off if it is on and on if it is off. Equivalent to choosing Text - Attributes - Other and then choosing "Db Underline" from the Attributes box.

### **Syntax**

[objectreference].DbUnderline()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: DeleteButton method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETEBUTTON\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a button from the status bar.

**Syntax**

[objectreference].DeleteButton()

**Parameters**

None

**Return value**

Integer. Always returns True.

**Usage**

Use this method when you want to delete a selected button from the status bar. Once you delete a button, you can no longer access it.

## **Word Pro: DeleteChars method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETECHARS\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes the specified number of characters following the insertion point.

### **Syntax**

[objectreference].DeleteChars(Count)

### **Parameters**

*Count*

An Integer expression which specifies the number of characters you want to delete.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

Similar to the forward delete (DEL or Delete) key on most keyboards.

## Word Pro: DeleteContainer method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCO  
LSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETECONTAINER\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a container object.

### Syntax

[objectreference].DeleteContainer()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

### Word Pro: DeleteContents method

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_DELETECONTENTS_METHOD_EXSCRIPT',1)} See example
```

Deletes the contents from an object.

### Syntax

```
[objectreference].DeleteContents()
```

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

**Word Pro: DeleteContent method**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETECONTENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes the content in a GraphicOleObject.

**Syntax**

[objectreference].DeleteContent()

**Parameters**

None

**Return value**

Integer

**Usage**

## **Word Pro: DeleteDivision method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETEDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes the currently active division object, its contents, and any child divisions. Equivalent to clicking the right mouse button on the division divider tab and choosing Delete Division.

### **Syntax**

[objectreference].DeleteDivision([DivisionName])

### **Parameters**

*DivisionName*

Data type is String. Optional parameter.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

Every document must have at least one division, so you cannot delete the last remaining division in a document.



## **Word Pro: DeleteField method**

{button ,AL('H\_DOCINFOFIELDMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETEFIELD\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes a document field from a document.

### **Syntax**

[objectreference].DeleteField(FieldName)

### **Parameters**

*FieldName*

The name of the field you want to delete. Data type is String.

### **Return value**

Returns True if the specified document field is deleted. Returns False if the specified document field is not deleted or if the document field does not exist.

### **Usage**

Use this method to delete a specific document field and its contents.

## Word Pro: Deleteltem method

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETEITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a menu item from a menu item object.

### Syntax

[objectreference].Deleteltem(MenuText)

### Parameters

*MenuText*

The String name of the menu item you want to delete from a menu.

### Return value

String

### Usage

You call the Deleteltem method from the parent menu and specify which item you want to delete from the Items property of the parent.

If you delete an existing Word Pro menu item, Word Pro regenerates the menu item when you terminate the current Word Pro session and launch Word Pro again. To disable a Word Pro menu item, you must create a startup script to remove the item each time you start Word Pro. Creating a startup script to remove an item is often used to prevent duplicate menu items on the Word Pro menu. By calling Deleteltem with the text you intend to use in the NewItem method, Word Pro does not add a second item if that item already exists.

## **Word Pro: DeleteKey method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETEKEY\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes the current selection. If nothing is selected, this method deletes the object in front of the insertion point. Equivalent to pressing the Delete key one time.

### **Syntax**

[objectreference].DeleteKey()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: DeleteLayout method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_DELETELAYOUT_METHOD_EXSCRIPT',1)} See example
```

Deletes a specified layout object and its contents.

### Syntax

[objectreference].DeleteLayout()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Word Pro does not delete the contents of a layout object if another object is using those contents. For example, if cell A1 is referenced in a formula, you can use the DeleteLayout method to delete the celllayout object of cell A1.

However, the content of that celllayout object will still be included in the calculation of the formula. The result of the formula will not change until the content of cell A1 is deleted as well.

## **Word Pro: DeleteMacroAccelerator method**

{button ,AL('H\_ACCELERATORS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETEMACROACCELERATOR\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes the macro accelerator key.

### **Syntax**

[objectreference].DeleteMacroAccelerator(MacroName)

### **Parameters**

*MacroName*

Data type is String.

### **Return value**

### **Usage**

### **Word Pro: DeleteMarker method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETEMARKER\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a marker from a document.

### **Syntax**

[objectreference].DeleteMarker()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: DeleteParallelColumns method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETEPARALLELCOLUMNS\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a parallel column from a document. Equivalent to choosing Columns - Delete - All Columns.

### **Syntax**

[objectreference].DeleteParallelColumns()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: DeleteSection method**

{button ,AL('H\_INDEXSECTION\_CLASS;H\_SECTION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETESECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a section object from a division. Equivalent to clicking the right mouse button on the section divider tab and choosing Delete Section.

**Syntax**

[objectreference].DeleteSection()

**Parameters****Return value****Usage**



## **Word Pro: DeleteSmartCorrect method**

{button ,AL('H\_SMARTCORRECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETESMARTCORRECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Turns off the SmartCorrect tool in a document.

### **Syntax**

[objectreference].DeleteSmartCorrect(Entry)

### **Parameters**

*Entry*

Data type is String.

### **Return value**

### **Usage**

## Word Pro: DeleteTable method

{button ,AL(^H\_BASetable\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DELETETABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

[WPAApplication]

Deletes the table object which has the focus. If no table object is in the focus, nothing is deleted.

[Table]

Deletes an entire table or specified components of a table.

### Syntax

[objectreference].DeleteTable()

[objectreference].DeleteTable(TableDelType, Start, Count)

### Parameters

[WPAApplication]

None

[Table]

*TableDelType*

Indicates which type of item is to be deleted. The value of this Variant parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpTableDelTypeRow (1867)	Deletes any rows which are included in the current selection, unless you specify values for the Start and Count parameters.
\$LwpTableDelTypeColumn (1868)	Deletes any columns which are included in the current selection, unless you specify values for the Start and Count parameters.
\$LwpTableDelTypeWholetable (1869)	Deletes the entire table.

*Start*

An Integer parameter which indicates the beginning column or row ID to be deleted. The ID is zero based, which means that the first row and column within a table have an ID of 0. The Start parameter must be used in combination with the Count parameter. This parameter has no effect when the TableDelType parameter value is 1869.

*Count*

An Integer parameter which indicates the number of rows or columns to be deleted. The Count parameter must be used in conjunction with the Start parameter. This parameter has no effect when the TableDelType parameter value is 1869.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

[WPAApplication]

Equivalent to choosing Table - Delete - Entire Table.

[Table]

Equivalent to choosing Table - Delete and choosing the object to be deleted.

## **Word Pro: DeleteTab method**

{button ,AL('H\_TABRACK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETETAB\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a tab setting from the horizontal ruler. Equivalent to choosing View - Show/Hide - Ruler to display the horizontal ruler, then dragging a tab off the ruler and releasing the mouse button. Also equivalent to clicking the right mouse button anywhere over the ruler and choosing Clear All Tabs.

### **Syntax**

[objectreference].DeleteTab(Index)

### **Parameters**

*Index*

Data type is Integer.

### **Return value**

### **Usage**

**Word Pro: DeleteVersion method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETEVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a version of a document. Equivalent to choosing File - Versions and clicking Delete Version on the version you want to delete.

**Syntax**

[objectreference].DeleteVersion(VersionID)

**Parameters**

*VersionID*

Data type is Long.

**Return value****Usage**

## Word Pro: Delete method

{button ,AL('H\_BAG\_CLASS;H\_BASETABLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DELETE\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes an object type from a Bag object. Deletes an object type from a Glossary or BaseTable object. This method is defined in the following classes:

[Bag]

[BaseTable]

Deletes row, column, and cell objects in a table object.

## Syntax

[objectreference].Bag.Delete()

[objectreference].BaseTable.Delete(TableDelType,[Start,] [Count])

## Parameters

*TableDelType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTableDelTypeColumn (1868)

\$LwpTableDelTypeRow (1867)

\$LwpTableDelTypeWholetable (1869)

*Start*

Data type is Integer. Optional parameter.

*Count*

Data type is Integer. Optional parameter.

## Return value

Bool

## Usage

**Word Pro: DemoteOutlineLevel method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DEMOTEOUTLINELEVEL\_METHOD\_EXSCRIPT',1)} [See example](#)

Demotes the current paragraph to the next lower outline level. Equivalent to choosing Text - Outline - Demote.

**Syntax**

[objectreference].DemoteOutlineLevel()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: Demote method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DEMOTE\_METHOD\_EXSCRIPT',1)} [See example](#)

Demotes a paragraph to the next lower outline level. Equivalent to choosing Text - Outline - Demote.

**Syntax**

[objectreference].Demote()

**Parameters**

None

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

This method affects the paragraph or paragraphs in the Text object from which you call the method.

## **Word Pro: DeselectRuler method**

{button ,AL('H\_RULER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DESELECTRULER\_METHOD\_EXSCRIPT',1)} [See example](#)

Word Pro calls this method when the ruler is hidden, and deselects a ruler mouse filter context on the context stack that intercepts mouse messages.

### **Syntax**

[objectreference].DeselectRuler()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

When the ruler is present, the mouse filter context filters mouse move messages and checks to see if a frame is currently being dragged. If a frame is being dragged, Word Pro displays special guides on the ruler.



**Word Pro: Deselect method**

```
{button ,AL(^H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL(^H_DESELECT_METHOD_EXSCRIPT',1)} See example
```

Deselects the current selection in a document. Equivalent to deselecting by clicking outside the current selection or by pressing Esc.

**Syntax**

```
[objectreference].Deselect()
```

**Parameters**

None

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: DestroyDocWindow method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DESTROYDOCWINDOW\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].DestroyDocWindow()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: Destroy method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DESTROY\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Destroy()

### **Parameters**

### **Return value**

### **Usage**

**Word Pro: DisconnectCells method**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)  
{button ,AL(^H\_DISCONNECTCELLS\_METHOD\_EXSCRIPT',1)} [See example](#)

Disconnects table cells which have been connected using the Table menu commands, the ConnectCells method, or the ConnectRows method.

**Syntax**

[objectreference].DisconnectCells()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

Equivalent to choosing Table - Disconnect Cell.

### **Word Pro: Disconnect method**

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCO  
NTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_DISCONNECT_METHOD_EXSCRIPT',1)} See example
```

Disconnects grouped container objects.

### **Syntax**

```
[objectreference].Disconnect()
```

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: DivideText method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DIVIDETEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].DivideText(TargetDivision,[TargetDivision2])

### **Parameters**

*TargetDivision*

Data type is String.

*TargetDivision2*

Data type is String. Optional parameter.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: DoesMarkerNameMatch method

{button ,AL(^H\_BASETABLE\_CLASS;H\_CELLGROUPLAYOUT\_CLASS;H\_CELLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GLOSSARY\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_PARALLEL\_COLUMNS\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBY\_LAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_DOESMARKERNAMEMATCH\_METHOD\_EXSCRIPT',1)} [See example](#)

Passes the name of the marker to determine whether the current object is marked by that marker name.

### Syntax

[objectreference].DoesMarkerNameMatch(MarkerName)

### Parameters

*MarkerName*

A String expression that represents the name assigned to the marker that is passed to determine whether the current object is marked by that marker name.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: DoneWithRightContextMenu method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_DONEWITHRIGHTMOUSEMENU\_METHOD\_EXSCRIPT',1)} [See example](#)

Indicates that the right mouse menu was used to create a graphic.

### **Syntax**

[objectreference].DoneWithRightContextMenu(MenuHandle)

### **Parameters**

*MenuHandle*

Data type is Long.

### **Return value**

Integer

### **Usage**



**Word Pro: EditClickHereLink method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_EDITCLICKHERELINK\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens the Edit Link dialog box so the user can edit a ClickHere link.

**Syntax**

[objectreference].EditClickHereLink()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

For this method to work, the insertion point must be in the ClickHereLink text or the ClickHereLink text must be selected. If the insertion point is not in the ClickHereLink text or other text is selected, this method will not function.

**Word Pro: EditLinkInfo method**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EDITLINKINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to update whatever you are linked to. Equivalent to choosing Edit - Manage Links..

**Syntax**

[objectreference].EditLinkInfo(LinkInfo)

**Parameters**

*LinkInfo*

Data type is String.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

When you choose Edit - Manage Links, you can change DdeLink information by editing it, changing the name of the link, opening the link source, or breaking the link.

**Word Pro: EmbedFormula method**

{button ,AL('H\_FORMULA\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EMBEDFORMULA\_METHOD\_EXSCRIPT',1)} [See example](#)

Embeds a formula in a table object.

**Syntax**

[objectreference].EmbedFormula(ObjectName)

**Parameters**

*ObjectName*

A String expression representing the name of the object.

**Return value****Usage**

## Word Pro: Embed method

```
{button ,AL(^H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL(^H_EMBED_METHOD_EXSCRIPT',1)} See example
```

Inserts the named object into the layout of the object from which you call this method: PageLayout, SuperTableLayout, FrameLayout.

### Syntax

```
[objectreference].Embed(ContentName, [Advance])
```

```
[objectreference].Embed(Objectname, [Advance])
```

### Parameters

*ContentName*

Data type String.

*Advance*

Indicates whether or not the insertion point is left at the beginning of the embedded object or is advanced to the end of the object. Data type is Boolean. Optional parameter. Default is True.

*Objectname*

Data type is String.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: EndChange method

{button ,AL(^H\_DIVISION\_CLASS;H\_DOCWINDOW\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ENDCHANGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Ends any changes made to the Word Pro application, a document window, a division, or a text document. This method is defined in the following classes: WPAApplication, Division, TextDocument, DocWindow.

### Syntax

[objectreference].WPAApplication.EndChange([MarkChanges])

[objectreference].Division.EndChange([MarkChanges])

[objectreference].Text.DocumentEndChange([MarkChanges])

[objectreference].DocWindow.EndChange()

### Parameters

*MarkChanges*

Data type is Boolean. Optional parameter. Default is False.

### Return value

Boolean

### Usage

## **Word Pro: EndCustomLines method**

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENDCUSTOMLINES\_METHOD\_EXSCRIPT',1)} [See example](#)

This method only displays in the LotusScript Editor during a recording to reflect the end of a customized table line style selection.

### **Syntax**

[objectreference].EndCustomLines()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: Ending method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCO  
LSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ENDING_METHOD_EXSCRIPT',1)} See example
```

In any container this method moves the insertion point from its current position to the end of the document.

## Syntax

[objectreference].Ending (ContainerEndEndOfDocument)

## Parameters

### *ContainerEnd*

The end of a specific container. Data type is Boolean. The value of this parameter must be "\$LwpContainerEndEndOfDocument" or its code equivalent (160).

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: EnumerateChartLinks method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENUMERATECHARTLINKS\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].EnumerateChartLinks(LinkCookie)

**Parameters**

*LinkCookie*

Data type is Long.

**Return value**

Long

**Usage**



## **Word Pro: EnumerateTerm method**

{button ,AL('H\_GLOSSARY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENUMERATETERM\_METHOD\_EXSCRIPT',1)} [See example](#)

Enumerates all the entries in a glossary.

### **Syntax**

[objectreference].EnumerateTerm(KeyNumber)

### **Parameters**

*KeyNumber*

The number associated with each entry in a glossary. Word Pro uses this number to enumerate the entries in the glossary.

### **Return value**

String

### **Usage**

## **Word Pro: EnvelopePrint method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ENVELOPEPRINT\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].EnvelopePrint()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: ExchangeItem method**

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXCHANGEITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Switches one menu item with another menu item in the same or different parent menu item.

### **Syntax**

[objectreference].ExchangeItem(WithItem)

### **Parameters**

*WithItem*

Specifies the menu from which you are moving the item.

### **Return value**

### **Usage**

This method allows you to switch the position of two menu items in any menu. For example, you could switch the location of two items anywhere on the Word Pro menu.

## **Word Pro: ExpandOutline method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPANDOUTLINE\_METHOD\_EXSCRIPT',1)} [See example](#)

Expands the highest level contracted heading(s) which are subordinate to the paragraph for which you are calling the method. For example, when you call this method for a Level 1 heading, it will expand the highest level contracted heading(s) which are subordinate to that Level 1 heading.

### **Syntax**

[objectreference].ExpandOutline([All])

### **Parameters**

*All*

Allows you to expand all the subordinate headings under the heading from which you call this method. Data type is Integer but the legal values for this parameter are -1 and 0. You may use the LotusScript constants True (-1) and False (0). A value of True causes all subordinate headings to be expanded, regardless of their level. Optional parameter. Default is True.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: Expand method**

{button ,AL('H\_SECTIONTABS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPAND\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays divider tabs that are the children of a parent division tab in a document.

**Syntax**

[objectreference].Expand()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to clicking the plus sign on the parent division tab to show all the children divider tabs.

'Example: DebugVariable property

'This example script has not yet been created.

'Example: DefaultBinName property

'This example script has not yet been created.

'Example: DefaultCellStyleDescription property  
'This example script has not yet been created.



'Example: DefaultColumnName property

'This example script has not yet been created.

'Example: DefaultFilePath property  
'This example script has not yet been created.

'Example: DefaultFrameStyleDescription property  
'This example script has not yet been created.

'Example: DefaultLatinFont property  
'This example script has not yet been created.

'Example: DefaultLeftColumnStyleName property  
'This example script has not yet been created.

'Example: DefaultNonLatinFont property  
'This example script has not yet been created.

'Example: DefaultPageHeight property

'This example script has not yet been created.

'Example: DefaultPageStyleDescription property  
'This example script has not yet been created.



'Example: DefaultPageWidth property

'This example script has not yet been created.

'Example: DefaultPitch property

'This example script has not yet been created.

'Example: DefaultRightColumnNameStyleName property  
'This example script has not yet been created.

'Example: DefaultTableStyleDescription property  
'This example script has not yet been created.

'Example: DefaultTextStyleDescription property  
'This example script has not yet been created.

'Example: DefCellStyleName property  
'This example script has not yet been created.

'Example: DefColWidth property

'This example script has not yet been created.

'Example: Definition property

'This example script has not yet been created.



'Example: DefRowHeight property

'This example script has not yet been created.

```
'Example: DeleteButton method  
' This example removes all custom statusbar buttons and repaints the  
' statusbar.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim StatBar As StatusBar  
Set StatBar = .ApplicationWindow.StatusBar  
Forall Button In StatBar.StatusBarButtons  
    ButtonId = Button.GetButtonId  
    If Button.GetButtonType = 153 Then  
        Print Button.DeleteButton()  
    End If  
End Forall  
StatBar.InvalidateWholeBar
```

```
'Example: DeleteChars method
' This example inserts two words into the current document. After the
' message box is closed, the first five characters are deleted.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "Some text."
  MessageBox "Click OK to delete the first 5 characters.",MB_OK,"Example Script"
.Text.Backward $LwpNavigateObjectTypeWord, 2
.Text.DeleteChars 5
```

'Example: DeleteContainer method

'This example script has not yet been created.

'Example: DeleteContents method

' This example deletes the contents of a Click Here Block.

' RUNTIME DEPENDENCIES: You must have a document open and a click here

' named 'ClickHere1' for this script to work.

.Division.Foundry.ClickHeres.Item("ClickHere1").DeleteContents

'Example: DeleteContent method

'This example script has not yet been created.

'Example: DeleteDivision method

- ' This example creates a new division based on the "DEFAULT.MWP" Smart Master. It is placed after the current
- ' division. After the division is created it is deleted.
- ' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SmartMaster as String

Dim NewDivId as String

SmartMaster = .ApplicationWindow.UserInterfacePrefs.StylePath & "\DEFAULT.MWP"  
.CreateDivision SmartMaster, "", \$LWPDivLocInsertAfterCurrentDiv, "", ""

NewDivId = .Division.Name

**.DeleteDivision(NewDivId)**

```
'Example: DeleteField method
' This example adds a new field named 'ExampleField' for the current document.
' The field's contents are inserted and the field is then deleted.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ActiveDocument.DocInfo.FieldManager.AddField "ExampleField", "Some data for
ExampleField ", 1
.InsertDocInfo $LwpDocVarField, "ExampleField"

.ActiveDocument.DocInfo.FieldManager.DeleteField "ExampleField"
```



'Example: DeleteFont property

'This example script has not yet been created.

```
'Example: DeleteItem method
' This example creates a new menu item named 'Example Menu' to the File
' menu. The mnuMenuSub subroutine is assigned to run each time the new
' menu item is selected
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim NewMenu As MenuItem
Dim MenuName as String
Dim MenuSpacer as String

MenuName = "&Example Menu"
MenuSpacer = Chr$(8)

' Set menu object
Set NewMenu = .ApplicationWindow.LwpMenuBar.Items.Item("&File")

' Create a new menu off of the File Menu and before the Save option
' Delete it first to prevent duplicates
NewMenu.DeleteItem MenuName
NewMenu.NewItem MenuName, "!mnuMenuSub", 0, "&Save" & MenuSpacer & "Ctrl+S"
End Sub
```

```
'Example: DeleteKey method
' This example positions the insertion point at the beginning of the current
' paragraph. Some text is inserted and then each character is deleted.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim Message As String
Dim LenMessage As Integer
Dim x As Integer

Message = "This is some text"
LenMessage = Len(Message)
.Text.MoveToStart $LwpLocationTypeParagraph
.Type Message
.Text.MoveToStart $LwpLocationTypeParagraph
For x = 1 To LenMessage
    .DeleteKey
Next
```

'Example: DeleteLayout method

'This example script has not yet been created.

'Example: DeleteMacroAccelerator method

'This example script has not yet been created.

'Example: DeleteMarker method

'This example script has not yet been created.

```
'Example: DeleteParallelColumns method
' This example creates then deletes a parallel column for the current
' document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, $LtsAlignmentHorizCenter
.DeleteParallelColumns
```

```
'Example: DeleteSection method
' This example inserts several sections in the active division and then
' deletes all sections.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
.InsertSection "Default Page", True, True, $LwpStartTypeNextpage, False, True
Forall Section In .Division.Foundry.Sections
    Section.DeleteSection
End Forall
```



```
'Example: DeleteSmartCorrect method  
' This example deletes an entry in the SmartCorrect list.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Language = "English (United States)"  
.Application.SmartCorrects(Language).DeleteSmartCorrect "lts"
```

```
'Example: DeleteTable method
' This example creates a table with 4 rows and 5 columns, then deletes the
' table when you click OK in the message box.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
MessageBox "Click OK to delete the current table.",MB_OK,"Example Script"

.DeleteTable
```

'Example: DeleteTab method

'This example script has not yet been created.

```
'Example: DeleteVersion method
' This example creates a version for the current document then deletes the
' version.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Division.VersionManager.CreateVersion "NewVersion"
Forall Version In .ActiveDocument.VersionManager.Versions
    If Version.name = "NewVersion" Then
        .ActiveDocument.VersionManager.DeleteVersion Version.DocVersionId
    End If
End Forall
```

'Example: Delete method

'This example script has not yet been created.

'Example: DemandLoad property

'This example script has not yet been created.

'Example: DemoteOutlineLevel method

'This example script has not yet been created.

'Example: Demote method

'This example script has not yet been created.



'Example: Descent property

'This example script has not yet been created.

'Example: DescriptionFileName property

'This example script has not yet been created.

'Example: Description property

'This example script has not yet been created.

'Example: DeselectRuler method

'This example script has not yet been created.

```
'Example: Deselect method
' This example inserts some text into the current document which is then
' selected. After the message box is closed, the text is deselected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "This is some text."
MessageBox "Click OK to select the text.",MB_OK,"Example Script"
.Select $LwpSelectObjectTypeParagraph
MessageBox "Click OK to deselect the text.",MB_OK,"Example Script"
.Text.Deselect
```

'Example: DestroyDocWindow method

'This example script has not yet been created.

'Example: Destroy method

'This example script has not yet been created.

'Example: DirectionDown property

'This example script has not yet been created.



'Example: DirectionLeft property

'This example script has not yet been created.

'Example: DirectionRight property

'This example script has not yet been created.

'Example: DirectionUp property

'This example script has not yet been created.

'Example: DirectiveColor property

'This example script has not yet been created.

'Example: DisableClickHeres property  
'This example script has not yet been created.

'Example: DisableExportToNotes property  
'This example script has not yet been created.

'Example: DisableVersionReview property

'This example script has not yet been created.

```
'Example: DisconnectCells method
' This example creates a table with 5 columns and 4 rows based on the
' Default Table style, selects the first column, and connects the cells.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.SelectColumn
.ConnectCells
MessageBox "Click OK to disconnect cells .",MB_OK,"Example Script"
.DisconnectCells
```



```
'Example: Disconnect method
' This example creates a table and connects the first row of cells. After the
' message box is closed, the table cells are disconnected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.SelectRow
.ConnectCells
MessageBox "Click OK to disconnect cells. ", MB_OK, "Example Script"
.TableContainer.Disconnect
```

'Example: DistanceFromLeftMargin property

'This example script has not yet been created.

```
'Example: DivideText method
' This example combines two divisions names 'Body' and 'Division' into one
' division.
' RUNTIME DEPENDENCIES: You must have a document open containing these two
' divisions for this script to work.
```

```
Dim DivIdName As String
Dim DivInfoName As String
Dim DivId1 As String
Dim DivId2 As String
```

```
'Get the hexadecimal id for the 'Body' division
DivInfoName = "Body"
Gosub GetDivId
DivId1 = DivIdName
```

```
'Get the hexadecimal id for the 'Division' division
DivInfoName = "Division"
Gosub GetDivId
DivId2 = DivIdName
```

```
'Move text from "Body" to "Division" using their hexadecimal ids.
```

```
.Text.DivideText DivId1,DivId2
```

```
Exit Sub
```

```
GetDivId:
```

```
Forall Div In .ActiveDocument.Divisions
```

```
    If (Ucase$(Div.Divisioninfo.Name) = Ucase$(DivInfoName)) Then
```

```
        DivIdName = Div.Name
```

```
    End If
```

```
End Forall
```

```
Return
```

'Example: DivisionInfo property

'This example script has not yet been created.

'Example: DivisionNames property

'This example script has not yet been created.

'Example: DivisionName property

'This example script has not yet been created.

'Example: DivisionOptions property  
'This example script has not yet been created.

'Example: DivisionsRequired property

'This example script has not yet been created.



'Example: Divisions property

'This example script has not yet been created.

'Example: Division property

'This example script has not yet been created.

'Example: DocControlRestrictedToEditor property  
'This example script has not yet been created.

'Example: DocControl property

'This example script has not yet been created.

'Example: DocInfo property

'This example script has not yet been created.

'Example: DocOptions property

'This example script has not yet been created.

'Example: DocPath property

'This example script has not yet been created.

'Example: DocSize property

'This example script has not yet been created.



'Example: DocTextSize property

'This example script has not yet been created.

'Example: DocumentClosed event

'This example script has not yet been created.

'Example: DocumentClose event

'This example script has not yet been created.

'Example: DocumentCreated event  
,

'Example: DocumentCreate event

'This example script has not yet been created.

'Example: DocumentExported event

'This example script has not yet been created.

'Example: DocumentExport event

'This example script has not yet been created.

'Example: DocumentImported event

'This example script has not yet been created.



'Example: DocumentImport event

'This example script has not yet been created.

'Example: DocumentInserted event

'This example script has not yet been created.

'Example: DocumentInsert event

'This example script has not yet been created.

'Example: DocumentLevel property

'This example script has not yet been created.

'Example: DocumentOpened event  
,

'Example: DocumentOpen event  
,

'Example: DocumentPaths property

'This example script has not yet been created.

'Example: DocumentPrinted event

'This example script has not yet been created.



'Example: DocumentPrint event

'This example script has not yet been created.

'Example: DocumentSaveAs event

'This example script has not yet been created.

'Example: DocumentSaved event

'This example script has not yet been created.

'Example: DocumentSave event

'This example script has not yet been created.

'Example: Documents property

'This example script has not yet been created.

'Example: Document property

'This example script has not yet been created.

'Example: Foundry property  
,

'Example: FindWithProperties property  
'This example script has not yet been created.



```
'Example: Find method
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script finds the first 'cat'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"
.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"

.Find
```

'Example: FirstChild property

'This example script has not yet been created.

'Example: FirstName property

'This example script has not yet been created.

'Example: FirstPage property

'This example script has not yet been created.

'Example: First property

'This example script has not yet been created.

'Example: FitType property

'This example script has not yet been created.

'Example: FixAcronymns property

'This example script has not yet been created.

'Example: FixBullets property

'This example script has not yet been created.



'Example: FixMargins property

'This example script has not yet been created.

'Example: FontColor property

'This example script has not yet been created.

'Example: FontMetrics property

'This example script has not yet been created.

```
'Example: FontName property
Sub Main
Print "======"
Forall x In .division.foundry.paragraphstyles
    Print x.font.ActualName & " = " & x.font.size
End Forall
Forall x In .Division.foundry.paragraphstyles
    x.font.FontName = "Arial"
End Forall
Print " -----"
Forall x In .Division.foundry.paragraphstyles
    Print x.font.ActualName & " = " & x.font.size
End Forall
End Sub
```

'Example: FontSize property

'This example script has not yet been created.

'Example: FontStyleName property

'This example script has not yet been created.

'Example: FontUnitName property

'This example script has not yet been created.

'Example: FontUnits property

'This example script has not yet been created.



```
'Example: Font property
Sub Main
Print "======"
Forall x In .division.foundry.paragraphstyles
    Print x.font.ActualName & " = " & x.font.size
End Forall
Forall x In .Division.foundry.paragraphstyles
    x.font.FontName = "Arial"
End Forall
Print " -----"
Forall x In .Division.foundry.paragraphstyles
    Print x.font.ActualName & " = " & x.font.size
End Forall
End Sub
```

'Example: Footers property

'This example script has not yet been created.

'Example: Footer property

'This example script has not yet been created.

'Example: FootnoteContSep property

'This example script has not yet been created.

'Example: FootnoteLayouts property

'This example script has not yet been created.

'Example: FootnoteNumbering property

'This example script has not yet been created.

'Example: FootnoteOptions property  
'This example script has not yet been created.

'Example: FootnoteSeparator property

'This example script has not yet been created.



'Example: Footnotes property

'This example script has not yet been created.

'Example: ForceDocToLoad method

'This example script has not yet been created.

'Example: FormatCheckAction method

'This example script has not yet been created.

'Example: FormatCheckLevel property

'This example script has not yet been created.

'Example: FormatCheckPreferences property  
'This example script has not yet been created.

'Example: FormatCheckReplace method  
'This example script has not yet been created.

'Example: FormatCheckRule property  
'This example script has not yet been created.

'Example: FormatCheckSuggestion property

'This example script has not yet been created.



'Example: FormatCheck method

'This example script has not yet been created.

'Example: FormatType property

'This example script has not yet been created.

'Example: Format property

'This example script has not yet been created.

'Example: Formula property

'This example script has not yet been created.

```
'Example: Forward method
' This example inserts a sentence of text into the current document. Each word
' starting with the first is then selected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "This is a Sentence of text."
.Text.MoveToStart $LwpLocationTypeLine

Do
    .SelectWord
    NextWord = .Text.Forward ($LwpNavigateObjectTypeWord, 1)
Loop Until (.Text.AtBeginningOfParagraph = True) Or (NextWord = False)
```

'Example: FrameRevert method

' This example inserts a frame into the current document and changes the  
' frame's background color. After the message box is closed the frame is  
' reverted to the default frame style attributes.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewFrame 3285, 1200, 1575, 1830

.Frame.Layout.Background.Color.Red = 82

.Frame.Layout.Background.Color.Blue = 239

.Frame.Layout.Background.Color.Green = 145

.Frame.Layout.Background.Color.Override = \$LwpColorOverrideRgb

.Frame.Anchor \$LwpAnchorWhereLayout, \$LwpConditionTypeOnlyspecificpage,  
\$LwpRelativeTypeLytParent

MessageBox "Click OK to revert frame to default attributes in frame style.

",MB\_OK,"Example Script"

**.FrameRevert**

'Example: FrameStyleName property

'This example script has not yet been created.

'Example: FrameStyles property

'This example script has not yet been created.



'Example: Frames property

'This example script has not yet been created.

'Example: Frame property

'This example script has not yet been created.

'Example: FreeMenus property

'This example script has not yet been created.

```
'Example: FullName property
With WordPro.ActiveDocument
MsgTxt = "Current Word Pro Doc is " & .FullName
MsgTxt = MsgTxt & ", the author is " & .Docinfo.AuthorName
Msgbox MsgTxt,64,"Word Pro Information"
MsgTxt = "It was created on " & .DocInfo.CreationDateString & " at "
& .DocInfo.CreationTimeString
Msgbox MsgTxt,64,"Word Pro Information"
End With
```

'Example: FXGetNotesString method

'This example script has not yet been created.

'Example: FXGetNotesWriteHandle method

'This example script has not yet been created.

'Example: FXSetNotesString method

'This example script has not yet been created.

'Example: GapBetweenPanels property  
'This example script has not yet been created.



'Example: GetActiveList method

'This example script has not yet been created.

'Example: GetAfidHelpFileName property

'This example script has not yet been created.

'Example: GetAfidHelpInfo property

'This example script has not yet been created.

'Example: GetArrayProp method

'This example script has not yet been created.

'Example: GetButtonId property

'This example script has not yet been created.

'Example: GetButtonType property

'This example script has not yet been created.

'Example: GetContents method

'This example script has not yet been created.

'Example: GetConversationHandle property  
'This example script has not yet been created.



```
'Example: GetCopyFormatCategories method
```

```
' This example retrieves the copy format categories and tests whether the 'text category is available.
```

```
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim FormatCategories as Integer
```

```
FormatCategories = .Text.GetCopyFormatCategories
```

```
If (FormatCategories And &H1) then
```

```
    MessageBox "Text format available.",MB_OK,"Example Script"
```

```
End If
```

```
If (FormatCategories And &H2) then
```

```
    MessageBox "Table format available.",MB_OK,"Example Script"
```

```
End If
```

```
'Example: GetCurrentMarkerName method
' This example displays the name of any clickhere or bookmark located
' at the cursor position. If the cursor position is not located on a
' clickhere or bookmark then no message is displayed.
' RUNTIME DEPENDENCIES: You must have a document open and the cursor
' located on a clickhere or bookmark for this script to work.
```

```
Dim MarkerName as string
```

```
MarkerName = .Text.GetCurrentMarkerName($LwpMarkerTypeClickhere)
```

```
If MarkerName <> "" then
    MessageBox "Clickhere found: " & MarkerName
Else
    MarkerName = .Text.GetCurrentMarkerName($LwpMarkerTypeBookmark)
    If MarkerName <> "" then
        Forall Marks In .Division.BookMarkManager.BookMarks
            If MarkerName = Marks.MarkerName Then
                MessageBox "Bookmark found: " & Marks.Name
            End If
        End Forall
    End If
End If
```

```
'Example: GetData method
' This example creates a dataset named 'ExampleDataSet' off of the active
' document. The 'FirstName' and 'LastName' items are created and filled with
' data. Finally the values for the dataset items are printed to the Script
' Editor Output panel. Since no dataset item named 'Address' was defined, the
' default dataset value will be printed in the last statement.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim DataSetName as String
Dim Defaultvalue as String
Dim DataSet As WPDataSetCollection

Set DataSet = .ActiveDocument.WPDataSets
DataSetName = "ExampleDataSet"
Defaultvalue = "Default"
DataSet(DataSetName).SetData "FirstName","John"
DataSet(DataSetName).SetData "LastName","Doe"

Print DataSet(DataSetName).GetData("FirstName",Defaultvalue)
Print DataSet(DataSetName).GetData("LastName",Defaultvalue)
Print DataSet(DataSetName).GetData("Address",Defaultvalue)
```

'Example: GetDefaultPageSize method

' This example prints the default page size for the active document to the.

' Lotus Script Output panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .ActiveDocument.PrintManager.GetDefaultPageSize()

'Example: GetDisplayableFilterName method

'This example script has not yet been created.

```
'Example: GetDocDescription method
' This example prints the document description for the active document to the
' Lotus Script Output panel.

Dim FullFilePath as String
FullFilePath = .ActiveDocument.FullName
Print .FileProtection.GetDocDescription($LwpFileExtractTypeFileprotection,
FullFilePath)
```

```
'Example: GetEnum method
' This example prints the numeric equivalent for the $LwpMergeActionNewfile
' constant to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim Value As Long
Value = .GetEnum("$LwpMergeActionNewfile")
Print Value
```

'Example: GetEnvelopeDefaults method

'This example script has not yet been created.



'Example: GetFileDescription method  
'This example script has not yet been created.

'Example: GetFilterExtension property

'This example script has not yet been created.

'Example: GetFilterExtForDialogBox property  
'This example script has not yet been created.

'Example: GetFilterId property

'This example script has not yet been created.

'Example: GetFormatName property

'This example script has not yet been created.

```
'Example: GetFormula method
' This example creates a table and enters a formula. The formula is then
' retrieved and printed to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.Table.CellLayout(4,0).GotoLayout
.Table.CellEngine.SetFormula 4,0,"@SUM(A1:A2)"
Print .Table.CellEngine.GetFormula(4, 0)
```

'Example: GetHomeDirectory property  
'This example script has not yet been created.

'Example: GetItemName property

'This example script has not yet been created.



'Example: GetLastUsedFilter method

' This example returns the last text filter type used in Word Pro.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .ApplicationWindow.Filter.GetLastUsedFilter(\$LwpFilterTypeText )

'Example: GetLineMix method

'This example script has not yet been created.

```
'Example: GetLineStyle method
' This example creates a table, changes the line style and then prints
' the line style to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.SelectEntireTable
.Table.TableLine.ChgLineStyle $LwpTableLineStyleAll
Print .Table.TableLine.GetLineStyle()
```

'Example: GetLinkDisplayNameFileLength method  
'This example script has not yet been created.

'Example: GetLinkName method

'This example script has not yet been created.

'Example: GetLinkSourceName method

'This example script has not yet been created.

```
'Example: GetMarkedText method
' This example creates a temporary text marker for the selected text and then
messages the text.
' RUNTIME DEPENDENCIES: You must have a document open with selected text
' for this script to work.
```

```
Dim MarkerName As String
Dim NewMarker As TextMarker
Dim MarkerText As String
MarkerName = .Mark($LWPMarkerTypeDefault)
Set NewMarker = .Division.Foundry.TextMarkers.Item(MarkerName)
MarkerText = NewMarker.GetMarkedText
```

```
'Example: GetMarkerName method
' This example displays the name of any clickhere or bookmark located
' at the cursor position. If the cursor position is not located on a
' clickhere or bookmark then no message is displayed.
' RUNTIME DEPENDENCIES: You must have a document open and the cursor
' located on a clickhere or bookmark for this script to work.
```

```
Dim MarkerName as string
MarkerName = .Text.GetCurrentMarkerName($LwpMarkerTypeClickhere)
If MarkerName <> "" then
    MsgBox "Clickhere found: " & MarkerName
Else
    MarkerName = .Text.GetCurrentMarkerName($LwpMarkerTypeBookmark)
    If MarkerName <> "" then
        Forall Marks In .Division.BookMarkManager.BookMarks
            If MarkerName = Marks.MarkerName Then
                MsgBox "Bookmark found: " & Marks.Name
            End If
        End Forall
    End If
End If
```



```

'Example: GetMispelledWord method
' This example sequentially highlights all misspelled words in the current
' document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

'move to the start of the document
.Text.MoveToStart $LwpLocationTypeDocument

Do
  ' get the current insertion point position on the page
  XPos1 = .Text.PositionXOnPage
  YPos1 = .Text.PositionYOnPage

  ' highlights the nearest misspelled word relative to insertion point
  Word = .Text.GetMispelledWord
  If (Word <> "") Then
    .Deselect
    .Type "[Right]"
  End If

  ' check position again to see if insertion point has moved
  XPos2 = .Text.PositionXOnPage
  YPos2 = .Text.PositionYOnPage

  ' make sure that don't keep checking the last misspelled word
  If ((XPos1 = XPos2) And (YPos1 = YPos2)) Then
    Exit Do
  End If
Loop While (Word <> "")

```

'Example: GetNamedProperty method

' This example creates a named property, 'ExampleProp' on the active document  
' and assigns it a value. The value is then printed to the Lotus Script Output  
' panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
.ActiveDocument.SetNamedProperty "ExampleProp", "Here is some data."
```

```
.ActiveDocument.GetNamedProperty "ExampleProp"
```

'Example: GetNameFromPage method

' This example prints the division name on page 1 to the Lotus Script Output panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .Division.GetNameFromPage(1)

'Example: GetObjectList method

'This example script has not yet been created.

'Example: GetOne method

'This example script has not yet been created.

'Example: GetParaNumber method

'This example script has not yet been created.

```
'Example: GetPasteFormatCategories method
' This example retrieves the paste format categories and tests whether several
' categories are available.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim FormatCategories As Integer
FormatCategories = .Text.GetPasteFormatCategories
Print FormatCategories
If (FormatCategories And &H1) Then
    MsgBox "Text format available.",MB_OK,"Example Script"
End If
If (FormatCategories And &H2) Then
    MsgBox "Table format available.",MB_OK,"Example Script"
End If
```

```

'Example: GetPosition method
' This example retrieves the position of the insertion point relative
' to the specified Marker object. A message box is displayed indicating
' the relative position.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MarkerName As String
Dim NewMarker As TextMarker

.Type "[Enter]"
.Type "Sample Text"
.SelectWord

' Create a new marker for the selected text
MarkerName = .Mark($LWPMarkerTypeDefault)
Set NewMarker = .Division.Foundry.TextMarkers.Item(MarkerName)

' Move the insertion point one line above the new marker
.Text.Deselect
.Text.MoveUp(1)

Position = .Text.GetPosition(MarkerName)
If Position = 0 Then
    MsgBox "Insertion point is located within the marker"
Elseif Position = 1 Then
    MsgBox "Insertion point is located after the marker"
Elseif Position = -1 Then
    MsgBox "Insertion point is located before the marker"
Elseif Position = -2 Then
    MsgBox "Insertion point is located in a different text stream than the marker"
End If

```



'Example: GetPowerFieldValue method

'This example script has not yet been created.

'Example: GetPrinterInfo method

' This example prints the printer name for the current document to the Lotus

' Script Output panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .ActiveDocument.PrintManager.GetPrinterInfo()

```
'Example: GetProtectionType method
' This example print the file protection type for the current document to
' the Lotus Script Output Panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim ProtectionType As Variant
FullFilePath = .ActiveDocument.FullName
ProtectionType
= .FileProtection.GetProtectionType($LwpFileExtractTypeFileprotection, FullFilePath)
Print ProtectionType
```

'Example: GetRedoWhatDesc property

'This example script has not yet been created.

'Example: GetRGB method

' This example creates a table with 5 rows and 5 columns into the current  
' document. The background and pattern colors are changed for the current  
' cell and the RGB color value is then printed to Lotus Script Output panel.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
.CreateTable False, "Default Table", 5,5  
.Table.TableFill.Background.Pattern = $LtsFillNwToSeGrad  
.Table.CurrentCell.Background.BackColor.SetRGB 255,255,255  
.Table.TableFill.Background.Color.Override = $LwpColorOverrideRgb  
.Table.TableFill.Background.BackColor.Red = 82  
.Table.TableFill.Background.BackColor.Blue = 239  
.Table.TableFill.Background.BackColor.Green = 145  
.Table.TableFill.Background.BackColor.SetRGB 82,239,145  
.Table.TableFill.Background.BackColor.Override = $LwpColorOverrideRgb  
.Table.TableFill.FillStyle = $LwpTableFillStyleAll
```

```
Print .Table.CurrentCell.Background.BackColor.GetRGB()
```

'Example: GetRightContextMenu method

'This example script has not yet been created.

'Example: GetServerName property

'This example script has not yet been created.

'Example: Foundry property  
,



'Example: DocumentSavedAs event

'This example script has not yet been created.

'Example: DocVersionID property

'This example script has not yet been created.

'Example: DocWindows property

'This example script has not yet been created.

'Example: DoesMarkerNameMatch method

'This example script has not yet been created.

'Example: DoInitialCaps property

'This example script has not yet been created.

'Example: DoneWithRightContextMenu method

'This example script has not yet been created.

'Example: DoSmartQuotes property

'This example script has not yet been created.

'Example: DoubleUnderline property

'This example script has not yet been created.



'Example: DragDropOn property

'This example script has not yet been created.

'Example: DriverNameDriverName property

'This example script has not yet been created.

'Example: DuplexType property

'This example script has not yet been created.

'Example: Editable property

'This example script has not yet been created.

'Example: EditClickHereLink method

'This example script has not yet been created.

'Example: EditLinkInfo method

'This example script has not yet been created.

'Example: EditorAttemptingOpen property

'This example script has not yet been created.

'Example: EditorInitials property

'This example script has not yet been created.



'Example: EditorManager property

'This example script has not yet been created.

'Example: EditorName property

'This example script has not yet been created.

'Example: Editors property

'This example script has not yet been created.

'Example: EditorVerificationType property

'This example script has not yet been created.

'Example: EffectiveColumnWidth property

'This example script has not yet been created.

'Example: EMail property

'This example script has not yet been created.

'Example: Embedded property

'This example script has not yet been created.

'Example: EmbedFonts property

'This example script has not yet been created.



'Example: EmbedFormula method

'This example script has not yet been created.

'Example: Embed method

'This example script has not yet been created.

'Example: EncryptPassword2 property

'This example script has not yet been created.

'Example: EncryptPassword property

'This example script has not yet been created.

```
'Example: EndChange method
' This example creates a table with 5 columns and 5 rows. Several background
' table cell properties are changed all at one using the BeginChange and
' EndChange methods.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ExampleTable As Table
Dim ExampleCell As CellLayout

.CreateTable False, "Default Table", 5,5
Set ExampleTable = .Table
ExampleTable.CellLayout(1,0).GotoLayout

Set ExampleCell = ExampleTable.CellLayout(1,0)

.BeginChange
ExampleCell.Content.InsertText "Hello"
With ExampleCell.BackGround
    .Pattern = $LtsFillSolid
    .Color.Red = 255
    .Color.Blue = 194
    .Color.Green = 255
    .Color.Override = $LwpColorOverrideRgb
    .BackColor.Red = 65
    .BackColor.Blue = 176
    .BackColor.Green = 0
End With

.EndChange
```

'Example: EndCustomLines method

'This example script has not yet been created.

'Example: EndingColOfSelection property  
'This example script has not yet been created.

'Example: EndingRowOfSelection property

'This example script has not yet been created.



```
'Example: Ending method
' This example moves the insertion point to the end of the document.
' RUNTIME DEPENDENCIES: You must have a document open and the insertion point
' within a frame for this script to work.
.CreateFrame False, "Default Frame",1440, 1440
.Page.Ending $LwpContainerEndEndOfDocument
```

'Example: EndnoteDivisionGroupNum property  
'This example script has not yet been created.

'Example: EndnoteDivisionNum property

'This example script has not yet been created.

'Example: EndnoteDocNum property

'This example script has not yet been created.

'Example: Endnotes property

'This example script has not yet been created.

'Example: EnterClickHere event

'This example script has not yet been created.

'Example: EnterLayout event  
,

'Example: EnumerateChartLinks method

'This example script has not yet been created.



```
'Example: EnumerateTerm method
' This example enumerates all the entries in the glossary named GLOSSARY.GLS.
' RUNTIME DEPENDENCIES: You must have a glossary named GLOSSARY.GLS and a
' document open for this script to work.

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = False
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = True
.GlossaryOpen "GLOSSARY.GLS", "Lotus Word Pro"
.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = True
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = False
Forall Gloss In .Division.Foundry.Glossarys
    Count% = Gloss.NumRows
    For Item% = 1 To (Count% - 1)
        Print "Item" & Format$(Item%) & "= " & Gloss.EnumerateTerm(Item%)
    Next
End Forall
.Close
```

```
'Example: Enumeration property
'This example sets the value of the Enumeration property to $LwpEnumScopeLocal.
'This script excludes any ParagraphStyle objects which originated
'with the SmartMaster from the ParagraphStyleCollection object in
'the currently active division's Foundry. This script then
'prints the names of any locally created ParagraphStyle objects
'in the Output panel.
'RUNTIME DEPENDENCIES: You must create one or more ParagraphStyle
'objects in addition to those in the SmartMaster. Otherwise, the
'Output panel will remain empty. You must have a document open for
'this script to work.
'Paste this example script into Sub Main and run it.

.Foundry.Enumeration = $LwpEnumScopeLocal
Forall ParagraphStyle In .Foundry.ParagraphStyles
    Print ParagraphStyle.Name
End Forall
```

'Example: EnvelopePrint method

'This example script has not yet been created.

'Example: Epoch property

'This example script has not yet been created.

'Example: EqnFontHeight property

'This example script has not yet been created.

'Example: ErrorColor property

'This example script has not yet been created.

```
'Example: ExchangeItem method
' This example exchanges the menu locations for the File and Create menus.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MenuBar As MenuItem
Set MenuBar = .ApplicationWindow.LwpMenuBar
MenuBar.Items("&File").ExchangeItem MenuBar.Items("&Create")
```

'Example: ExcludeRectBottom property

'This example script has not yet been created.



'Example: ExcludeRectLeft property

'This example script has not yet been created.

'Example: ExcludeRectRight property  
'This example script has not yet been created.

'Example: ExcludeRectTop property

'This example script has not yet been created.

```
'Example: ExitClickHere event
Sub Exitclickhere(Source As Clickhere, Clickherename As String)
Const Maximum = 15
Temp$ = Source.GetMarkedText
If Len(Temp$) > Maximum Then
    MsgBox "Contents is too long, try again"
    Source.Goto(True)
'This would notify the user of the problem,
'then put them back in the ClickHere block. The parameter True on the
'Source.Goto line is to actually select the contents of the ClickHere block.
'False would simply put them back in the ClickHere.
End If
End Sub
```

'Example: Expandable property

'This example script has not yet been created.

'Example: ExpandOutlineLevel method  
'This example script has not yet been created.

'Example: ExpandOutline method

'This example script has not yet been created.

```
'Example: Expand method
' This example creates two child divisions and then contracts and expands the
' divider tabs.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
.ActiveDocWindow.WinViewPrefs.IsViewSectionTabs = True
.ApplicationWindow.SectionTabs.ConnectSectionTabs
.ApplicationWindow.SectionTabs.ConnectSectionTabs
```

```
.ApplicationWindow.SectionTabs.Contract
.ApplicationWindow.SectionTabs.Expand
.ApplicationWindow.SectionTabs.Contract
```



```
'Example: ExportAllAsNotesFX method
' This example inserts two docinfo fields and makes them exportable to Lotus
' Notes.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ActiveDocument.DocInfo.IsExportedAsNotesFX $LwpDocVarVersionnumrevisions
.InsertDocInfo $LwpDocVarCreatedby
.Text.SplitParagraph
.InsertDocInfo $LwpDocVarDatecreated
.ActiveDocument.DocInfo.ExportAllAsNotesFX True
.ActiveDocument.DocInfo.UpdateSelectedFields
```

'Example: ExportAsNotesFX method

'This example script has not yet been created.

'Example: ExportToNotesFX property  
'This example script has not yet been created.

'Example: ExternalFileID property

'This example script has not yet been created.

'Example: ExternalFileName property  
'This example script has not yet been created.

'Example: ExternalName property

'This example script has not yet been created.

'Example: ExternalType property

'This example script has not yet been created.

```
'Example: ExtractText method
' This example opens the glossary file named GLOSSARY.GLS and prints all of
' the glossary entries to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = False
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = True
.GlossaryOpen "GLOSSARY.GLS", "Lotus Word Pro"
.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = True
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = False
Forall Gloss In .Division.Foundry.Glossarys
    Count% = Gloss.NumRows
    For Item% = 1 To (Count% - 1)
        GlossItem$ = Gloss.EnumerateTerm(Item%)
        GlossText$ = Gloss.ExtractText(GlossItem$)
        Print GlossText$           Next
    End Forall
.Documents("GLOSSARY.GLS").Close
```



'Example: FaceNames property

'This example script has not yet been created.

'Example: FaceStyleName property

'This example script has not yet been created.

'Example: FastFormatType property

'This example script has not yet been created.

```
'Example: FastFormat method
' This example enables Fast Format if it is not enabled and disables it if
' it is enabled. Note that the FormatStatus variable must have a Variant
' data type. This is because the FastFormatType property can return a String
' or an integer.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim FormatStatus As Variant
FormatStatus = .Application.Format.FastFormatType

If FormatStatus = $LwpFastFormatTypeNone Then
    .FastFormat True
Else
    .Application.Format.FastFormatType = $LwpFastFormatTypeNone
End If
```

'Example: FaxNumber property

'This example script has not yet been created.

'Example: FieldDelimiterText property

'This example script has not yet been created.

'Example: FieldDelimiter property

'This example script has not yet been created.

'Example: FieldManager property

'This example script has not yet been created.



'Example: FieldNumber property

'This example script has not yet been created.

'Example: Fields property

'This example script has not yet been created.

'Example: FieldType property

'This example script has not yet been created.

'Example: FileName property

'This example script has not yet been created.

'Example: FileProtectionType property

'This example script has not yet been created.

'Example: FileProtection property

'This example script has not yet been created.

'Example: FilesToCompare property

'This example script has not yet been created.

'Example: FileType property

'This example script has not yet been created.



'Example: FillerPageText property

'This example script has not yet been created.

'Example: FillStyle property

'This example script has not yet been created.

'Example: FilterHelper property

'This example script has not yet been created.

'Example: FilterName property

'This example script has not yet been created.

'Example: Filter property

'This example script has not yet been created.

```
'Example: FindAndReplace property
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script finds the first 'cat' then replaces it with
' 'dog'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace

.Application.FindAndReplace.FindString = "cat"
.Application.FindAndReplace.ReplaceString = "dog"

.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find
.ReplaceCmd
```

'Example: FindCellLayout method

'This example script has not yet been created.

```
'Example: FindClass method
' This example obtains the header class name for the current document and
' uses that name print some header properties to the Lotus Script Output
' panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim HeaderComponent As String
HeaderClass = .Layout.FindClass("Header")
Print .Division.Foundry.Layouts.Item(HeaderClass).ClassName

Print .Division.Foundry.Layouts.Item(HeaderClass).MarginLeft
Print .Division.Foundry.Layouts.Item(HeaderClass).MarginTop
Print .Division.Foundry.Layouts.Item(HeaderClass).Background.Color.GetRGB()
Print .Division.Foundry.Layouts.Item(HeaderClass).Content.GetText($LwpGetObjectTypeInfoPar
agraph, False)
```



'Example: FindExactCase property

'This example script has not yet been created.

'Example: FindFont property

'This example script has not yet been created.

'Example: FindForwardDirection property

'This example script has not yet been created.

'Example: FindParent method

'This example script has not yet been created.

```
'Example: FindString property
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script finds the first 'cat' then replaces it with
' 'dog'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace

.Application.FindAndReplace.FindString = "cat"

.Application.FindAndReplace.ReplaceString = "dog"
.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find
.ReplaceCmd
```

'Example: FindStyleName property

'This example script has not yet been created.

```
'Example: FindTerm method
' This example opens the glossary file named 'GLOSSARY.GLS', searches for the
' glossary item named 'TestItem'. If the item is found, its glossary value is
' printed to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = False
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = True
.GlossaryOpen "GLOSSARY.GLS", "Lotus Word Pro"
.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = True
.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = False
Forall Gloss In .Division.Foundry.Glossarys
    GlossItem$ = "TestItem"
    Status% = Gloss.FindTerm(GlossItem$)
    If Status% = 1 Then
        GlossText$ = Gloss.ExtractText(GlossItem$)           Print GlossText$
        Exit Forall
    End If
End Forall
.Documents("GLOSSARY.GLS").Close
```

## **Word Pro: DropCapContainer class**

This language element is not yet defined.

### **Base Classes**

unkown

### **Derived Classes**

None

### **Contained by**

None

### **Usage**



## **Word Pro: DropCapLayoutCollection class**

This language element is not yet defined.

### **Base Classes**

unkown

### **Derived Classes**

None

### **Contained by**

None

### **Usage**

**Word Pro: DropCapLayout class**

The drop cap layout for a DropCapContainer object.

**Base Classes**

BaseObject\Layout\FrameLayout

**Derived Classes****Contained by**

DropCapContainer in the Layout Property

**Usage**

## **Word Pro: RowGroupLayout class**

This language element is not yet defined.

### **Base Classes**

unkown

### **Derived Classes**

None

### **Contained by**

None

### **Usage**

## **Word Pro: SmartCorrectCollection class**

This language element is not yet defined.

### **Base Classes**

unkown

### **Derived Classes**

None

### **Contained by**

None

### **Usage**

## **Word Pro: SmartFillCollection class**

This language element is not yet defined.

### **Base Classes**

unkown

### **Derived Classes**

None

### **Contained by**

None

### **Usage**

**Word Pro: SmartFill class**

This language element is not yet defined.

**Base Classes**

unkown

**Derived Classes**

None

**Contained by**

None

**Usage**

## **Word Pro: ExpandOutlineLevel method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPANDOUTLINELEVEL\_METHOD\_EXSCRIPT',1)} [See example](#)

Expands the highest level contracted heading(s) which are subordinate to the paragraph for which you are calling the method. For example, when you call this method for a Level 1 heading, it will expand the highest level contracted heading(s) which are subordinate to that Level 1 heading.

### **Syntax**

[objectreference].ExpandOutlineLevel([IsExpandAllLevels])

### **Parameters**

*IsExpandAllLevels*

Allows you to expand all the subordinate headings under the heading from which you call this method. Data type is Integer but the legal values for this parameter are -1 and 0. You may use the LotusScript constants True (-1) and False (0). A value of True will cause all subordinate headings to be expanded, regardless of their level. When called from WPAApplication, this parameter is optional. Default is True.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: ExportAllAsNotesFX method**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPORTALLASNOTESFX\_METHOD\_EXSCRIPT',1)} [See example](#)

Exports all DocInfo fields in a document as Notes/FX.

**Syntax**

[objectreference].ExportAllAsNotesFX(Export)

**Parameters***Export*

An Integer value indicating whether or not to export all DocInfo fields in a document to Notes. You can use the LotusScript constants of True (-1) and False (0) for this parameter.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## Word Pro: ExportAsNotesFX method

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXPORTASNOTESFX\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets or resets a DocInfo field as Notes/FX.

### Syntax

[objectreference].ExportAsNotesFX(Type, Export)

### Parameters

*prexType*

You can select and set any one of the DocInfo field data types below to export as Notes FX. The value of this Variant parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpDocVarAllversionnames (210)	The names of all the versions of a document.
\$LwpDocVarCreatedby (2046)	The Word Pro user name of the person who first saved the document.
\$LwpDocVarDatecreated (196)	The date and time the document was first created.
\$LwpDocVarDatelastrevision (197)	The last time the document was opened, edited, and saved.
\$LwpDocVarDescription (195)	Displays a description of the document.
\$LwpDocVarDivisionname (203)	The name of a division within the document.
\$LwpDocVarDoccategory (212)	The assigned category of the division or document.
\$LwpDocVarDocsize (202)	The size of the document in kilobytes.
\$LwpDocVarField (191)	Specifies the DocInfo field you want to export as Notes FX.
\$LwpDocVarFilename (192)	The file name you specified in the Save As dialog box when you saved the document.
\$LwpDocVarKeywords (215)	Displays the assigned keywords for the document.
\$LwpDocVarLasteditor (2047)	The initials of the editor who last saved the document.
\$LwpDocVarNone (190)	No DocInfo fields are set or reset as Notes FX.
\$LwpDocVarNumchars (201)	The number of characters in the document.
\$LwpDocVarNumpages (199)	The number of pages in the document.
\$LwpDocVarNumversions (209)	The number of versions in the document.
\$LwpDocVarNumwords (200)	The number of words in the document.
\$LwpDocVarOthereditors (2048)	The Word Pro user names of any other people who saved the document.
\$LwpDocVarOtherversioneditors (207)	The Word Pro user names of any other people who saved a version of the document.
\$LwpDocVarPath (193)	The drive and folder where the document is located.
\$LwpDocVarSectionname (204)	The name of a section in the document.
\$LwpDocVarStylesheet (194)	The style sheet used in the document.
\$LwpDocVarTotaledittime (198)	The total number of minutes the document was open.
\$LwpDocVarVersioncreatedate (206)	The date and time the version was first created.
\$LwpDocVarVersioncreatedby (205)	The Word Pro user name of the person who first saved the version of the document.
\$LwpDocVarVersionlasteditdate (213)	The date and time that the version was last saved.

\$LwpDocVarVersionlasteditedby (214)	The initials of the editor who last saved the version.
\$LwpDocVarVersionname (208)	The name of the current version.
\$LwpDocVarVersionnumrevisions (2049)	The number of times the version was opened, edited, and saved.
\$LwpDocVarVersionremarks (211)	Any remarks written by an editor to be reviewed by any other assigned editors.

#### *prexExport*

Indicates whether or not to export a DocInfo field to Notes. Data type is Integer.

#### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

#### **Usage**

Setting this method to True instructs Word Pro to export a DocInfo field to Notes. Setting this method to False instructs Word Pro not to export a DocInfo field to Notes. Equivalent to choosing File - Document Properties, choosing Document, clicking the Fields panel and selecting "Export as Notes/Fx field data."

## **Word Pro: ExtractText method**

{button ,AL('H\_GLOSSARY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_EXTRACTTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Extracts the text in the meaning of a term and changes that text to a null terminated string.

### **Syntax**

[objectreference].PowerField.ExtractText()

[objectreference].Glossary.ExtractText(Term)

### **Parameters**

[Glossary]

*Term*

Specifies the string term from which you extract text.

### **Return value**

String

### **Usage**

## **Word Pro: FastFormat method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FASTFORMAT\_METHOD\_EXSCRIPT',1)} [See example](#)

Turns on the FastFormat feature, using either the local attributes found at the insertion point or the paragraph style attributes found in the focus.

### **Syntax**

[objectreference].FastFormat(IsUseStyle)

### **Parameters**

*IsUseStyle*

A Numeric expression which allows you to use the style attributes or the local text attributes for the Fast Format feature. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: FindCellLayout method**

{button ,AL('H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLU  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDCELLLAYOUT\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to determine whether or not there is a layout object within the specified cell.

### **Syntax**

[objectreference].FindCellLayout(Row, Column)

### **Parameters**

#### *Row*

Data type is Integer. The row ID is zero based, which means that the first row in a table has a row ID value of zero.

#### *Column*

Data type is Integer. The column ID is zero based, which means that the first column in a table has a column ID value of zero.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Cell layout objects are sparse, which means that they are only created when Word Pro needs to use them. This is done to conserve system memory.

## Word Pro: FindClass method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_FINDCLASS_METHOD_EXSCRIPT',1)} See example
```

Finds the first child of a layout object of the class that you specify.

## Syntax

```
[objectreference].FindClass(LayoutClassName)
```

## Parameters

*LayoutClassName*

The name of the class that contains the layout you want to find. The name of the class must be one of the String values listed below:

Frame  
DropCap  
SuperTable  
Viewport  
Footer  
TableHeading  
Footnote  
Endnote  
NoteFrame  
NoteHeader  
NoteText  
Cell  
FnCell  
FnContinueOn  
FnContinueFrom  
Page  
Table  
Connected  
Hidden  
Row  
Header  
GroupFrame

## Return value

Returns a string value representing the name of the layout that matches the class you want to find.

## Usage

The LayoutClass name parameter is case sensitive.

## Word Pro: FindParent method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCO  
NTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_FINDPARENT_METHOD_EXSCRIPT',1)} See example
```

Returns the name of the parent container's layout.

## Syntax

```
[objectreference].FindParent(ParentType)
```

## Parameters

### *ParentType*

Indicates whether the name of the parent container layout you want to return is a cell or frame container. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpParentTypeCell (1630)	Indicates whether the name of the parent container layout you want to return is a cell container.
\$LwpParentTypeFrame (1629)	Indicates whether the name of the parent container layout you want to return is a frame container.

## Return value

A string value representing the name of the parent container's layout.

## Usage

## **Word Pro: FindTerm method**

{button ,AL('H\_GLOSSARY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FINDTERM\_METHOD\_EXSCRIPT',1)} [See example](#)

Instructs Word Pro to find the term to be used as a glossary entry.

### **Syntax**

[objectreference].FindTerm(Term)

### **Parameters**

*Term*

Specifies the String term to be found and used as a glossary entry.

### **Return value**

String.

### **Usage**



## Word Pro: Find method

{button ,AL('H\_BOOKMARKMANAGER\_CLASS;H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASSES;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FIND\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is defined in the following classes:

[WPApplication]

Executes a Find based on the values set in the FindAndReplace object. No parameters are required when calling this method from WPApplication.

[BookmarkManager]

Finds a specific bookmark by looking for the user-defined bookmark name or the internal (hexadecimal) marker name.

[ClickHere]

[Text]

Conducts a Find based on the current Find and Replace options. Equivalent to clicking the Find button in the Find and Replace bar.

[TextMarker]

## Syntax

[objectreference].WPApplication.Find()

[objectreference].BookmarkManager.Find(FindName, Name)

[objectreference].ClickHere.Find([MarkerName,] [ReplaceAll,] [ UseTempOptions])

[objectreference].Text.Find([MarkerName,] [ReplaceAll,] [ UseTempOptions])

[objectreference].TextMarker.Find([MarkerName,] [ReplaceAll,] [ UseTempOptions])

## Parameters

### *FindName*

Indicates whether you want to find a bookmark by its user-defined name or by the internal (hexadecimal) marker name. The value of this parameter must be one of the strings or codes listed below:

\$LwpBookmarkFindNameBookmark (31) The user defined name of the bookmark.

\$LwpBookmarkFindNameMarker (32) The internal (hexadecimal) name of the bookmark.

\$LwpBookmarkFindNameBookmark (2054) The user defined name of the bookmark.

\$LwpBookmarkFindNameMarker (2055) The internal (hexadecimal) name of the bookmark.

### *Name*

Data type is String. Contains the name. A bookmark parameter must have a Name or MarkerName, depending on the first parameter set. If the FindName parameter is \$LwpBookmarkFindNameBookmark, then the string should be the user-defined name. If the FindName parameter is \$LwpBookmarkFindNameMarker, then the string should be the internal (hexadecimal) name.

### *MarkerName*

Data type is String. Required if no Name parameter. A bookmark parameter must have a Name or MarkerName, depending on the first parameter set.

### *ReplaceAll*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False (0).

### *UseTempOptions*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False (0).

## Return value

BookmarkManager - String

WPApplication and ClickHere - This method returns a value of -1 (True) or 0 (False) indicating that the method

succeeded or failed respectively.

### **Usage**

Text - The Find method in the Text class can be used to initiate a Find, based on the current parameters in the FindAndReplace class off Application. You will generally use this method directly off the Application class where the FindAndReplace class is located.

**Note** Do not execute this method off a text object without parameters and without first setting the parameters in the FindAndReplace class for the current Application object. If you do, it can cause unreliable results.

## **Word Pro: ForceDocToLoad method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORCEDOCTOLOAD\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].ForceDocToLoad()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: FormatCheckAction method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORMATCHECKACTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to specify a particular FormatCheck option.

### Syntax

[objectreference].FormatCheckAction([Action])

### Parameters

#### *Action*

Data type is Variant. Optional parameter. Default is "\$LwpFormatCheckActionUpdateoptions". The value of this parameter must be one of the strings below or its code equivalent.

\$LwpFormatCheckActionRestart (325)

\$LwpFormatCheckActionUpdateoptions (326)

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: FormatCheckReplace method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FORMATCHECKREPLACE\_METHOD\_EXSCRIPT',1)} [See example](#)

Instructs the FormatCheck tool to replace the selected text with the suggested correction. The FormatCheck tool then moves on to the next error and repeats the process.

### **Syntax**

[objectreference].FormatCheckReplace(Automatic,[ReplaceAllOfRule])

### **Parameters**

#### *Automatic*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False (0).

#### *ReplaceAllOfRule*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False (0).

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: FormatCheck method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_FORMATCHECK\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts the FormatCheck tool. Equivalent to choosing Edit - Check Format.

**Syntax**

[objectreference].FormatCheck(Level)

**Parameters**

*Level*

Data type is Integer.

**Return value****Usage**

## Word Pro: Forward method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CEL  
LAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CL  
ASS;H_DROPCAPCONTAINER_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOO  
TERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_FRAMEGROUPLAYO  
UT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_  
CLASS;H_NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELAYO  
UT_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGROUPLAYOUT_C  
LASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYLAYOUT_CLASS;H_SUBPAGECONTAI  
NER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGR  
OUPPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEHEADINGLA  
YOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_C  
LASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_FORWARD_METHOD_EXSCRIPT',1)} See example
```

Moves an object or the insertion point forward. A Backward method is also available.

## Syntax

When called from a Layout object:

```
[objectreference.]Forward()
```

When called from a container object:

```
[objectreference.]Forward(Direction)
```

When called from a Text, TextMarker, or ClickHere object:

```
[objectreference.]Forward(Unit, N[, Cursoring][, TextOnly])
```

## Parameters

### *Direction*

Specifies whether Word Pro should move the insertion point forward by page or by window. Data type for this parameter is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpDirectionPage (182)

Moves the insertion point forward by one page.

\$LwpDirectionWindow (183)

Moves the insertion point forward by one window.

Only used when calling this method from a container object. A container object is any object created from a container class. A container class is any class derived from the BaseContainer class including: CellContainer, DropCapContainer, FrameContainer, NoteContainer, PageContainer, ParallelColsContainer, RowContainer, RubyContainer, SubPageContainer, SuperPageContainer, SuperTableContainer, TableContainer, and TableOnlyCont.

### *Unit*

Specifies the unit of measurement you want to use in moving the insertion point. Use this parameter only when calling this method from a Text, TextMarker, or ClickHere object. You must also use the N parameter to indicate how many of these units to move forward. Data type for this parameter is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpNavigateObjectTypeCharacter (1519)

Moves the insertion point the specified number of characters. Set the Cursoring parameter to True to mimic the use of the arrow keys.

\$LwpNavigateObjectTypeChunk (1522)

Moves the insertion point the specified number of chunks. A chunk is comprised of a single word (a contiguous group of characters with no spaces) and all the spaces following that word. If the insertion point is at the beginning, the end or anywhere within a word, the chunk is comprised of that word and the spaces which follow it. If the insertion point is between two spaces, the chunk is seen as all the spaces following the insertion point to the beginning of the next word. If there is no word between the spaces and the end of the paragraph, the chunk is comprised of all the spaces up to the end of the paragraph.

\$LwpNavigateObjectTypeObject (1520)

Any of the objects defined in this list.

\$LwpNavigateObjectTypePage (1518)

Moves the insertion point the specified number of pages, leaving it at the top of the page.

`$LwpNavigateObjectTypeParagraph` (1524)

Moves the insertion point the specified number of paragraphs. A paragraph is comprised of all the text and tables between two paragraph markers as well as any frames whose "Place frame" option is set to "With paragraph above."

`$LwpNavigateObjectTypeSentence` (1523)

Moves the insertion point the specified number of sentences. A sentence is comprised of all the text between two periods.

`$LwpNavigateObjectTypeWord` (1521)

Moves the insertion point the specified number of words. A word is comprised of a contiguous string of alphanumeric characters. Punctuation or a space is seen as the end of a word. If the insertion point is between two spaces, the word is comprised of all the spaces on both sides of the insertion point as well as the word preceding the spaces.

*N*

An Integer expression which specifies the number of units you want to move the insertion point. Use this parameter only when calling this method from a Text, TextMarker, or ClickHere object.

#### *Cursoring*

Use this parameter only when the Unit parameter has a value of `$LwpNavigateObjectTypeCharacter`. This parameter takes an Integer expression which indicates whether or not you want Word Pro to move the insertion point as if you were using the arrow keys to move the cursor through a document. When you use the arrows keys, Word Pro skips over hidden markers such as bookmarks. The default value for this parameter is False (0) which causes Word Pro to include any hidden markers when it moves the insertion point by characters. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Use this parameter only when calling this method from a Text, TextMarker, or ClickHere object.

#### *TextOnly*

An Integer expression which indicates whether or not you want Word Pro to exclude tables and frames marked as "With paragraph above" when moving the insertion point. The data type for this parameter is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False (0) which includes tables and certain frames. A value of True will cause Word Pro to skip over tables and frames when moving the insertion point. Use this parameter only when calling this method from a Text, TextMarker, or ClickHere object.

#### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

#### **Usage**

What this method moves forward and how is determined, in part or in whole, by the object from which you call the method.

When you call this method from a Layout object, it moves that Layout object forward one level in relation to the other layout objects of the same type.

When you call this method from a container object, Word Pro places the insertion point at the beginning of the next page.

When you call this method from a Text, TextMarker, or ClickHere object, Word Pro moves the insertion point forward the specified number of units.



**Word Pro: FrameRevert method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FRAMEREVERT\_METHOD\_EXSCRIPT',1)} [See example](#)

Reverts the attributes of the currently active frame to the attributes specified in the frame style.

**Syntax**

[objectreference].FrameRevert()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: FXGetNotesString method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FXGETNOTESSTRING\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the string expression representing a FX Notes field.

### **Syntax**

[objectreference].FXGetNotesString(FieldName)

### **Parameters**

*FieldName*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: FXGetNotesWriteHandle method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FXGETNOTESWRITEHANDLE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].FXGetNotesWriteHandle()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: FXSetNotesString method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FXSETNOTESSTRING\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the string expression representing the name and value of a FX Notes field.

### **Syntax**

[objectreference].FXNotesString(FieldName, Value)

### **Parameters**

*FieldName*

Data type is String.

*Value*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: GetActiveList method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETACTIVELIST\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].GetActiveList()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: GetArrayProp method

{button ,AL('H\_SILVERBULLET\_CLASS;H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETARRAYPROP\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the properties of an array. This method is defined in the following classes:

[SilverBullet]

[UserInterfacePrefs]

### Syntax

[objectreference].SilverBullet.GetArrayProp(BulletArrayProp, Level)

[objectreference].UserInterfacePrefs.GetArrayProp(PrefPropScope, Index)

### Parameters

#### *BulletArrayProp*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpBulletArrayPropCumulative (79)

\$LwpBulletArrayPropDivision (81)

\$LwpBulletArrayPropLesser (78)

\$LwpBulletArrayPropLesserspecific (77)

\$LwpBulletArrayPropSection (80)

#### *Level*

Data type is Integer.

#### *PrefPropScope*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpPrefPropScopeFindString (1640). Equivalent to the "Find" box in the Find & Replace bar.

\$LwpPrefPropScopeReplaceString (1641) Equivalent to the "Replace with" box in the Find & Replace bar.

#### *Index*

Data type is Integer. Legal values are 0, 1, 2, and 3.

### Return value

String

### Usage

[SilverBullet]

[UserInterfacePrefs]

Use this method to obtain a value indicated in the Find & Replace bar.

## **Word Pro: GetContents method**

{button ,AL(`H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_GETCONTENTS\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the contents of a marker in a document.

### **Syntax**

[objectreference].GetContents(IncludeMarkers)

### **Parameters**

*IncludeMarkers*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0).

### **Return value**

### **Usage**

## Word Pro: GetCopyFormatCategories method

{button ,AL(^H\_CLICKHERE\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GETCOPYFORMATCATEGORIES\_METHOD\_EXSCRIPT',1)} [See example](#)

Looks at the selection in a Word Pro document and determines which OLE data formats are present within that selection.

### Syntax

[objectreference].GetCopyFormatCategories()

### Parameters

None

### Return value

The return value for this method is always an Integer representing one or more of the values below:

- 0 - No file types
- 1 - Text in the selection
- 2 - A table or any part of a table
- 4 - A graphic
- 8 - A linked or embedded OLE object

If more than one of these format categories exists within the selection, the return value is the sum of integers for each format in the selection. For example, if both text and a graphic are in the selection, the return value is 5.

### Usage

To make good use of this method's return value, you should have a thorough understanding of OLE data formats, as defined in the OLE2 for Windows specifications.



## Word Pro: GetCount method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETCOUNT\_METHOD\_EXSCRIPT',1)} [See example](#)

Counts the words or characters in the current selection or object.

### Syntax

[objectreference].GetCount(What, Which)

### Parameters

#### *What*

Allows you to choose between the current selection and the object. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is

\$LwpGetCountWhatText.

\$LwpGetCountWhatSelection (2203)

Counts the words or characters in the current selection.

\$LwpGetCountWhatText (2204)

Counts the words or characters in the object from which you call this method.

#### *Which*

Allows you to choose between counting characters and counting words. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is

\$LwpGetCountWhichCharacter.

\$LwpGetCountWhichCharacter (2205)

Counts the characters.

\$LwpGetCountWhichWord (2206)

Counts the words.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

If nothing is selected and you specify \$LwpGetCountWhatSelection as the value of the What parameter, this method counts all the words or characters in the Text, TextMarker, or ClickHere object from which you call this method.

## Word Pro: GetCurrentMarkerName method

```
{button ,AL('H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_GETCURRENTMARKERNAME_METHOD_EXSCRIPT',1)} See example
```

Returns the name of a marker.

### Syntax

```
[objectreference].GetCurrentMarkerName(MarkerType)
```

### Parameters

#### *MarkerType*

Specifies which type of marker Word Pro should look for. If Word Pro find this type of marker in the focus, it returns the name of the marked object. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpMarkerTypeBookmark (590)

Looks for a Bookmark marker.

\$LwpMarkerTypeClickhere (593)

Looks for a Click Here marker.

\$LwpMarkerTypeDde (591)

Looks for a DDE marker.

\$LwpMarkerTypeDefault (589)

\$LwpMarkerTypeField (592)

Looks for a Power Field marker.

### Return value

Returns the name of the marker which has the focus when the method is called. Data type is String.

Bookmarks return the internal bookmark name which can be used to get the external bookmark name.

ClickHere blocks return the name you provide in the Click Here Block Options dialog box.

Power Fields return the internal power field name which can be used to get the external power field name.

### Usage

The focus must include the type of marker you specify in the MarkerType parameter.

The insertion point must be within the text of the Bookmark, ClickHere block, or Power field.

## Word Pro: GetData method

{button ,AL('H\_SCRIPTDATASET\_CLASS;H\_WPDATASET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the value of a variable in a data set.

### Syntax

[objectreference].GetData(DataName,Default)

### Parameters

#### *DataName*

The variable name of which you want to return the value. DataName is a String expression.

#### *Default*

The default value of the variable that you want to return.

### Return value

String

### Usage

Use this method to return the data associated with a variable. When you return the data, you must specify the variable name and default value. If the variable does not exist, it should return the default value.

**Note** In the 16-bit Version of Word Pro, you specify the variable name and a null string default value. If the variable does not exist, it should return the null string default value.

## **Word Pro: GetDefaultPageSize method**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETDEFAULTPAGESIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the default size of a page in the current document.

### **Syntax**

[objectreference].GetDefaultPageSize()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: GetDisplayableFilterName method**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETDISPLAYABLEFILTERNAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the name displayable filters, such as Word Pro or Ami Pro filters. Filter names from other applications may not display in Word Pro.

### **Syntax**

[objectreference].GetDisplayableFilterName(FilterId)

### **Parameters**

*FilterId*

Data type is Integer.

### **Return value**

### **Usage**

## **Word Pro: GetDocDescription method**

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETDOCDESCRIPTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the specified document's description.

### **Syntax**

[objectreference].GetDocDescription(FileExtractType, FullFilePath)

### **Parameters**

#### *FileExtractType*

Data type is Variant. The value of this parameter must be the string below or its code equivalent.  
\$LwpFileExtractTypeFileprotection (259)

#### *FullFilePath*

Stores the path to the desired file. Data type is String.

### **Return value**

Equivalent to the document description entered into the "Description" field on the General panel of the Document Properties dialog box.

### **Usage**

## **Word Pro: GetEnum method**

{button ,AL('H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETENUM\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the numeric equivalent of an enumerated string value.

### **Syntax**

[objectreference].GetEnum(Name)

### **Parameters**

*Name*

The String value for which you need the numeric equivalent.

### **Return value**

Variant

### **Usage**

In many properties and methods, Word Pro provides an enumerated list of legal values. Each value can be expressed as either a string or a number while working within LotusScript. However, if you need to use one of the values through another programming language such as Visual Basic, you must use the numeric value.

This method returns the numeric equivalent of an enumerated string value.

## **Word Pro: GetEnvelopeDefaults method**

{button ,AL(^H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GETENVELOPEDEFAULTS\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves the default envelope setting in the current document.

### **Syntax**

[objectreference].GetEnvelopeDefaults()

### **Parameters**

### **Return value**

### **Usage**



## **Word Pro: GetFileDescription method**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETFILEDESCRIPTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves a description of a file.

### **Syntax**

[objectreference].GetFileDescription(FileName)

### **Parameters**

*FileName*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: GetFormula method**

{button ,AL('H\_CELLENGINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETFORMULA\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the formula that resides in a table cell.

### **Syntax**

[objectreference].GetFormula(Row, Column)

### **Parameters**

#### *Row*

Allows you to specify the row from which you want to return a formula. Data type is Integer.

#### *Column*

Allows you to specify the column from which you want to return a formula. Data type is Integer.

### **Return value**

### **Usage**

**Word Pro: GetInternetFile method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETINTERNETFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves the text and HTML code of the specified URL and stores them in a temporary file.

**Syntax**

[objectreference].GetInternetFile(URL)

**Parameters**

*URL*

A String expression which specifies the URL for the Internet file you want to retrieve.

**Return value**

Returns the name of the temporary file in which the URL contents are stored. Data type is String.

**Usage**

Graphics and other external files referenced in the HTML code are not retrieved by this method.

To display the Internet file, the user can use the Web browsing features in Word Pro or the RetrieveInternetFileAndOpen method in LotusScript.

## **Word Pro: GetLastUsedFilter method**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLASTUSEDFILTER\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves the last filter used to import or export the current document.

### **Syntax**

[objectreference].GetLastUsedFilter(FilterType)

### **Parameters**

*FilterType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpFilterTypeGraphic (280)

\$LwpFilterTypeTable (281)

\$LwpFilterTypeText (279)

### **Return value**

### **Usage**

**Word Pro: GetLineMix method**

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLINEMIX\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns whether or not all sides of a table have the same type of line style.

**Syntax**

[objectreference].GetLineMix()

**Parameters****Return value****Usage**

**Word Pro: GetLineStyle method**

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLINESTYLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the borderline, outline, or diagonal line style used in a table in the current document.

**Syntax**

[objectreference].GetLineStyle()

**Parameters****Return value**

Returns the legal values found in the ChgLineStyle method. For information, see the ChgLineStyle method.

**Usage**

## Word Pro: GetLinkDisplayNameFileLength method

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLINKDISPLAYNAMEFILELENGTH\_METHOD\_EXSCRIPT',1)} [See example](#)

### Syntax

[objectreference].GetLinkDisplayNameFileLength(LinkCookie)

### Parameters

*LinkCookie*

Data type is Long.

### Return value

Long

### Usage

**Word Pro: GetLinkName method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLINKNAME\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].GetLinkName(LinkCookie)

**Parameters**

*LinkCookie*

Data type is Long.

**Return value**

String

**Usage**



## **Word Pro: GetLinkSourceName method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETLINKSOURCENAME\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].GetLinkSourceName(LinkCookie)

### **Parameters**

*LinkCookie*

Data type is Long.

### **Return value**

String

### **Usage**

### **Word Pro: GetMarkedText method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GETMARKEDTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves marked text in the current document.

### **Syntax**

[objectreference].GetMarkedText()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: GetMarkerName method

```
{button ,AL('H_BASSETABLE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DDELINKMANAGER_CLASS;H_DROPAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNOTETABLE_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GLOSSARY_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARALLELOLUMNS_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_GETMARKERNAME_METHOD_EXSCRIPT',1)} See example
```

This method is defined in the following classes:

[BaseTable]

Retrieves a list of all the marker names of the specified type associated with a table object layout.

[DdeLinkManager]

Retrieves the name of the Dde marker which you requested.

[Layout]

Retrieves a list of all the marker names of the specified type associated with a layout.

## Syntax

```
[objectreference].DdeLinkManager.GetMarkerName(DdeFind,p2)
```

```
[objectreference].Layout.GetMarkerName(MarkerType)
```

```
[objectreference].BaseTable.GetMarkerName(MarkerType)
```

## Parameters

*DdeFind*

Data type is Variant. The value of this parameter must be one of the strings below or its equivalent.

\$LwpDdeFindLinknameUsingMarker (173) Returns the user name (you have the internal [hexidecimal] name).

\$LwpDdeFindMarkerNameUsingHandle (172) Returns the internal name (you have the conversation handle).

\$LwpDdeFindMarkerUsingLinkInfo (174) Returns the internal name (you have the user name).

*p2*

Data type is Variant.

[Layout] [BaseTable]

*MarkerType*

Data type is Variant which must be one of the strings below or its equivalent.

\$LwpMarkerTypeDefault (589) Returns a positional marker. It is not of any other type and is only used to mark a spot or position.

\$LwpMarkerTypeBookmark (590) Returns all the specified bookmark markers.

\$LwpMarkerTypeDde (591) Returns all the specified Dde markers.

\$LwpMarkerTypeField (592) Returns all the specified field markers.

\$LwpMarkerTypeClickhere (593) Returns all the specified Click Here markers.

\$LwpMarkerTypeRuby (594) Returns all the specified Ruby markers.

## Return value

[DdeLinkManager]

String.

[Layout]

Returns a list of names of all marker objects that are associated with the layout object.

## Usage



## Word Pro: GetMisspelledWord method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETMISPelledWORD\_METHOD\_EXSCRIPT',1)} [See example](#)

Finds the next misspelled word in the current text stream.

### Syntax

[objectreference].GetMisspelledWord([EndMarkerName])

### Parameters

*EndMarkerName*

A String expression which specifies the name of the marked range within which you want to find a misspelled word. Optional parameter. If you provide a name and the insertion point is located within that named range, Word Pro searches the range for the next misspelled word. If the insertion point is not within the named range, Word Pro does nothing.

### Return value

Returns a String representing the next misspelled word in the text stream.

### Usage

Equivalent to choosing Edit - Check Spelling, with the exception that it does not highlight all misspelled words and it does not open the Spell Check bar.

If the insertion point is within or at the end of a misspelled word when you call this method, Word Pro returns that misspelled word. Otherwise, Word Pro moves to, selects, and returns the next misspelled word in the stream.

## Word Pro: GetNamedProperty method

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_CLICK_HERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_GETNAMEDPROPERTY_METHOD_EXSCRIPT',1)} See example
```

Retrieves the value of the named property in the current object.

### Syntax

```
[objectreference].GetNamedProperty(PropertyName)
```

### Parameters

*PropertyName*

A String expression representing the name of the property which you want to retrieve.

### Return value

String.

### Usage

A named property is a user-defined property assigned to an object. Unlike variables, named properties are persistent. They continue to exist when a script stops executing, and when a document is closed and reopened.

This method returns the String value stored in a specific named property. If you refer to a named property which doesn't exist on an object, the GetNamedProperty method returns an empty string. A run-time error does not occur. Use the HasNamedProperty method to determine whether or not a particular named property exists for an object.

## **Word Pro: GetNameFromPage method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETNAMEFROMPAGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the name of the page of the current division.

### **Syntax**

[objectreference].GetNameFromPage(PageNumber,[IsReturnInternalName])

### **Parameters**

*PageNumber*

Data type is Integer.

*IsReturnInternalName*

Data type is Boolean. Optional parameter. Default is False.

### **Return value**

### **Usage**

## Word Pro: GetObjectList method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETOBJECTLIST\_METHOD\_EXSCRIPT',1)} [See example](#)

(ReadOnly) Returns the current container's type. The type can be page, table, frame, cell, row, and so on.

### Syntax

[objectreference].GetObjectList()

### Parameters

### Return value

### Usage



## **Word Pro: GetOne method**

{button ,AL('H\_TABRACK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETONE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].GetOne(StartingIndex, Count)

### **Parameters**

*StartingIndex*

Data type is Integer.

*Count*

Data type is Integer.

### **Return value**

### **Usage**

## Word Pro: GetParaNumber method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETPARAMNUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the number which indicates the count of the current paragraph as it relates to other paragraphs which are in the same position. Position refers to the position of the paragraph as defined by its Outline Sequence level.

### Syntax

[objectreference].GetParaNumber(Position)

### Parameters

*Position*

Data type is Integer.

### Return value

The number of the paragraph.

### Usage

There are nine possible positions available for an outline sequence. You can see these positions displayed in the Set Outline Style Sequence dialog box by choosing Text - Outline - Outline Styles. You can assign one or more paragraph styles to each of these positions.

This method checks the outline style sequence position of the current paragraph. It then returns the total number of paragraphs prior to and including the current paragraph which are at that same position. For example, if the current paragraph is at position 2 in the outline style sequence, and there are five more paragraphs before it which are also at position 2, the GetParaNumber method has a return value of 6. The return value is not affected by any position 2 paragraphs that may appear after the current paragraph.

## Word Pro: GetPasteFormatCategories method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CLICKHERE_CLASS;H_DROPCAPCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_GRAPHIC_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEONLYCONT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_GETPASTEFORMATCATEGORIES_METHOD_EXSCRIPT',1)} See example
```

Looks at the focus in a Word Pro document and determines which OLE data formats can be pasted at the insertion point.

### Syntax

```
[objectreference].GetPasteFormatCategories()
```

### Parameters

None

### Return value

The return value for this method is always an Integer representing one or more of the values below:

- 0 - No file types
- 1 - Text in the selection
- 2 - A table or any part of a table
- 4 - A graphic
- 8 - A linked or embedded OLE object
- 15 - All of the formats listed above

If more than one of these format categories can be pasted in the current context, the return value is the sum of integers for each format that can be pasted. For example, if it is possible to paste all the formats, the return value is 15 (the sum of 1, 2, 4 and 8).

### Usage

To make good use of this method's return value, you should have a thorough understanding of OLE data formats, as defined in the OLE2 for Windows specifications.

## Word Pro: GetPosition method

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GETPOSITION\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves the position of the insertion point relative to the specified Marker object.

### Syntax

[objectreference].GetPosition(MarkerName)

### Parameters

*MarkerName*

A String expression which specifies the name of the Marker object for which you want the position.

### Return value

Returns one of the four integers listed below:

<u>Return Value</u>	<u>Definition</u>
-2	The insertion point is not located in the same text stream as the specified Marker object.
-1	The insertion point is located before the Marker object.
0	The insertion point is located within the Marker object.
1	The insertion point is located after the Marker object.

### Usage

**Word Pro: GetPowerFieldValue method**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETPOWERFIELDVALUE\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the value of the Power field in the current document.

**Syntax**

[objectreference].GetPowerFieldValue(PowerFieldName)

**Parameters**

*PowerFieldName*

Data type is String.

**Return value****Usage**

## **Word Pro: GetPrinterInfo method**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETPRINTERINFO\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays information about the selected printer for the current document.

### **Syntax**

[objectreference].GetPrinterInfo()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: GetProfileString method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETPROFILESTRING\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the specified profile string from the specified .INI file. You can get a profile string from any of the standard Word Pro .INI files or you can specify another .INI file.

### Syntax

[objectreference].GetProfileString(Section, Key[, DefString][, IniFileType][, WhichIniLocation][, IniName])

### Parameters

#### *Section*

A String expression which specifies a name of a section in the .INI. Word Pro searches only the section you name in this parameter. If the named section does not match a section in the specified INI, this method fails. If you use an empty string ("") Word Pro assumes you are searching the LWPUSER.INI file (IniFileType parameter = "\$LwpIniUserPrefs") and looks for the "WordProUser" section. Most .INIs have more than one section. The section name you provide in this parameter must match the section name in the INI exactly.

#### *Key*

A String expression which specifies the key name in the section you are searching.

#### *DefString*

An optional String expression which allows you to return a default string, if Word Pro fails to find the specified key.

#### *IniFileType*

Specifies the .INI from which you want to get the profile string. You can choose one of the standard Word Pro .INI files or choose "\$LwpIniCustomFile" to search another .INI file. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). You do not have to provide a value for this parameter. Default is "\$LwpIniUserPrefs."

\$LwpIniUserPrefs (2101)

Default for this parameter. This is the .INI file used to store Word Pro's user preference information (lwpuser.ini).

\$LwpIniConfigPrefs (2102)

The .INI file used to store Word Pro's configuration preference information.

\$LwpIniEnvelopeAndMerge (2105)

The .INI file used to store Word Pro's envelope and merge information.

\$LwpIniLanguages (2107)

The .INI file used to store some of Word Pro's language information.

\$LwpIniSharedLotusInfo (2103)

The .INI file used to store shared information between Word Pro and other Lotus products.

\$LwpIniSmartcorrect (2106)

The .INI file used to store Word Pro's SmartCorrect information.

\$LwpIniSmartfill (2104)

The .INI file used to store Word Pro's SmartFill lists.

\$LwpIniCustomfile (2100)

Allows you to get a profile string from an .INI file which is not one of the standard Word Pro .INI files. If you use this value, you must use the IniName parameter to specify the name of the .INI file (Windows 3.1 or OS/2) or .INI entry (Windows 95) in which the profile string is located.

#### *WhichIniLocation*

Tells Word Pro whether to look on the network or the local machine for the specified .INI file. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). You do not have to provide a value for this parameter. Default is "\$LwpUserIniLocation."

\$LwpNetworkIniLocation (2171)

Searches directory for network .INI files.

\$LwpUserIniLocation (2172)

Searches directory for user .INI files.

**Note** For Windows 95, in the registry, the user location is HKEY\_CURRENT\_USER or HKEY\_USERS. The network location is HKEY\_LOCAL\_MACHINE. Within either of these locations, the path below this would be: Software\Lotus\WordPro\96.0.

### *IniName*

An optional String expression that identifies which .INI you want to search. Use this parameter only if you used "\$LwpIniCustomFile" as the value of the IniFileType parameter. This .INI must be stored in the same directory as the standard Word Pro .INIs.

**Note** If you are using Windows 3.1 or OS/2, this value is an .INI file name. If you are using Windows 95, this value is an .INI entry as seen in the Windows Registry application (REGEDIT.EXE).

### **Return value**

Returns the profile string. Data type is String.

### **Usage**



## **Word Pro: GetProtectionType method**

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETPROTECTIONTYPE\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the specified document's protection type.

### **Syntax**

[objectreference].GetProtectionType(FileExtractType, FullFilePath)

### **Parameters**

*FileExtractType*

Data type is Variant. The value of this parameter must be "\$LwpFileExtractTypeFileprotection" or its code equivalent (259).

*FullFilePath*

Data type is String.

### **Return value**

\$LwpFileProtectTypeEditors (261)

\$LwpFileProtectTypeNone (260)

\$LwpFileProtectTypeOrigAuthor (262)

\$LwpFileProtectTypePassword (263)

\$LwpFileProtectTypeReserved (264)

### **Usage**

## **Word Pro: GetRGB method**

{button ,AL('H\_COLOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETRGB\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the RGB (red, green, and blue) value of a color.

### **Syntax**

[objectreference].GetRGB()

### **Parameters**

#### **Return value**

Returns the RGB values for a specific object. If the color of a specific object is a predefined Word Pro color, then Word Pro returns the RGB value of that predefined color.

### **Usage**

**Word Pro: GetRightContextMenu method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETRIGHTMOUSEMENU\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].GetRightContextMenu()

**Parameters**

None.

**Return value**

Long

**Usage**

## **Word Pro: GetSource method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSOURCE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].GetSource(LinkCookie)

### **Parameters**

*LinkCookie*

Data type is Long.

### **Return value**

Integer

### **Usage**

### **Word Pro: GetSpellStatus method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSPELLSTATUS\_METHOD\_EXSCRIPT',1)} [See example](#)

Checks the status of the Spell Check utility and forces it to load, if not already loaded.

### **Syntax**

[objectreference].GetSpellStatus([Language])

### **Parameters**

*Language*

Do not use this parameter. Word Pro sets the value automatically.

### **Return value**

Should always return a value of -1 (True). If the return value is 0, there is a problem loading the Spell Check utility.

### **Usage**

## Word Pro: GetSpellUserDictStatus method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSPELLUSERDICTSTATUS\_METHOD\_EXSCRIPT',1)} [See example](#)

Checks to see whether or not there is a user dictionary available for the specified language.

### Syntax

[objectreference].GetSpellUserDictStatus([Language])

### Parameters

#### *Language*

Specifies which dictionary you want to check. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). Default is "\$LwpLanguagesSystem" (434).

- \$LwpLanguagesAfrikaans (474)
- \$LwpLanguagesAlbanian (475)
- \$LwpLanguagesAmerican (442)
- \$LwpLanguagesArabic (476)
- \$LwpLanguagesArabicAlgeria (2111)
- \$LwpLanguagesArabicBahrain (2121)
- \$LwpLanguagesArabicEgypt (2109)
- \$LwpLanguagesArabicIraq (2108)
- \$LwpLanguagesArabicJordan (2117)
- \$LwpLanguagesArabicKuwait (2119)
- \$LwpLanguagesArabicLebanon (2118)
- \$LwpLanguagesArabicLibya (2110)
- \$LwpLanguagesArabicMorocco (2112)
- \$LwpLanguagesArabicOman (2114)
- \$LwpLanguagesArabicQatar (2122)
- \$LwpLanguagesArabicSyria (2116)
- \$LwpLanguagesArabicTunisia (2113)
- \$LwpLanguagesArabicUae (2120)
- \$LwpLanguagesArabicYemen (2115)
- \$LwpLanguagesAustralian (444)
- \$LwpLanguagesBasque (2123)
- \$LwpLanguagesBrazilian (468)
- \$LwpLanguagesBritish (443)
- \$LwpLanguagesBritishmedize (451)
- \$LwpLanguagesBrmedical (449)
- \$LwpLanguagesBulgarian (478)
- \$LwpLanguagesByelorussian (2124)
- \$LwpLanguagesCatalan (436)
- \$LwpLanguagesChineseHongkong (2126)
- \$LwpLanguagesChinesePrchina (2125)
- \$LwpLanguagesChineseSingapore (2127)
- \$LwpLanguagesChineseTraditional (479)
- \$LwpLanguagesCroatian (2128)
- \$LwpLanguagesCroatianCyrillic (2130)
- \$LwpLanguagesCroatianSerbian (2131)
- \$LwpLanguagesCzech (437)

\$LwpLanguagesDanish (438)  
\$LwpLanguagesDutch (439)  
\$LwpLanguagesDutchBelgian (440)  
\$LwpLanguagesEnglishCanadian (445)  
\$LwpLanguagesEnglishCaribbean (2134)  
\$LwpLanguagesEnglishIreland (447)  
\$LwpLanguagesEnglishJamaica (2133)  
\$LwpLanguagesEnglishNewzealand (446)  
\$LwpLanguagesEnglishSafrica (2132)  
\$LwpLanguagesEstonian (2135)  
\$LwpLanguagesFaeroese (2136)  
\$LwpLanguagesFarsi (2137)  
\$LwpLanguagesFinnish (452)  
\$LwpLanguagesFrench (453)  
\$LwpLanguagesFrenchBelgian (454)  
\$LwpLanguagesFrenchCanadian (455)  
\$LwpLanguagesFrenchLuxembourg (2138)  
\$LwpLanguagesFrenchSwiss (456)  
\$LwpLanguagesGerman (457)  
\$LwpLanguagesGermanAustrian (459)  
\$LwpLanguagesGermanLiechtenstein (2140)  
\$LwpLanguagesGermanLuxembourg (2139)  
\$LwpLanguagesGermanSwiss (458)  
\$LwpLanguagesGreek (460)  
\$LwpLanguagesHebrew (483)  
\$LwpLanguagesHungarian (461)  
\$LwpLanguagesIcelandic (484)  
\$LwpLanguagesIndonesian (2141)  
\$LwpLanguagesItalian (462)  
\$LwpLanguagesItalianSwiss (463)  
\$LwpLanguagesJapanese (485)  
\$LwpLanguagesKorean (486)  
\$LwpLanguagesKoreanJohab (2142)  
\$LwpLanguagesLatvian (2143)  
\$LwpLanguagesLithuanian (2144)  
\$LwpLanguagesMedical (448)  
\$LwpLanguagesNorwegian (464)  
\$LwpLanguagesNynorsk (465)  
\$LwpLanguagesPolish (466)  
\$LwpLanguagesPortuguese (467)  
\$LwpLanguagesRhaetoRoman (2145)  
\$LwpLanguagesRomanian (488)  
\$LwpLanguagesRussian (469)  
\$LwpLanguagesRussianio (470)  
\$LwpLanguagesSlovak (492)  
\$LwpLanguagesSlovene (493)  
\$LwpLanguagesSorbian (2146)

\$LwpLanguagesSpanish (471)  
\$LwpLanguagesSpanishArgentina (2155)  
\$LwpLanguagesSpanishBolivia (2160)  
\$LwpLanguagesSpanishChile (2157)  
\$LwpLanguagesSpanishColombia (2153)  
\$LwpLanguagesSpanishCostarica (2149)  
\$LwpLanguagesSpanishDominican (2151)  
\$LwpLanguagesSpanishEcuador (2156)  
\$LwpLanguagesSpanishGuatemala (2148)  
\$LwpLanguagesSpanishMexican (472)  
\$LwpLanguagesSpanishModern (2147)  
\$LwpLanguagesSpanishPanama (2150)  
\$LwpLanguagesSpanishParaguay (2159)  
\$LwpLanguagesSpanishPeru (2154)  
\$LwpLanguagesSpanishUruguay (2158)  
\$LwpLanguagesSpanishVenezuela (2152)  
\$LwpLanguagesSwedish (473)  
\$LwpLanguagesSystem (434)  
\$LwpLanguagesThai (494)  
\$LwpLanguagesTurkish (495)  
\$LwpLanguagesUkrainian (496)  
\$LwpLanguagesUniversal (435)  
\$LwpLanguagesUrdu (497)  
\$LwpLanguagesVoorkeur (441)

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**



## Word Pro: GetStandardButtonId method

{button ,AL('H\_STATUSBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSTANDARDBUTTONID\_METHOD\_EXSCRIPT',1)} [See example](#)

Obtains the ID of a standard Word Pro status bar button.

### Syntax

[objectreference].GetStandardButtonId(ButtonType)

### Parameters

#### *Button Type*

Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default constant.

\$LwpStandButtBoldButton (1812) A value that specifies the ID of a bold button.

\$LwpStandButtCollapsibleButton (1816) A value that specifies the ID of a collapsible button.

\$LwpStandButtColorButton (1811) A value that specifies the ID of a color button.

\$LwpStandButtCustomButton (1823) A value that specifies the ID of a custom button.

\$LwpStandButtFontButton (1809) A value that specifies the ID of a font button.

\$LwpStandButtItalicButton (1813) A value that specifies the ID of an italic button.

\$LwpStandButtPagedownButton (1818) A value that specifies the ID of a page down button.

\$LwpStandButtPagenumberButton (1819) A value that specifies the ID of a page number button.

\$LwpStandButtPageupButton (1817) A value that specifies the ID of a page up button.

\$LwpStandButtPointSizeButton (1810) A value that specifies the ID of a point size button.

\$LwpStandButtSpacer1Button (1820) A value that specifies the ID of the first spacer button.

\$LwpStandButtSpacer2Button (1821) A value that specifies the ID of the second spacer button.

\$LwpStandButtSpacer3Button (1822) A value that specifies the ID of the third spacer button.

\$LwpStandButtStyleButton (1815) A value that specifies the ID of a style button.

\$LwpStandButtUnderlineButton (1814) A value that specifies the ID of an underline button.

### Return value

Long.

### Usage

Use this method to obtain the ID of a standard Word Pro status bar button. For example, when you are creating a new button, you can use this method to identify the standard button, after which the new button should be inserted.

## **Word Pro: GetStorageProtectionType method**

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETSTORAGEPROTECTIONTYPE\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the document protection type for an OLE object.

### **Syntax**

[objectreference].GetStorageProtectionType(FileExtractType, plStorage)

### **Parameters**

*FileExtractType*

Data type is Variant. The value of this parameter must be the string below or its code equivalent.

\$LwpFileExtractTypeFileprotection (259)

*plStorage*

Data type is Long.

### **Return value**

\$LwpFileProtectTypeEditors (261)

\$LwpFileProtectTypeNone (260)

\$LwpFileProtectTypeOrigAuthor (262)

\$LwpFileProtectTypePassword (263)

\$LwpFileProtectTypeReserved (264)

### **Usage**

## Word Pro: GetText method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns text from the specified part of a Text, ClickHere, or TextMarker object.

### Syntax

[objectreference].GetText(GetObjectType, Advance, 0, 0, [EndMarkerName], [AcrossParagraphs])

### Parameters

#### *GetObjectType*

Specifies the type of text object you want to return. Each type of object is described below. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses).

There is no default value.

#### *\$LwpGetObjectTypeChunk (367)*

Returns the text chunk at the insertion point. A chunk is comprised of a single word (a group of characters with no spaces) and all the contiguous spaces following that word. If the insertion point is at the beginning, the end, or anywhere within a word, the chunk is comprised of that word and the spaces which follow it. If the insertion point is between two spaces, the chunk is seen as all the spaces following the insertion point to the beginning of the next word. If there is no word between the spaces and the end of the paragraph, the chunk is comprised of all the spaces up to the end of the paragraph.

#### *\$LwpGetObjectTypeObject (365)*

Returns the specified internal text object located at the insertion point. For more on internal text objects, see [Overview: Word Pro Text Subobjects](#)

#### *\$LwpGetObjectTypeParagraph (369)*

Returns all the text in the current paragraph.

#### *\$LwpGetObjectTypeParatag (371)*

Returns the text used for the TeamConsolidate/TeamReview marker. This marker only appears on paragraphs that are marked as revisions. This option returns the marker text for the paragraph at the insertion point.

#### *\$LwpGetObjectTypeSelection (370)*

Returns the selected text. Use this option in conjunction with the [Select](#) method when you want to return the bullet text on a bulleted paragraph.

#### *\$LwpGetObjectTypeSentence (368)*

Returns the text of the sentence at the insertion point.

#### *\$LwpGetObjectTypeWord (366)*

Returns the text of the word at the insertion point.

#### *Advance*

Specifies whether or not Word Pro should move the insertion point to the next instance of the type of object specified in the GetObjectType parameter. For example, if you set the value of this parameter to True, Word Pro will find the first instance of the object type you specify, return the text from that object, and then move the insertion point to the next instance of that object type. This parameter is ignored if you use \$LwpGetObjectTypeSelection (370), or \$LwpGetObjectTypeParatag (371) as the value for the GetObjectType parameter. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter.

#### *R1*

Not used. Default is 0.

#### *R2*

Not used. Default is 0.

#### *EndMarkerName*

A String expression which specifies the name of the marker at which you want to end the GetText operation. For example, if you have a marked range of text and the marker name is "MyRange," you could use "MyRange" as the value for this parameter and Word Pro would stop the execution of this method, if it encountered the end of the marked range. Optional parameter.

#### *AcrossParagraphs*

Allows you to get all the text in a selection instead of stopping at the first paragraph. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False (0).

**Return value**

String.

**Usage**

## **Word Pro: GetTOCProperties method**

{button ,AL('H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETTOCPROPERTIES\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the properties of a table of contents.

### **Syntax**

[objectreference].GetTOCProperties(TOCScope, Index)

### **Parameters**

*TOCScope*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTOCScopeDestPageStyle (1853)

\$LwpTOCScopeDestStyle (1854)

\$LwpTOCScopeLevelNumber (1851)

\$LwpTOCScopeRtAlignPgNum (1850)

\$LwpTOCScopeSearchStyle (1852)

\$LwpTOCScopeTypeOfLeader (1846)

\$LwpTOCScopeUseLeader (1847)

\$LwpTOCScopeUsePageNumber (1849)

\$LwpTOCScopeUseText (1848)

*Index*

Data type is Integer.

### **Return value**

### **Usage**

**Word Pro: GetUniqueName method**

{button ,AL('H\_BOOKMARKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETUNIQUENAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns a unique name for a bookmark.

**Syntax**

[objectreference].GetUniqueName()

**Parameters**

None.

**Return value**

String.

**Usage**

Returns a unique name for a bookmark.

**Word Pro: GetUserClassNameFull method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETUSERCLASSNAMEFULL\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].GetUserClassNameFull(LinkCookie)

**Parameters**

*LinkCookie*

Data type is Long.

**Return value**

String.

**Usage**

**Word Pro: GetUserClassNameShort method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETUSERCLASSNAMESHORT\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].GetUserClassNameShort(LinkCookie)

**Parameters**

*LinkCookie*

Data type is Long.

**Return value**

String.

**Usage**



## **Word Pro: GetWordMisspelled method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GETWORDMISPELLED\_METHOD\_EXSCRIPT',1)} [See example](#)

Finds misspelled words in the current document. Equivalent to choosing Edit - Check Spelling.

### **Syntax**

[objectreference].GetWordMisspelled()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: Get method**

{button ,AL('H\_SMARTCORRECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GET\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns the string expression representing an entry in the SmartCorrect tool.

### **Syntax**

[objectreference].Get(Entry)

### **Parameters**

*Entry*

Data type is String.

### **Return value**

### **Usage**

**Word Pro: Glossarize method**

{button ,AL('H\_GLOSSARY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a term to a glossary.

**Syntax**

[objectreference].Glossarize(Term)

**Parameters**

*Term*

Specifies the String term you want to add to the glossary.

**Return value**

String.

**Usage**

## **Word Pro: GlossaryInsert method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYINSERT\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a glossary entry at the insertion point in the currently active document. Equivalent to choosing Edit - Glossary and clicking Insert in the Glossary dialog box.

### **Syntax**

[objectreference].GlossaryInsert(GlossFilePath, Key)

### **Parameters**

*GlossFilePath*

A String expression specifying the path and name of the Glossary file in which the glossary entry is located.

*Key*

A String expression specifying the abbreviation for the glossary entry you want to insert into the document.

### **Return value**

None.

### **Usage**

## Word Pro: GlossaryOpen method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYOPEN\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens a Glossary file for use with the currently active Word Pro document. You may use only one Glossary file for a document at a time.

### Syntax

[objectreference].GlossaryOpen([FilePath,] [FileType,] [Password,] [AddToLastFileOpenList,] [Restore])

### Parameters

#### *FilePath*

A String expression specifying the path and name of the Glossary file you want to open. Optional parameter.

#### *FileType*

A String expression indicating the file type of the file you want to use as the glossary. Word Pro automatically recognizes and imports many file types. Optional parameter. Use this parameter only if the file specified in the Path parameter is not one of these file types:

DCA/RFT	Lotus Manuscript 2.x	MS Word for Windows 1.0
DIF	Lotus Organizer 1.x	MS Word for Windows 2.0
DisplayWrite	Lotus Word Pro	MS Word for Windows 6.0
HTML	Lotus Word Pro SmartMaster	MS Word for Windows95 7.0
Lotus 1-2-3	MS Excel	MS WordPad 1.0
Lotus 1-2-3 for OS/2	MS Excel 3.0	OfficeWriter 4,5,6
Lotus 1-2-3 R3	MS Excel 4.0	Rich Text Format(RTF)
Lotus 1-2-3 R4,5	MS Excel 5.0	SAMNA Word
Lotus 1-2-3 R6	MS Excel 7.0	WordPerfect 5.0
Lotus Ami Pro	MS Windows Write 3.x	WordPerfect 5.1
Lotus Ami Pro 3.x Macro	MS Word for DOS 3,4,5,6	WordPerfect 6.x
Lotus Ami Pro 3.x Styles	MS Word for OS/2	WordStar 2000 R3

#### *Password*

Use this parameter to provide a password if the Glossary file is password protected. Data type is String. Optional parameter.

#### *AddToLastFileOpenList*

This parameter allows you to show or hide the Glossary file from the last file opened list. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False which hides the Glossary file from the list.

#### *Restore*

This parameter is not implemented.

### Return value

None.

### Usage

## Word Pro: GoToBookmark method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOBOOKMARK\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to a specified bookmark. If more than one bookmark exists with the name you provide, Word Pro moves the insertion point to the first bookmark it encounters with that name.

### Syntax

[objectreference].GoToBookmark(Name)

### Parameters

*Name*

A String expression specifying the name of the bookmark. You can indicate the name of the division that contains the bookmark by using the following statement: *divisionname!bookmarkname*

### Return value

A Boolean value of type Integer which indicates success (True) or failure (False). The constants True and False are returned as -1 and 0 respectively.

### Usage

## Word Pro: GoToContainer method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCO  
NTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOCONTAINER\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to the container within the current context of the document.

### Syntax

[objectreference].GoToContainer(GoToLocation, [p2])

### Parameters

#### *GoToLocation*

The value of this Variant parameter moves the insertion point to any one of the values below and must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpGoToLocationPage (372)	The method will move the insertion point to a specific page container within the current context of the document.
\$LwpGoToLocationHeader (373)	The method will move the insertion point to a specific header container within the current context of the document.
\$LwpGoToLocationFooter (374)	The method will move the insertion point to a specific footer container within the current context of the document.
\$LwpGoToLocationFrame (375)	The method will move the insertion point to a specific frame container within the current context of the document.
\$LwpGoToLocationDivision (376)	The method will move the insertion point to a specific division container within the current context of the document.
\$LwpGoToLocationExit (377)	The method will move the insertion point out of a specific container within the current context of the document.

#### *p2*

This optional Variant parameter specifies the string name of a page container or a division container.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: GoToLayout method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_GOTOLAYOUT_METHOD_EXSCRIPT',1)} See example
```

Moves the insertion point to a specific layout object.

### Syntax

When called from WPAApplication:

```
[objectreference].GoToLayout(Name)
```

When called from any other object:

```
[objectreference].GoToLayout()
```

### Parameters

#### *Name*

Only used when you call this method from the WPAApplication object. This parameter provides the name of the layout object to which you want to move the insertion point. This name must be the same as that found in the Name property of the layout object. Data type is String.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

This method is available on most objects which are created from the Layout class or one of its child classes. When you call this method from one of these objects, Word Pro moves the insertion point into the object from which you called the method.

When you call this method from the WPAApplication object, you must identify the layout you want to go to by providing the layout's name in the Name parameter.



**Word Pro: GotoNextParallelColumn method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GOTONEXTPARALLELCOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to the beginning of the parallel column to the right of the current parallel column. If the insertion point is already in the last parallel column, nothing happens. Equivalent to choosing Columns - Goto Next Column Block.

**Syntax**

[objectreference].GotoNextParallelColumn()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## Word Pro: GoToObject method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOOBJECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to the next object of the type you specify.

### Syntax

[objectreference].GoToObject(ObjectName, Forward)

### Parameters

#### *ObjectName*

A String expression which specifies the type of the object to which you want to go. The legal values for this parameter are listed below:

Align	HardSpace	PageNumber
AnchoredFrame	Indent	PowerField
BookMark	Index	RulerMark
ColumnBreak	Layout	SoftHyphen
DDE	LineBreak	Spacing
DocVariable	Note	Table
FootnoteMark	PageBreak	TOC

#### *Forward*

A Numeric expression which allows you to specify whether you want to go to an object behind or in front of the insertion point. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is True.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## **Word Pro: GoToPage method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GOTOPAGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to the specified page number.

### **Syntax**

[objectreference].GoToPage(PageNumber)

### **Parameters**

*PageNumber*

An Integer representing the page number.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: GoToSection method**

{button ,AL(^H\_INDEXSECTION\_CLASS;H\_SECTION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GOTOSECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to a specified section of a division.

### **Syntax**

[objectreference].GoToSection([SectionGoTo])

### **Parameters**

*SectionGoTo*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpSectionGoToBeforeSectionMarker (1746)

\$LwpSectionGoToEndOfSection (1747)

### **Return value**

### **Usage**

## Word Pro: GoToTableCell method

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_GOTOTABLECELL\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to a specific location within the table.

### Syntax

[objectreference].GoToTableCell(CellObjectType, [Next])

### Parameters

*CellObjectType*

Indicates the specific cell to which you want to move the insertion point. Data type is Variant. The value of this parameter must be the string below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpCellObjectType (117)	The CellObjectType parameter must be this value.

*Next*

Indicates whether the focus should be moved to the next location within the table. Data type is Integer. The legal values are -1 (True) or 0 (False). Optional parameter. Default is True. If this parameter is set to False, the method will do nothing and return a value of 0.

### Return value

This method returns an Integer value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Assuming the Next parameter is set to True, the following table describes possible initial insertion point states and what the result of the GoToTableCell method will be in that situation. If the Next parameter is set to False, the GoToTableCell method will do nothing and return a value of 0.

<u>Cursor located in the last cell of a row</u>	<u>Cursor at the end of the text within the cell</u>	<u>Cursor located in last cell of the table.</u>	<u>Result of the GoToTableCell method</u>
No	Doesn't matter	Doesn't matter	Cursor moves to the next cell.
Yes	No	Doesn't matter	Cursor moves to the end of the text within the cell.
Yes	Yes	No	Cursor moves to the next cell.
Yes	Yes	Yes	New row is appended to the table, and the cursor moves to the first cell of the new row.

## Word Pro: GoTo method

{button ,AL(`H\_CLICKHERE\_CLASS;H\_FOOTNOTE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_R  
UBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_GOTO\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point to the specified footnote object or marker object.

### Syntax

[objectreference].GoTo(SelectAll)

[objectreference].GoTo()

[objectreference].GoTo()

### Parameters

*SelectAll*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0).

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: GroupDivision method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GROUPDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a parent division for the division specified in the DivisionName parameter. Equivalent to clicking the right mouse button on a division tab and selecting Group Tabs.

### **Syntax**

[objectreference].GroupDivision([DivisionName])

### **Parameters**

*DivisionName*

A String expression which specifies the internal name of the division you want to place in the new parent division. If you do not provide a value for this parameter, the currently active division will be placed in the new parent division.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

For more information on division names in LotusScript, see [Overview: Division names in LotusScript](#)

## **Word Pro: HandsOffStorage method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HANDSOFFSTORAGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Causes Word Pro to release all references to an embedded Word Pro object's IStorage.

### **Syntax**

[objectreference].HandsOffStorage()

### **Parameters**

None.

### **Return value**

None.

### **Usage**

This method is typically used internally by Word Pro during an OLE operation. You may need this method if you write a script in which you access a Word Pro document that is stored as an OLE object in another application's document (such as a Word Pro document stored in a 1-2-3 worksheet). In such a script, the HandsOffStorage method instructs Word Pro to release its connection to the IStorage in which the Word Pro OLE object is stored.



## Word Pro: HasNamedProperty method

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_RBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_HASNAMEDPROPERTY_METHOD_EXSCRIPT',1)} See example
```

Checks for the presence of a named property within the object from which this method is called.

### Syntax

```
[objectreference.]HasNamedProperty(PropertyName)
```

### Parameters

*PropertyName*

The name of the property that you want to check for within an object. Data type is String.

### Return value

The return value for this method will always be 1 or 0. A return value of 1 indicates that the specified object does have a named property that matches the PropertyName parameter.

### Usage

A named property is a user-defined property assigned to an object. Unlike variables, named properties are persistent. They continue to exist when a script stops executing, and when a document is closed and reopened.

Use this method to determine whether or not a specific named property exists on an object. For example, if you call this method from a ClickHere object, Word Pro checks that ClickHere object for the named property specified in the PropertyName parameter. Use the GetNamedProperty method to actually retrieve the value stored in a named property.

## **Word Pro: Help method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HELP\_METHOD\_EXSCRIPT',1)} [See example](#)

Launches the specified Help file.

### **Syntax**

[objectreference].Help ([HelpFile][, HelpContextID][, HelpString][, Parent])

### **Parameters**

#### *HelpFile*

A String expression which specifies the name of the Help file you want to open. Optional parameter. If you do not provide a value for this parameter, Word Pro will open the Help file which is appropriate in the current context.

#### *HelpContextID*

A Numeric expression which allows you specify which topic you want to turn to, by its resource code or Help panel ID number.

#### *HelpString*

A String expression which allows you specify which topic you want to turn to by the topic's unique Context ID string. For example, the Context ID for this topic is "H\_HELP\_METHOD\_MEMDEF."

#### *Parent*

A Numeric expression of type Long which allows you to specify the parent. Required for use with an OS/2 Help file.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: HideCaretAndSelection method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_HIDECARETANDSELECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].HideCaretAndSelection()

### **Parameters**

Intbool HideCaret = True.

Intbool HideSelection = True.

Intbool;Exposed;Text

### **Return value**

### **Usage**

**Word Pro: HideIconBar method**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIDEICONBAR\_METHOD\_EXSCRIPT',1)} [See example](#)

Hides an icon bar set temporarily until the next context change. Equivalent to hiding a bar from the drop down menu that appears when you click on the Close box.

**Syntax**

[objectreference].HideIconBar()

**Parameters**

Data type is Integer. The legal values for this parameter will always be -1 or 0 but you may use the LotusScript constants of True (-1) and False (0).

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

To hide an icon bar object permanently, use the ShowInContext property.

## Word Pro: Hide method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DOCWINDOW_CLASS;H_DROPCAP  
CONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS  
;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUB  
YCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTAB  
LECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEONLYCONT_CLASS;H_WPAPPLICATION_CL  
ASS',0)} See list of classes
```

```
{button ,AL('H_HIDE_METHOD_EXSCRIPT',1)} See example
```

Reduces a specific note layout object to an icon.

## Syntax

```
[objectreference].Hide()
```

## Parameters

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This method applies only to note container objects.

## How do I create a document with divisions?

Creating documents and divisions within new documents is straight forward in Word Pro. The document and division objects support the Create method with numerous options. The following example illustrates how to create a named document and a named division within that new document.

The following sub, named CreateDocAndDiv, calls two worker subs and specifies names for the new document and division.

```
Sub CreateDocAndDiv
    ' Call the CreateDoc sub to create a document named "WEEK6.LWP".
    CreateDoc "WEEK6.LWP"

    ' Call the CreateDiv sub to create a division named "MondaySales"
    ' in the new document.
    CreateDiv "MondaySales"
End Sub
```

The CreateDoc sub takes the name of the new document as a parameter and passes it to the NewDocument method. You can add other parameters to specify a particular file path or SmartMaster.

```
Sub CreateDoc(NewDocName as String)
' * RUNTIME DEPENDENCIES
' *   Files and paths: You must have the SmartMaster LETTER1.MWP in
'   the subdirectory C:\LOTUS\SMARTERS\WORDPRO.

' Create the document with the following options:
'   Document name = the value of parameter NewDocName
'   File path = "C:\DATA\WORDPRO\"
'   SmartMaster = "C:\LOTUS\SMARTERS\WORDPRO\LETTER1.MWP"
.NewDocument NewDocName, _
    "C:\DATA\WORDPRO\", _
    "C:\LOTUS\SMARTERS\WORDPRO\LETTER1.MWP", "", "", ""

' Save the new document.
.Save
End Sub
```

The CreateDiv sub takes the name of the new division as a parameter and passes it to the CreateDivision method. You can add parameters to specify the style of the new division and its relative position in the current sequence of divisions.

```
Sub CreateDiv(NewDivisionName as String)
    ' Declare a path for the Division SmartMaster.
    Dim SMasterFullPath As String
    ' Declare a variable for the new division.
    Dim MyNewDivision As String

    ' Get the file path for the division SmartMaster.
    SMasterFullPath = _
        .ApplicationWindow.UserInterfacePrefs.StylePath _
        & "\DEFAULT.MWP"

    ' Create the division.
    MyNewDivision = _
        .CreateDivision(SMasterFullPath, "", _
            $LWPDivLocInsertAfterCurrentDiv, "", "")

    ' Name the new division. The division tab reflects the new name.
    .Division.DivisionInfo.Name = "NewDivisionName"

End Sub
```

---

{button ,AL('0)} [See related topics](#)

## How do I create and use a custom dialog?

The process of creating and using custom dialog boxes in Word Pro should be familiar if you have used Microsoft Visual Basic or IBM Visual Age:

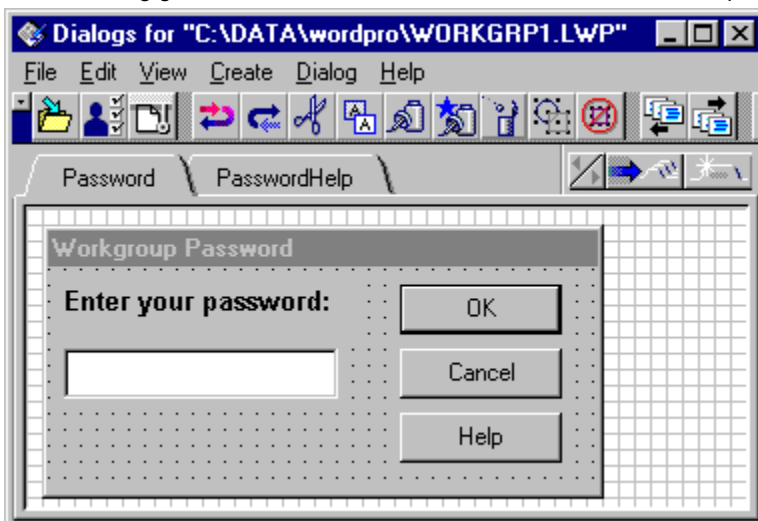
- Create the dialog using the Lotus Dialog Editor.
- Add controls to the dialog using the Lotus Dialog Editor.
- Write scripts for the controls using the Lotus Integrated Development Environment (IDE).
- Run the scripts and display the dialog in Word Pro.

Custom dialogs are stored in Word Pro documents along with your other scripts in your application.

### Creating a custom dialog

To open the Lotus Dialog Editor for your current document, choose Edit - Script & Macros - Show Dialog Editor in Word Pro.

The Dialog Editor provides a tabbed panel for each dialog in your current document. The following example uses two dialogs, one for a user to enter a password and another to display some Help text about the password dialog. The second dialog gets invoked from the first if the user clicks on the Help command button.



Use the InfoBox for the dialog to set properties such as its name, title bar caption, and Help context ID.

To create additional dialogs, choose Create - Dialog in the Dialog Editor.

**Note** Modal and modeless dialogs are not differentiated at design time in the Dialog Editor. You determine whether the dialog is modal or modeless when you call the dialog from your script.

### Adding controls to the dialog

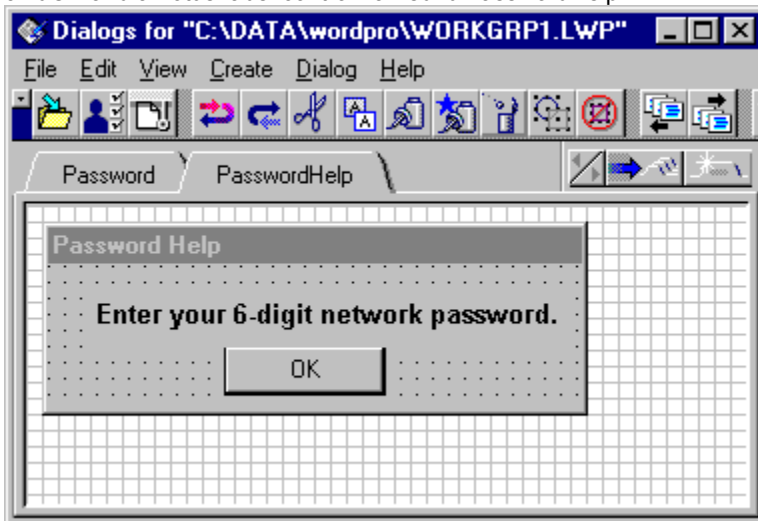
The Dialog Editor is an OLE container that supports OLE Custom (OCX) Controls developed by Lotus and other control vendors. The Dialog Editor provides 13 controls and displays icons for each control in the control toolbox.



To add one of these controls to your dialog, click its icon in the toolbar and size the control in the dialog panel. The



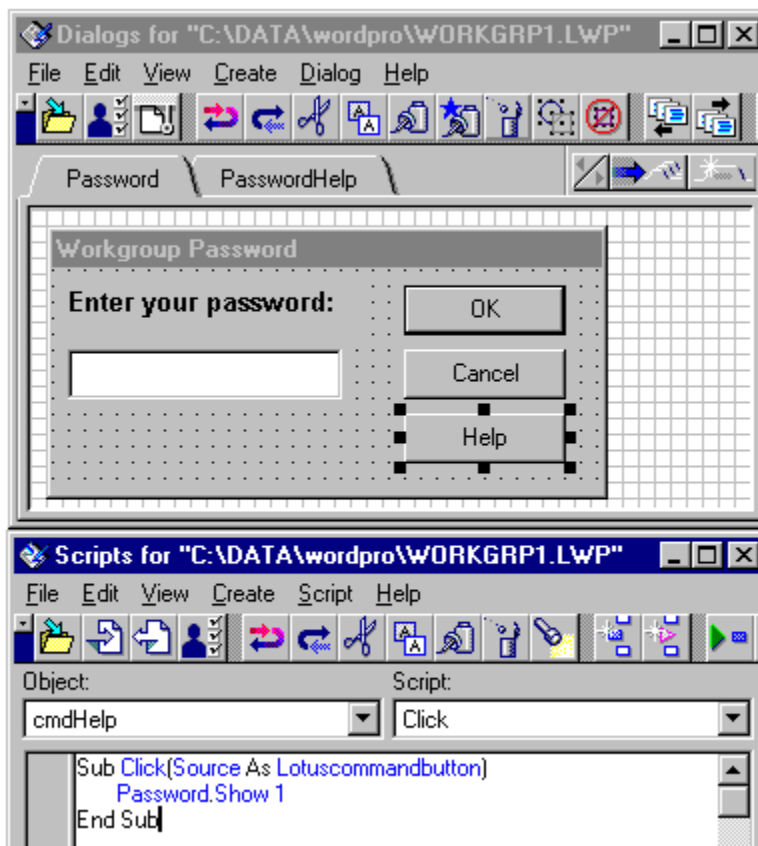
second dialog in our example, named PasswordHelp, has two controls: a LotusCommandButton control named cmdOK and a LotusLabel control named lbIPasswordHelp.



Use the InfoBox for each control to determine text colors, fonts, borders, names, captions, and default values in lists. **Tip** To add third party controls to your dialogs, choose Create - Control - More ... in the Dialog Editor and select from among the OLE controls registered on your system.

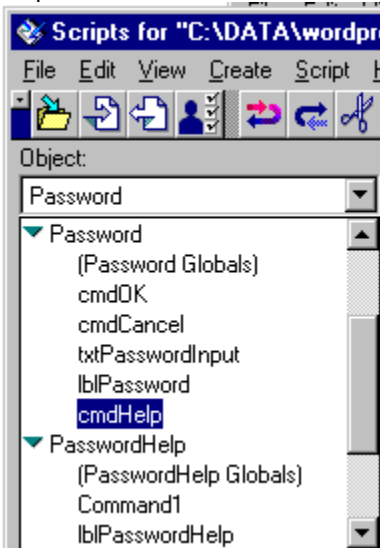
### Writing scripts for controls

The Dialog Editor and the Integrated Development Environment (IDE) are closely integrated. To write scripts for a particular control in your dialog, double-click that control. Word Pro displays the IDE and navigates to the default event procedure for the selected control in your dialog. The following illustration shows the control named cmdHelp selected in the Password dialog and its corresponding default event procedure, Click, in the IDE.

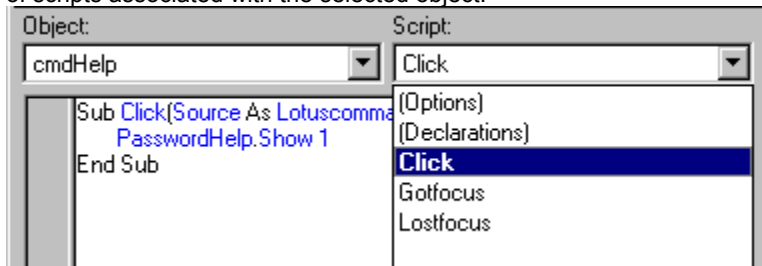


As you add dialogs and controls in dialogs, the IDE adds their names to the list of scriptable objects in its Object

drop-down box.



Click the object in the drop-down list to select it for scripting. Click the Script drop-down box in the IDE to display a list of scripts associated with the selected object.



In this case, the Click procedure for the cmdHelp button contains the statement required to display the dialog named PasswordHelp. When the user clicks the button labeled Help in the Password dialog, the second dialog named PasswordHelp will display.

To process user input in a control, use properties associated with that control. The LotusTextBox control in the Password dialog is named txtPasswordInput and Text property to store its values. To get the current value of the txtPasswordInput control in a script, use the following statements:

```
Dim PasswordValue as String
PasswordValue = Password.txtPasswordInput.Text
Print PasswordValue
```

You can then use the value of PasswordValue to perform validation routines.

### Running the dialog from your application

To call a dialog from your scripts in Word Pro, you must use methods for the dialog object. The following example illustrates some of the methods for managing dialogs at runtime.

```
Sub DialogFireDrill
' Display the dialog as a modal.
Password.Show 1
' Close the dialog.
Password.Close
' Display the dialog as a modeless.
Password.Show
' Hide the dialog temporarily.
Password.Hide
' Redisplay the dialog.
Password.Show
End Sub
```

---

{button ,AL('0)} [See related topics](#)

## How do I call DLL functions?

Your applications are not limited to calling procedures developing in the LotusScript language. If you have developed procedures in a language such as C, C++, or Visual Basic and compiled them in a dynamic-link library (DLL), you can call these procedures from within your script application.

To call procedures in a DLL, you need to know about the following:

- the name of the DLL
- the file path for the DLL (if it is not on your default path)
- the names and parameters for procedures that you want to call

The following example illustrates how to call a Win32 API function named `sndPlaySound` that is stored in the DLL file `C:\WINDOWS\SYSTEM\WINMM.DLL`. To play a Windows .WAV file in your script application using this Win32 function, you must first declare the function and then call it from one of your scripts.

**Note** You can copy the entire example into the IDE; the Script Editor automatically moves declarations and the sub into the appropriate sections of the script.

```
' Put the following statements in (Declarations) for (Globals) if you
' want to be able to call .WAV files from any script in your application.

' * RUNTIME DEPENDENCIES
' *   Files and paths: WINMM.DLL must be installed in C:\WINDOWS\SYSTEM
' *   or somewhere in your current file path. The sound file
' *   OFF2RACE.WAV must be installed in the subdirectory
' *   C:\WINDOWS\MEDIA.

' Declare a return value to use when you call the DLL
' function in your scripts.
Dim SoundReturnValue As Integer

' Declare the DLL function as a public function in LotusScript.
Declare Public Function sndPlaySound Lib "winmm" _
    Alias "sndPlaySoundA" _
    ( Byval WaveFile As String, Byval theFlags As Long ) _
    As Integer

' Declare some of the constants used by parameters of the DLL function.
Public Const SND_SYNC      = &H0000 ' Play synchronously (default)
Public Const SND_ASYNC     = &H0001 ' Play asynchronously
Public Const SND_NODEFAULT = &H0002 ' Silence (!default) if not found
Public Const SND_MEMORY    = &H0004 ' pszSound points to a memory file
Public Const SND_LOOP      = &H0008 ' Loop until next sndPlaySound
Public Const SND_NOSTOP    = &H0010 ' Don't stop currently playing sound

Sub TestSoundFiles
    ' The sub calls the declared function and specifies a .WAV file to play.
    SoundReturnValue = _
        sndPlaySound( "C:\WINDOWS\MEDIA\OFF2RACE.WAV", SND_SYNC )
End Sub
```

---

{button ,AL(,0)} [See related topics](#)

## How do I get information about documents?

Word Pro maintains detailed information about each of the documents that you use. While you are writing a document, choose File - Document Properties - Document to view all the information that Word Pro maintains for it. This information is also available to your scripts in the form of DocInfo properties. The current size of your file, for example, is available in the property DocInfo.DocSize.

The following example illustrates how to extract and use DocInfo property information for documents on disk and documents in memory. If you work with a collection of documents, you can build a report summarizing information about each document in the library by opening each document and collecting DocInfo information about it. The sub DocInfoProfiles passes the name of a document to profile to the sub BuildDocInfoProfile. All output from BuildDocInfoProfile is stored in the current document; no documents being profiled are modified.

```
Sub DocInfoProfiles
' * RUNTIME DEPENDENCIES
' *   Files and paths: The document GRP1STAT.LWP must be in the
' *     subdirectory C:\LOTUS\WORK\WORDPRO.

' Call the worker sub that loads the specified document and builds a
' profile of its DocInfo properties.
BuildDocInfoProfile "C:\LOTUS\WORK\WORDPRO\GRP1STAT.LWP"
' Profile a second document.
' BuildDocInfoProfile "YOURDOC.LWP"
End Sub

Sub BuildDocInfoProfile(DocName As String)
' Declare some variables to hold DocInfo information for each document.
Dim DocAuthor As String
Dim DocCreationDate As String
Dim DocPageCount As Long
Dim DocDescription As String

' Declare a variable for DocInfo values displayed in a message box.
Dim MsgText as String

' Open the document specified in the parameter DocName.
.OpenDocument DocName, "", "", "", False, True
' Do not close replace the current document with the one opened.
.ApplicationWindow.UserInterfacePrefs.IsReplacement = False

' Extract some DocInfo values for the document just opened.
With .ActiveDocument
    DocName = .FullName
    DocAuthor = .Docinfo.AuthorName
    DocCreationDate = .DocInfo.CreationDateString _
        & " at " & .DocInfo.CreationTimeString
    DocPageCount = .DocInfo.NumPagesInDoc
    DocDescription = .DocInfo.Description
End With

' Close the current document before writing any profile information.
.Close DocName

' Display profile results in a message box.
' Comment out if you don't want the sub to pause.
MsgText = "DocInfo: " _
    & DocName _
    & DocAuthor _
    & DocCreationDate _
    & DocPageCount
```

```

' Write the profile information to the end of the current document.
' Go to the end of the current document. Write a little record for
' each document profiled.
.Type "[ctrlEnd]"
.Type "[ENTER]======"
.Type "[ENTER]Document Name:[TAB][TAB]"
.Type DocName
.Type "[ENTER]Document Author:[TAB][TAB]"
.Type DocAuthor
.Type "[ENTER]Document Creation Date:[TAB]"
.Type DocCreationDate
.Type "[ENTER]Document Page Count:[TAB]"
.Type Str(DocPageCount)
.Type "[ENTER]=====[ENTER]"
.Type DocDescription
.Type "[ENTER]=====[ENTER]"
End Sub

```

Here is sample output from the BuildDocInfoProfile sub:

```

=====
Document Name:          C:\LOTUS\WORK\WORDPRO\GRP1STAT.LWP
Document Author:       James T. O'Connell
Document Creation Date: 6/14/96 at 1:38PM
Document Page Count:   3
=====

```

---

{button ,AL(`,0)} [See related topics](#)

## How do I add graphics to my document?

You can enhance your document by adding graphic elements such as diagram frames, imported pictures, and tables.

### Adding frames, tables, and and bitmaps

The following example illustrates how to add a frame for diagrams, a table, and a bitmap image to your current document.

```
Sub CreateGraphics
' * RUNTIME DEPENDENCIES
' *   Files and paths: You must have the bitmap file HELPBUTN.BMP in the
' *       subdirectory C:\LOTUS\WORDPRO.

' Declare a text variable for message box output.
Dim MsgText as String

' Declare a Layout object for the new diagram frame.
Dim MyFrame As Layout

' Declare a text variable for the name and file
' path of a bitmap file to import.
Dim BMPName As String
' Specify a bitmap file to import.
BMPName = "C:\LOTUS\WORDPRO\HELPBUTN.BMP"

' Enter some blank lines at the top of the document.
.Type "[ctrlHome][Enter][Enter][Enter][Enter][Enter]"
.Type "[Enter][Enter][Enter][Enter]A new frame.[Enter]"

' Create a new 1" x 1" diagram frame at the cursor.
.CreateFrame False, "Default Frame", 1440, 1440

' Anchor the frame to the preceding paragraph.
.Frame.Anchor 0, $LwpConditionTypeAllPages, _
    $LwpRelativeTypeLytInlineNewline

' Bind the layout variable MyFrame to the new frame.
' The default name for the first frame you create is Framel.
Set MyFrame = Bind("!Body:Framel")
' Display the name of the new diagram frame.
MsgText = "Created " & MyFrame.Name
MsgBox MsgText

' Enter some blank lines at the top of the document.
.Type "[ctrlHome][Enter][Enter][Enter][Enter][Enter]"
.Type "[Enter][Enter][Enter][Enter]A new table.[Enter]"

' Create a table with the following options:
'   Table style = "Default table"
'   Number of columns = 5
'   Number or rows = 6
.CreateTable False, "Default Table", 5, 6

' Enter some blank lines at the top of the document.
.Type "[ctrlHome][Enter][Enter][Enter][Enter][Enter]"
.Type "[Enter][Enter][Enter][Enter]A new bitmap.[Enter]"

' Create a frame at the cursor that contains
' a bitmap image (".bmp" format) imported from a disk file.
.ImportGraphic BMPName, ".bmp", False, False, "INTERNAL_ID"
```

```

' Anchor the frame to the preceding paragraph.
.Frame.Anchor 0, $LwpConditionTypeAllPages, _
  $LwpRelativeTypeLytInlineNewline
' Assign a name to the new frame.
.Layout.Name = "MyBMPFrame"
End Sub

```

**Tip** It is useful to name objects at the time you create them, because you can subsequently select named objects in a collection by iterating through collection members.

### Navigating to graphic objects

The following example illustrates how to navigate to a frame named "MyBMPFrame".

```

Sub GoToBMP
' * RUNTIME DEPENDENCIES
' *   Objects: You must have a frame named MyBMPFrame in your document.

Dim TargetFrame As Layout
Set TargetFrame = Bind("!Body:MyBMPFrame")
TargetFrame.GotoLayout
End Sub

```

### Manipulating graphics

You may need to modify the placement or appearance of a graphic. The following example illustrates how to create a frame, change some properties for it, and then change its placement on each page of the document.

```

Sub MoveBannerGraphic
' Declare a Layout object for a new diagram frame.
Dim BannerGraphic as Layout

' Create a 1" x 1" frame at the insertion point.
' Use current defaults for frames.
.CreateFrame False, "Default Frame", 1440, 1440

' Assign a name for the new frame.
.Frame.Layout.Name = "NewTestFrame"
' Assign the variable BannerGraphic to the new frame.
Set BannerGraphic = Bind("!Body:NewTestFrame")

' Set some frame properties: the graphic should appear on every
' page and its placement on each page should be relative
' to page margins (versus paragraphs or columns surrounding it).
.Frame.Anchor $LwpAnchorWhereLayout, _
  $LwpConditionTypeAllpages, _
  $LwpRelativeTypeLytParent
.Frame.Layout.WrapType = $LwpWrapTypeLayoutNoWrapAround

' Place the frame 2" below the top margin on each page.
.Frame.Layout.RelativeYDistance = 2880
' Place the frame 3" from the right margin on each page.
.Frame.Layout.RelativeXDistance = 3960

End Sub

```

---

{button ,AL(,0)} [See related topics](#)



## How do I navigate to objects in a document?

There are several ways to select or navigate to objects in a document:

- searching collections for named objects
- using GoTo methods for named objects
- searching for text

## Selecting or activating named objects

If you opened more than one document in your script application, you can activate a named document by working with one of the collection objects in Word Pro called DocWindows. Word Pro builds collections for many type of objects: bookmarks, document windows, cells in tables, divisions, footnotes, glossary items, layout objects, and document versions.

The following example illustrates the basic process of navigating to a named object in a collection by iterating through members of the collection.

```
Sub ActivateDocByName
    ' Get the name of an active document.
    Dim DocumentName as String
    DocumentName = Inputbox$("Please Enter the Name of the Document to show")
    ' Iterate through the members of the DocWindows collection.
    Forall myDoc In .ApplicationWindow.DocWindows
        ' Find the name of the active document and activate the
        ' document window.
        If myDoc.Name = DocumentName Then
            myDoc.Show
        End If
    End Forall
End Sub
```

## Navigating to named objects with a GoTo method

Another method of navigating to named objects is more direct. The following example illustrates how to navigate to a named frame and a named bookmark using GoTo methods.

**Note** The names of bookmarks in Word Pro are case sensitive.

```
Sub GoToNamedObjects
    ' Declare a layout variable for the named frame.
    Dim MyFrame As Layout

    ' Create a named frame.
    .CreateFrame False, "Default Frame", 1440, 1440
    .Frame.Layout.Name = "TestFrame"

    ' Assign the layout variable to the named frame.
    Set MyFrame = Bind("!Body:TestFrame")

    ' Execute a goto method on the named frame.
    MyFrame.GotoLayout

    ' An alternative way to go to the named frame using a goto method.
    .GoToLayout "Body:TestFrame"

    ' Create a bookmark named "ReadThisKyle"
    TEMP__ = Mark($LwpMarkerTypeBookmark)
    .Division.Foundry.Markers(TEMP__).PageNumber = 1
    .Division.BookmarkManager.AddBookmark "ReadThisKyle", TEMP__

    ' Go to the bookmark named "ReadThisKyle".
    .GoToBookmark("ReadThisKyle")
End Sub
```

Although some objects do not have a persistent name, you can use a GoTo method to navigate to them. The following example illustrates how to navigate to a specific page by page number or by its position in the file (first or last).

```
Sub GoToPage
' * RUNTIME DEPENDENCIES
' *   Objects: There must be three or more pages in your document.

'   Go to page three in the current document.
.GoToPage 3
'   Go to the last page in the current document.
.ApplicationWindow.UserInterfacePrefs.GoToSelection = _
    $LwpGoToTypeLastpage
.GoToPage 32767
End Sub
```

### **Navigating to text with search**

Individual words, sentences, and paragraphs in your documents do not have unique names or identifiers. One way to support navigation through passages in your document is to create named bookmarks at specific places; another is to search for unique text within passages.

The following example illustrates how to copy a paragraph containing a keyword, "Bennings Electronics", from the body of your document to the end of your document.

```
Sub ExtractKeywordParagraph
' * RUNTIME DEPENDENCIES
' *   Objects: There must be at least one occurrence of the string
' *       "Bennings Electronics" in your document.

'   Search for the first instance of the keyword.
.Application.FindAndReplace.FindString = "Bennings Electronics"
.InitFindAndReplace True
.Find
'   Select the entire paragraph containing the keyword.
.SelectParagraph
'   Copy the selected paragraph.
.CopySelection
.InitFindAndReplace True
'   Go to the end of the current document.
.Type "[ctrlEnd]"
'   Paste the copied paragraph.
.Type "[ENTER]=====[ENTER]"
.Paste
End Sub
```

---

{button ,AL('',0)} [See related topics](#)

## How do I use OLE objects?

As an OLE container, Word Pro lets you create and automate a variety of OLE objects in your documents. There are three types of OLE objects that you can create and automate in Word Pro:

- Embedded OLE 2 objects
- Embedded files as OLE objects
- Embedded OLE Custom Controls (OCX)

## Embedding OLE 2 objects in your document

The following example illustrates how to create OLE 2 objects in your Word Pro document.

```
Sub EmbedObject
    ' Embed a Paintbrush object.
    .CreateOleNew "PBrush", 0, True
    ' Embed a Lotus Approach application object.
    .CreateOleNew "ApproachApplication", 0, True
End Sub
```

**Tip** The names of the OLE servers that you can specify in your CreateOleNew statement are available in the Windows Registry in HKEY\_CLASSES\_ROOT.

## Embedding files as OLE objects

You can also add to your Word Pro document OLE objects instantiated from existing documents on disk. Unlike the CreateOleNew method described above, the OLE objects created in the following example contain data.

```
Sub EmbedFile
    ' Embed an Approach database file.
    .CreateOleEmbeddedFile _
        "{00000000-0000-0000-0000-000000000000}", _
        "C:\LOTUS\WORK\APPROACH\TEST.APR", 0
End Sub
```

## Embedding OLE Custom Controls

You can add registered OLE Custom Controls to your Word Pro document and develop scripts for those controls as you would any custom control in the Dialog Editor.

```
Sub EmbedOCX
    ' Embed the OLE control named LotusCommandButton.
    .CreateOleNew "Lotus.CommandButton.1", 0, True
End Sub
```

---

{button ,AL(`,0)} [See related topics](#)

## How do I call Windows applications?

You can call Windows applications and services from your Word Pro script applications with the LotusScript Shell() function. To make a basic call, you must know the name and path of the Windows application, for example C:\WINDOWS\IMPLAYER.EXE. More sophisticated calls require knowing the command line parameters for the application that would allow you to pass information from your script application to the Windows application. The MPLAYER.EXE application will load a multimedia file if you specify one as a parameter in your call, for example C:\WINDOWS\IMPLAYER TADA.WAV.

The following example demonstrates how to call a Windows Help file or particular topics within a custom Help file that you develop for your application using the Microsoft Windows help compiler. Once you have a script that can call Windows Help topics, you can add context-sensitive help to many objects in your own script application.

```
Sub CallHelp
' * RUNTIME DEPENDENCIES
' *   Files and paths: The Word Pro LotusScript help file WP071EN.HLP
' *   must be in the subdirectory C:\LOTUS\WORDPRO.

' Declare a return variable for the Shell() function.
Dim HelpReturnValue As Integer

' Declare a variable for the name and path of the Help file.
Dim HelpFilename As String
' Specify the name and path of the Help file.
HelpFileName = "C:\LOTUS\WORDPRO\WP0N71EN.HLP"

' Display the Contents of a help file named WP0N71EN.HLP.
HelpReturnValue = Shell("WINHLP32.EXE " & HelpFileName, 1)

' Call a topic in WP0N71EN.HLP named "The Help Method" identified by its
' the help context ID H_HELP_METHOD_MEMDEF. The command-line switch -I
' specifies that WinHelp search for the specified context ID.
HelpReturnValue = Shell("WINHLP32.EXE -I H_HELP_METHOD_MEMDEF & HelpFileName, 1)

' Call the same topic in WP0N71EN.HLP identified by its resource ID
' number 20105. These resource IDs appear in the header file that you
' compile with the help file. The command-line switch -N specifies that
' WinHelp search for the specified resource ID.
HelpReturnValue = Shell("WINHLP32.EXE -N 20105 " & HelpFileName, 1)
End Sub
```

**Note** Word Pro also supports a Help method that lets you call Help files or specific Help topics within a Help file.

---

{button ,AL(`,0)} [See related topics](#)

## How do I manage find and replace?

Word Pro provides sophisticated Find and Replace features that you can incorporate in your script applications. You can perform Find and Replace at three levels:

- against text in your documents
- against named styles in your document
- against strings that you extract from your document

### Finding and replacing text

The following example illustrates how to do a global Find and Replace of text.

```
Sub ReplaceList
' * RUNTIME DEPENDENCIES
' *   Objects: There must be at least one occurrence of the word "OCX"
' *       in your document.

' Specify find and replace strings for the procedure
' managing the global find and replace.
GlobalTextFindReplace "OCX", "ActiveX"
End Sub

Sub GlobalTextFindReplace(tmpFindString As String, tmpReplaceString As String)
' Use the value of the tmpFindString parameter to the
' FindString property.
.Application.FindAndReplace.FindString = tmpFindString
' Use the value of the tmpReplaceString parameter to the
' ReplaceString property.
.Application.FindAndReplace.ReplaceString = tmpReplaceString
' Search the entire document.
.Application.FindAndReplace.Where = $LwpLookWhereEntireDocument
' Include all text streams in the document.
.Application.FindAndReplace.IncludeList = $LwpIncludeListAllText
' Search for exact case matches.
.Application.FindAndReplace.FindExactCase = True
' Replace with an exact case.
.Application.FindAndReplace.ReplaceExactCase = True

.InitFindAndReplace True
' Find the first occurrence.
.Find
' Replace all occurrences.
.ReplaceAll
.InitFindAndReplace False
End Sub
```

### Finding and replacing named styles

You can also Find and Replace named styles in your document using a variation of the above statements.

```
Sub GlobalStyleFindReplace
' * RUNTIME DEPENDENCIES
' *   Objects: There must be at least one occurrence of a paragraph in the
' *       named style "Bullet 1".

' (Optional) Go to the beginning of the document as
' a little insurance.
.Type "[ctrlHome]"
' Specify the name of the style to search for.
.Application.FindAndReplace.FindStyleName = "Bullet 1"
' Specify the name of the style to substitute.
```

```

        .Application.FindAndReplace.ReplaceStyleName = "Bullet 2"
        .InitFindAndReplace True
        .ReplaceAll
    End Sub

```

### Finding and replacing string elements

Find and Replace works against text and styles in your document. You may also need to replace elements of a longer string that you are working with in your scripts. You could write the string to your document, but it would be more effective to perform the Find and Replace within your working script.

```

Sub StringElementReplacement
    ' Declare a string variable for the ReplaceString function.
    Dim ReplaceIt As String
    ' Declare a string variable to hold the string containing
    ' an element to be replaced.
    Dim MyInput As String
    ' Specify the string containing an element to be replaced.
    MyInput = "onetwothreefourfive"
    ' Call the ReplaceTextString function to perform the find and replace.
    ReplaceIt = ReplaceTextString(MyInput, "two", "Three")
    ' Direct the results of the function call to the IDE output panel.
    Print ReplaceIt
    ' Output = "oneThreethreefourfive"
End Sub

```

```

Function ReplaceTextString (inputString As String, findString As String, replaceString
As String) As String
    ' Declare a temporary string variable.
    Dim tempString As String
    ' Declare two variables to manage find and replace boundaries
    ' within the complete input string.
    Dim findPos As Integer
    Dim startSearchAt As Integer

    ' Assign the temporary string variable to the value of the
    ' inputString parameter.
    tempString = inputString
    findPos = Instr(tempString, findString)
    ' Select findString within tempString.
    While findPos
        tempString = Left$(tempString, findPos - 1) _
            & replaceString _
            & Right$(inputString, Len(tempString) _
                - (findPos + Len(findString)) + 1)
        findPos = Instr(findPos + Len(replaceString), _
            tempString, _
            findString)
    Wend

    ' Replace the selected findString element with the value of the
    ' replaceString parameter. Return the entire modified string to
    ' to the calling procedure.
    ReplaceTextString = tempString
End Function

```

---

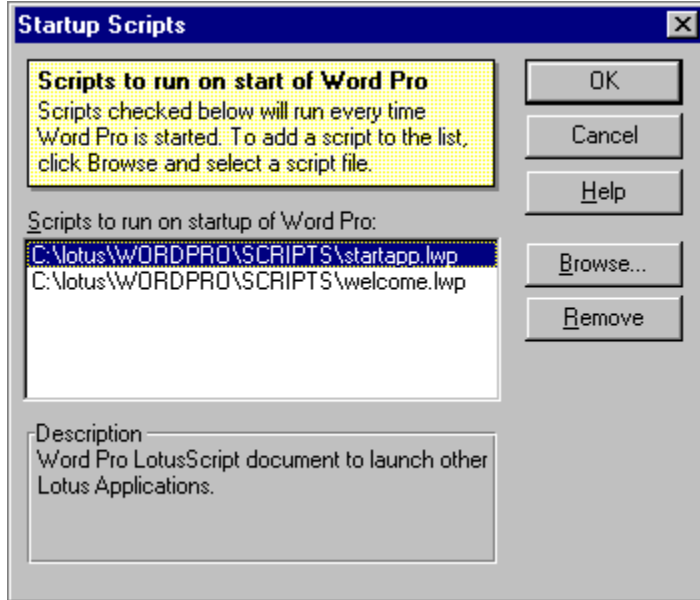
{button ,AL(,0)} [See related topics](#)

## How do I run scripts when opening Word Pro or a document?

There are several ways to have scripts execute when you start a Word Pro session or open a document.

### Running scripts when you load Word Pro

If you have scripts that you want to execute each time you start a Word Pro session, choose Edit - Scripts & Macros - Set Startup Scripts.



You can specify one or more Word Pro documents containing scripts, text files containing scripts, or LotusScript Object (LSO) files. Any global scripts contained in the documents you specify will be executed whenever you start your Word Pro session.

**Tip** Another way to run a script each time you load Word Pro would be to add the name of that script to the startup command that you use when you load Word Pro from Windows:

```
C:\LOTUS\WORDPRO.EXE C:\SCRIPTS\STARTUP.LWP!STARTUPSUB
```

### Running scripts when you open a document

If you have procedures that you want to execute whenever you open a particular document, there are several possibilities to consider.

When you open a document, Word Pro executes procedures in the following scripts in the following order:

- !Globals Initialize
- !Document Initialize
- !Document Opened
- !WordPro Documentopened
- !Globals Main

**Note** The !Globals Initialize and !Document Initialize scripts also execute when you close your document.

### Controlling display features at startup

If your script application does not need all the default display features offered in Word Pro, you can selectively turn them off when you load your document.

Place the following statements in any of the scripts mentioned above to simplify the number of features displayed when you run your script application.

```
' Startup settings to turn off palettes and tools.
' =====
' CleanScreen mode is useful in setting startup display preferences.
' If you want to reduce the number of palettes and toolbars
' displayed while you run your script application, you can turn
' tools off one by one or you could set preferences for CleanScreen
```

```

' mode and then turn that on.
With .ApplicationWindow.UserInterfacePrefs.WinViewPrefs
    IsViewTitleBarCleanScrn = True
    IsViewMenuCleanScrn = True
    IsViewSmartIconsCleanScrn = False
    IsViewStatusBarCleanScrn = False
    IsVerticalScrollBarCleanScrn = False
    IsHorizontalScrollBarCleanScrn = False
    IsViewReturnIconCleanScrn = True
End With

' Use CleanScreen mode.
.ApplicationWindow.UserInterfacePrefs.CleanScreenMode = True
.ActiveDocWindow.WinViewPrefs.Refresh

' Startup settings to turn off text markers and symbols.
' =====
' Turn off outline tools (if currently displayed).
.ActiveDocWindow.WinViewPrefs.IsInOutline = False

' Turn off highlighting for misspelled words.
.ActiveDocWindow.WinViewPrefs.IsDisplayMisspelled = False

' Turn off displayed symbols for tabs, carriage returns, bookmarks, section
' breaks, and so on.
With .ActiveDocWindow.WinViewPrefs
    IsViewTabs = False
    IsViewReturns = False
    IsViewRulerMarks = False
    IsPageBreakMarks = False
    IsViewSectionBreakMarks = False
    IsViewColumnBreakMarks = False
    IsViewBookmarks = False
    ViewType = &H1
    IsHideHeaderFooter = False
End With

```

---

{button ,AL(' ,0)} [See related topics](#)



## How do I style text and paragraphs?

You can style text and paragraphs in your documents in three ways:

- Change the style properties of text.
- Apply a named character style to some text.
- Change the paragraph style for a paragraph.

### Changing the style properties of text

The following example illustrate how to change the font, point size, italicization, and color of a selected sentence.

```
Sub LocalStyleChange
    ' Select the current sentence.
    .SelectSentence
    ' Apply a new font.
    .Text.Font.FontName = "Gill Sans"
    ' Apply a new point size.
    .Text.Font.Size = 9.00
    ' Make the text italic.
    .Text.Font.RevertToStyle $LwpFontPropertyItalic
    ' Change the text color to red.
    .Text.Font.FontColor.Red = 255
    .Text.Font.FontColor.Blue = 0
    .Text.Font.FontColor.Green = 0
    .Text.Font.FontColor.Override = $LwpColorOverrideRgb
End Sub
```

### Applying a named character style

Named character styles in Word Pro let you apply a set of text properties to a piece of selected text without changing its paragraph style. The following example illustrates how to apply a character style named "My Character Style" to a selected sentence.

```
Sub ApplyCharacterStyle
' * RUNTIME DEPENDENCIES
' *   Objects: There must be a character style named "My New Char Style"
' *       in your document or attached SmartMaster.

    ' Select the current sentence.
    .SelectSentence
    ' Apply the named character style to the selected sentence.
    .Text.CharacterStyleName = "My New Char Style"
End Sub
```

### Applying a named paragraph style

Named paragraph styles are powerful tools for managing the overall format of your document. Changing the paragraph style for one or more paragraphs in your document is similar to applying a name character style.

```
Sub ApplyNewParagraphStyle
' * RUNTIME DEPENDENCIES
' *   Objects: There must be a paragraph style named "Default Text"
' *       in your document or attached SmartMaster.

    ' Select the current paragraph as insurance.
    .SelectParagraph
    ' Apply the named paragraph style "Default Text" to the
    ' selected paragraph.
    .Text.ParagraphStyleName = "Default Text"
End Sub
```

---

{button ,AL(,0)} [See related topics](#)



## How do I use timer functions?

It is sometimes necessary to pause execution of a script for a number of seconds. The following scripts illustrate how to use a generic timer sub to manage such pauses in your script application.

```
Sub CallTheTimer
    ' Calls the GenericTimer sub and specifies the
    ' a number of seconds that it should pause before
    ' continuing execution.
    GenericTimer 6
End Sub

Sub GenericTimer(duration As Integer)
    MessageBox("Begin a pause for " & duration & " seconds")

    stopWatch = Timer
    While ((Timer - stopWatch) < duration)
    Wend
    MessageBox(duration & " seconds is up")
End Sub
```

---

{button ,AL(`,0)} [See related topics](#)

'Example: Shadow property  
,

'Example: Shadow property  
,

```
'Example: SendFrameToBackOne method
' This example creates two frames and changes the order of the layering
' for the two frames.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewFrame 3285, 1200, 1575, 1830
.Frame.Layout.Background.Color.Red = 82
.Frame.Layout.Background.Color.Blue = 239
.Frame.Layout.Background.Color.Green = 145
.Frame.Layout.Background.Color.Override = $LwpColorOverrideRgb
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent

.Deselect

.NewFrame 5285, 2200, 1575, 1830
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent
.Frame.Layout.Background.Color.Red = 182
.Frame.Layout.Background.Color.Blue = 139
.Frame.Layout.Background.Color.Green = 45
.Frame.Layout.Background.Color.Override = $LwpColorOverrideRgb

MessageBox "Click OK to send frame to back. ",MB_OK,"Example Script"
.SendFrameToBackOne
```

```
'Example: SendFrameToBack method
' This example creates two frames and changes the order of the layering
' for the two frames.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewFrame 3285, 1200, 1575, 1830
.Frame.Layout.Background.Color.Red = 82
.Frame.Layout.Background.Color.Blue = 239
.Frame.Layout.Background.Color.Green = 145
.Frame.Layout.Background.Color.Override = $LwpColorOverrideRgb
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent

.Deselect

.NewFrame 5285, 2200, 1575, 1830
.Frame.Anchor $LwpAnchorWhereLayout, $LwpConditionTypeOnlyspecificpage,
$LwpRelativeTypeLytParent
.Frame.Layout.Background.Color.Red = 182
.Frame.Layout.Background.Color.Blue = 139
.Frame.Layout.Background.Color.Green = 45
.Frame.Layout.Background.Color.Override = $LwpColorOverrideRgb

MessageBox "Click OK to send frame to back. ",MB_OK,"Example Script"
.SendFrameToBack

MessageBox "Click OK to bring frame to front. ",MB_OK,"Example Script"
.BringFrameToFront
```

```
'Example: SendMailSelectedText method
' This example inserts some sample text into the current document, selects the
' text, and then displays the team mail dialog box.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work ' and a
VIM compliant mail application must be available.

.Type "This is some sample text to be mailed."
.SelectParagraph
.SendMailSelectedText
```



'Example: ServerFormat property

'This example script has not yet been created.

'Example: SetArrayProp method

'This example script has not yet been created.

```
'Example: SetButtonText method
' This example creates a new button to the status bar and then adds text to
' the button. The STATUSBARBUTTONOVERRIDE TEXT is then bound to the
' SetTheButtonText subroutine to set the button text during needs repainting.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ButtonName As String
Dim NewButton As StatusBarButton
With .ApplicationWindow.StatusBar
    ButtonName = .CreateNewbutton (0,0,100,&H1) 'create the new button
    Set NewButton = .StatusBarButtons(ButtonName)
    With .StatusBarButtons(ButtonName)
        .SetOverRideText("New Button...")
        Call .SetButtonText("Button",True)
        .InvalidateButton
        On Event STATUSBARBUTTONOVERRIDE TEXT From NewButton Call
            SetTheButtonText
    End With
    .InvalidateWholeBar ' Force the bar to repaint
End With
End Sub

Sub SetTheButtonText (Source As StatusBarButton, ButtonName As String)
    'Add the the button text each time the status bar needs repainting.
    Source.SetOverRideText("New Button...")
    End 2
End Sub
```

'Example: SetContextOfBar property

'This example script has not yet been created.

'Example: SetData method

' This example creates a dataset named 'ExampleDataSet' off of the active  
' document. The 'FirstName' and 'LastName' items are created and filled with  
' data. Finally the values for the dataset items are printed to the Script  
' Editor Output panel. Since no dataset item named 'Address' was defined, the  
' default dataset value will be printed in the last statement.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
Dim DataSetName as String
```

```
Dim Defaultvalue as String
```

```
Dim DataSet As WPDataSetCollection
```

```
Set DataSet = .ActiveDocument.WPDataSets
```

```
DataSetName = "ExampleDataSet"
```

```
Defaultvalue = "Default"
```

```
DataSet(DataSetName).SetData "FirstName","John"
```

```
DataSet(DataSetName).SetData "LastName","Doe"
```

```
Print DataSet(DataSetName).GetData("FirstName",Defaultvalue)
```

```
Print DataSet(DataSetName).GetData("LastName",Defaultvalue)
```

```
Print DataSet(DataSetName).GetData("Address",Defaultvalue)
```

'Example: SetDocumentEpoch method

'This example script has not yet been created.

'Example: SetFieldFormula method

'This example script has not yet been created.

'Example: SetFocus method

'This example script has not yet been created.



```
'Example: SetFormula method
' This example creates a table and enters a formula. The formula is then
' retrieved and printed to the Lotus Script Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.Table.CellLayout(4,0).GotoLayout
.Table.CellEngine.SetFormula 4,0,"@SUM(A1:A2)"
Print .Table.CellEngine.GetFormula(4, 0)
```

```
'Example: SetLastUsedFilter method
' This example inserts the 'TOON1' Word Pro Drawing graphic into the current
' document. The graphic is placed in a frame based upon the 'Default
' Graphic/OLE' frame style.
' RUNTIME DEPENDENCIES: You must have a document open and have installed the
' the Word Pro clipart into the \\DRAWSYM subdirectory for this script to '
' work.

Dim FileName as String
FileName = .Application.Path & "\\DRAWSYM\\TOON1.SDW"
.ApplicationWindow.Filter.SetLastUsedFilter $LwpFilterTypeGraphic,"Word Pro draw"
.ApplicationWindow.UserInterfacePrefs.LastGraphicType = "Word Pro Draw"

.ImportGraphic FileName, ".SDW", False, False, "Default Graphic/OLE"
```

'Example: SetLineOneSide method

'This example script has not yet been created.

'Example: SetLinesAllSides method

'This example script has not yet been created.

'Example: SetLinkSource method

'This example script has not yet been created.

'Example: SetMinimumOrigin method

'This example script has not yet been created.

'Example: SetNamedProperty method

' This example creates a named property, 'ExampleProp' on the active document  
' and assigns it a value. The value is then printed to the Lotus Script Output  
' panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

**.ActiveDocument.SetNamedProperty "ExampleProp", "Here is some data."**

.ActiveDocument.GetNamedProperty "ExampleProp"

'Example: SetOverrideGraphic method

'This example script has not yet been created.



```
'Example: SetOverrideText method
' This example creates a new button to the status bar and then adds text to
' the button. The STATUSBARBUTTONOVERRIDE TEXT is then bound to the
' SetTheButtonText subroutine to set the button text during needs repainting.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ButtonName As String
Dim NewButton As StatusBarButton
With .ApplicationWindow.StatusBar
    ButtonName = .CreateNewbutton (0,0,100,&H1) 'create the new button
    Set NewButton = .StatusBarButtons(ButtonName)
    With .StatusBarButtons(ButtonName)
        .SetOverrideText("New Button...")
        Call .SetButtonText("Button",True)
        .InvalidateButton
        On Event STATUSBARBUTTONOVERRIDE TEXT From NewButton Call
            SetTheButtonText
    End With
    .InvalidateWholeBar ' Force the bar to repaint
End With
End Sub

Sub SetTheButtonText (Source As StatusBarButton, ButtonName As String)
    'Add the the button text each time the status bar needs repainting.
    Source.SetOverrideText("New Button...")
    End 2
End Sub
```

```
'Example: SetPattern method
' This example create a table with 10 columes and 1 row and changes the color
' pattern for every other row.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 1, 10
.SelectEntireTable

BackgroundColor = 92654
PatternColor = 23424

.Table.TableFill.SetPattern $LwpTableFillStyleEveryotherrow,
$LtsFillDottedZigzag ,BackgroundColor, PatternColor
.Deselect
```

'Example: SetPopupAlignment method

'This example script has not yet been created.

'Example: SetPopupIndex method

'This example script has not yet been created.

'Example: SetPopupWidthType method

'This example script has not yet been created.

'Example: SetPopupWidth method

'This example script has not yet been created.

'Example: SetPowerFieldValue method

'This example script has not yet been created.

```
'Example: SetRGB method
' This example creates a table with 5 rows and 5 columns into the current
' document. The background and pattern colors are then changed for the current
' cell.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.Table.TableFill.Background.Pattern = $LtsFillNwToSeGrad

.Table.CurrentCell.Background.BackColor.SetRGB 255,255,255

.Table.TableFill.Background.Color.Override = $LwpColorOverrideRgb
.Table.TableFill.Background.BackColor.Red = 82
.Table.TableFill.Background.BackColor.Blue = 239
.Table.TableFill.Background.BackColor.Green = 145

.Table.TableFill.Background.BackColor.SetRGB 82,239,145

.Table.TableFill.Background.BackColor.Override = $LwpColorOverrideRgb
.Table.TableFill.FillStyle = $LwpTableFillStyleAll
```



'Example: SetStorage method

'This example script has not yet been created.

```
'Example: SetStyle method
' This example inserts some text into the current document, changes some
' text properties and then creates a new paragraph style based on those
' properties.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
With .Text
    .InsertText "This is some sample text."
    .Select $LwpSelectObjectTypeParagraph
    .Font.Size = 15.00
    .Font.Bold = True
    .Alignment.AlignmentType = $LtsAlignmentHorizCenter
    .SetStyle $LwpStyleTypeParagraph, "New Example Style",
End With
```

```
'Example: SetPattern method
' This example create a table with 10 columes and 1 row and changes the color
' pattern for every other row.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
.CreateTable False, "Default Table", 1, 10
.SelectEntireTable
```

```
BackgroundColor = 92654
PatternColor = 23424
```

```
.Table.TableFill.SetPattern $LwpTableFillStyleEveryotherrow,
$LtsFillDottedZigzag ,BackgroundColor, PatternColor
.Deselect
```

'Example: SetTabsDialog property

'This example script has not yet been created.

'Example: SetTOCProperties method

'This example script has not yet been created.

'Example: SetUpEnvelopeMerge method  
'This example script has not yet been created.

```
'Example: Shade method
' This example shades the first 8 words the the example sentence inserted
' into the current document.
' RUNTIME DEPENDENCIES: You must have a document open with the cursor
' positioned to the left of some text for this script to work.

.Text.InsertText "The first eight words of this sentence will be shaded."
.Text.Backward $LwpNavigateObjectTypeSentence, 1
.Text.Shade $LwpLocationTypeWord,$LwpNavigateDirectionRight,8
```

'Example: ShowContainers method

'This example script has not yet been created.



'Example: ShowCursor method

'This example script has not yet been created.

'Example: ShowDivisionTabs property

'This example script has not yet been created.

'Example: ShowGraphicPreview property  
'This example script has not yet been created.

'Example: ShowHiddenText property

'This example script has not yet been created.

'Example: ShowIconBars method

' This example hides all SmartIcon bars the redisplays them after the message  
' box is closed.

.ApplicationWindow.IconBarManager.HideIconBars

MessageBox "Click OK to show SmartIcon bars.",MB\_OK,"Example Script"

**.ApplicationWindow.IconBarManager.ShowIconBars**

```
'Example: ShowInContext property
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager
IcnPallet = "Comment Tools"
Set IcnMgr = .ApplicationWindow.IconBarManager

' Set icon pallet to show in its context
IcnMgr.IconBars(IcnPallet).ShowInContext = True
' This will force a redraw of IconBars
IcnMgr.ShowIconBars
MsgStr = "|" & IcnPallet
MsgStr = MsgStr & "|" pallet is now displayed, click OK to hide this pallet"|
MessageBox MsgStr, 48, "Script Example"

' Reset icon pallet not to show in its context
IcnMgr.IconBars(IcnPallet).ShowInContext = False
IcnMgr.ShowIconsBars
```

'Example: ShowMailDisabled property

'This example script has not yet been created.

'Example: ShowNoWelcomeBox property

'This example script has not yet been created.



```
'Example: ShowScrollBars method
' This example hides the vertical and horizontal scroll bars if the are
' visible and shows them if they are hidden.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ScrollVert as Long
Dim ScrollHorz as Long
MessageBox "Click OK to hide scroll bars ",MB_OK,"Example Script"
ScrollVert = False
ScrollHorz = False
.ActiveDocWindow.ShowScrollBars ScrollVert,ScrollHorz
MessageBox "Click OK to show scroll bars ",MB_OK,"Example Script"
ScrollVert = True
ScrollHorz = True
.ActiveDocWindow.ShowScrollBars ScrollVert,ScrollHorz
```

'Example: ShowStatistics property

'This example script has not yet been created.

```
'Example: ShowStatusBar method
' This example hides the status bar if it is visible, and shows it if it is
' hidden.
If .ApplicationWindow.StatusBar.Visible = True Then
    .ApplicationWindow.StatusBar.HideStatusBar
Else
    .ApplicationWindow.StatusBar.ShowStatusBar
End If
```

'Example: ShowTabs property

'This example script has not yet been created.

```
'Example: Show method
'[DocWindow.Show]
' This example gets the name of the document to activate and then
' cycles through all the documents to find the one requested
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim DocName As String
DocName = InputBox("Enter the name of the document to activate:")
Forall Document In .Documents
    If Document.Name = DocName Then
        Document.Activate
        .ActiveDocWindow.Show
    End If
End Forall
```

```
'[IconBar.Show]
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager
IcnPallet = "Comment Tools"
Set IcnMgr = .ApplicationWindow.IconBarManager

' Set the context and show the bar
IcnMgr.IconBars(IcnPallet).ShowInContext = True
IcnMgr.IconBars(IcnPallet).Show
```

```
MsgStr = "|" & IcnPallet
MsgStr = MsgStr & "|" pallet is now displayed, click OK to hide this pallet"|
MessageBox MsgStr, 48, "Script Example"
```

```
' Reset the context and hide the bar. You can hide the bar without resetting
' the context first. But the bar will reappear once the context in the
' document returns to the setting for this pallet
IcnMgr.IconBars(IcnPallet).ShowInContext = False
IcnMgr.IconBars(IcnPallet).HideIconBar
```

'Example: SilverBullets property

'This example script has not yet been created.

'Example: SilverBullet property

'This example script has not yet been created.

```
'Example: SimulateButtonClick method  
' This example simulates clicking on the font status bar button.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim StatBar As StatusBar  
Dim ButtonId as Integer
```

```
Set StatBar = .ApplicationWindow.StatusBar  
Forall Button In StatBar.StatusBarButtons  
    ButtonId = Button.GetButtonId  
    If (ButtonId = StatBar.GetStandardButtonId($LwpStandButtFontButton)) Then  
        Button.SimulateButtonClick  
    End If  
End Forall
```



'Example: SizeStyleName property

'This example script has not yet been created.

```
'Example: Size property
'Sub Main
Print "======"
Forall x In .division.foundry.paragraphstyles
Print x.font.ActualName & " = " & x.font.size
End Forall
Forall x In .Division.foundry.paragraphstyles
x.font.FontName = "Arial"
End Forall
Print " -----"
Forall x In .Division.foundry.paragraphstyles
Print x.font.ActualName & " = " & x.font.size
End Forall
End Sub
```

'Example: SizingUnitName property

'This example script has not yet been created.

'Example: SizingUnits property

'This example script has not yet been created.

'Example: Skipped property

'This example script has not yet been created.

'Example: SkipWordMode property

'This example script has not yet been created.

'Example: Skip method

'This example script has not yet been created.

```
'Example: SmallCaps method
' This example displays the popup style menu located on Word Pro's status bar.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
.Type "This Is A Test Of Small Caps."
.SelectParagraph
.SmallCaps
MessageBox "Click OK undo the smallcaps change.",MB_OK,"Example Script"
.SmallCaps
```



'Example: SmallCaps property

'This example script has not yet been created.

'Example: SmallFileFormat property  
'This example script has not yet been created.

'Example: SmartCorrects property

'This example script has not yet been created.

'Example: SmartCorrect property

'This example script has not yet been created.

'Example: SmartFill property

'This example script has not yet been created.

'Example: SmartLevel property

'This example script has not yet been created.

```
'Example: SmartSumColumn method
' This example creates a table with 5 rows and 5 columns into the current
' document. Headers are created for the first column and row. Numbers are
' inserted elsewhere and the column and rows are totaled.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim MyTable As Table
Dim RowCounter As Integer
Dim ColumnCounter As Integer

.CreateTable False, "Default Table", 5,5
Set MyTable = .Table

For RowCounter = 1 To 3
    MyTable.CellLayout(RowCounter,0).GotoLayout
    .Text.InsertText "Sales Rep" & Format$(RowCounter)
Next
For ColumnCounter = 1 To 3
    MyTable.CellLayout(0,ColumnCounter).GotoLayout
    .Text.InsertText "Year" & Format$(ColumnCounter)
Next
For ColumnCounter = 1 To 3
    For RowCounter = 1 To 3
        MyTable.CellLayout(RowCounter,ColumnCounter).GotoLayout
        .Text.InsertText Format$(RowCounter * ColumnCounter)
    Next
Next
For ColumnCounter = 1 To 3
    MyTable.CellLayout(4,ColumnCounter).GotoLayout
    .SmartSumColumn
Next
For RowCounter = 1 To 4
    MyTable.CellLayout(RowCounter,4).GotoLayout
    .SmartSumRow
Next
```

```
'Example: SmartSumRow method
' This example creates a table with 5 rows and 5 columns into the current
' document. Headers are created for the first column and row. Numbers are
' inserted elsewhere and the column and rows are totaled.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim MyTable As Table
Dim RowCounter As Integer
Dim ColumnCounter As Integer

.CreateTable False, "Default Table", 5,5
Set MyTable = .Table

For RowCounter = 1 To 3
    MyTable.CellLayout (RowCounter,0) .GotoLayout
    .Text.InsertText "Sales Rep" & Format$(RowCounter)
Next
For ColumnCounter = 1 To 3
    MyTable.CellLayout (0,ColumnCounter) .GotoLayout
    .Text.InsertText "Year" & Format$(ColumnCounter)
Next
For ColumnCounter = 1 To 3
    For RowCounter = 1 To 3
        MyTable.CellLayout (RowCounter,ColumnCounter) .GotoLayout
        .Text.InsertText Format$(RowCounter * ColumnCounter)
    Next
Next
For ColumnCounter = 1 To 3
    MyTable.CellLayout (4,ColumnCounter) .GotoLayout
    .SmartSumColumn
Next
For RowCounter = 1 To 4
    MyTable.CellLayout (RowCounter,4) .GotoLayout
    .SmartSumRow
Next
```



'Example: SnapshotOffset property

'This example script has not yet been created.

'Example: SnapshotPath property

'This example script has not yet been created.

'Example: SnapShotSaveOptions property

'This example script has not yet been created.

'Example: SnapshotSize property

'This example script has not yet been created.

'Example: SortLevel1 property

'This example script has not yet been created.

'Example: SortLevel2 property

'This example script has not yet been created.

'Example: SortLevel3 property

'This example script has not yet been created.

'Example: SortNumbers property

'This example script has not yet been created.



'Example: SortOptions property

'This example script has not yet been created.

'Example: SortOrder property

'This example script has not yet been created.

```

'Example: SortParagraphs method
' This example inserts 5 rows and 5 columns of text into the current document
' where each column is tab delimited and then pause to display a message box.
' When you click OK, the message box closes, the sort options are set, and the
' grid is sorted in descending order relative to the first column.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

'Insert tab-delimited text grid
Dim RowCount as Integer
Dim ColumnCount as Integer
Dim OutputText as string
For RowCount = 1 To 5
    For ColumnCount = 1 To 5
        OutputText = "r" & RowCount & "c" & ColumnCount
        .Text.InsertText OutputText
        .Text.InsertTab
    Next
    .Text.SplitParagraph
Next
.Text.Shade $LwpLocationTypeParagraph,$LwpNavigateDirectionLeft,5

'Set sort options
MessageBox "Click OK to sort selected text.",MB_OK,"Example Script"
.ActiveDocument.SortOptions.SortLevel1.FieldNumber = 1
.ActiveDocument.SortOptions.SortLevel2.SortWordOption = LwpSortWhichWordAllwords
.ActiveDocument.SortOptions.FieldDelimiter = $LwpDelimiterTypeTabdelimited
.ActiveDocument.SortOptions.SortLevel1.SortOrder = $LtsSortDescending
.ActiveDocument.SortOptions.SortLevel1.SortType = $LwpSortTypeAlphanumeric

'Sort paragraphs
.Text.SortParagraphs

```

'Example: SortType property

'This example script has not yet been created.

'Example: SortWordOption property

'This example script has not yet been created.

'Example: SortWord property

'This example script has not yet been created.

'Example: SpaceAbove property

'This example script has not yet been created.

'Example: SpaceBelow property

'This example script has not yet been created.



'Example: SpacesBetweenSentences property

'This example script has not yet been created.

'Example: SpacingStyleName property  
'This example script has not yet been created.

'Example: SpacingUnitName property

'This example script has not yet been created.

'Example: SpacingUnits property

'This example script has not yet been created.

'Example: Spacing property

'This example script has not yet been created.

```
'Example: SpellAddToUserDict method
' This example prompts for a word to be added to the user dictionary. The word
' is inserted into the current document, selected and then added to the user
' dictionary.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim NewWord as String
NewWord = InputBox ("Enter a word to add to the user dictionary:", "Example Script", "")
If NewWord <> "" Then
    .Type NewWord
    .Text.MoveToStart $LwpLocationTypeWord
    .SelectWord
    .SpellAddToUserDict
    MessageBox NewWord & " was added to the user dictionary.", MB_OK, "Example Script"
End If
```

'Example: SpellCheckIncludesOtherTextStreams property  
'This example script has not yet been created.

'Example: SpellCheckInitialCaps property  
'This example script has not yet been created.



'Example: SpellCheckRepeatedWords property  
'This example script has not yet been created.

'Example: SpellCheckStartsAtBeginning property  
'This example script has not yet been created.

'Example: SpellCheckUserDictAlternatives property  
'This example script has not yet been created.

'Example: SpellCheckWordsWithNums property  
'This example script has not yet been created.

'Example: SpellClearSkippedWords method

'This example script has not yet been created.

'Example: SpellColor property

'This example script has not yet been created.

'Example: SpellFocusedColor property

'This example script has not yet been created.

```
'Example: SpellMarkSkippedWords method  
' This example marks the selected word as skipped.  
' RUNTIME DEPENDENCIES: You must have a document open and a word selected  
' for this script to work.
```

```
.SpellMarkSkippedWords
```



'Example: SpellSkipAll method

' This example inserts a word, and selects it and then

' adds the selected word to Spell Check's skip all list.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "Osborne"

.SelectWord

.SpellSkipAll

'Example: SpellWord method

'This example script has not yet been created.

## Word Pro: Shadow property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASSES;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_FONTMETRICS_SHADOW_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Turns the shadow effect on for a text object.

### Data Type

Boolean

### Syntax

[objectreference.]Shadowvalue = Shadowvalue

Shadowvalue = [objectreference.]Shadowvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

### **Word Pro: IsViewVertRuler property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWVERTRULER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewvertrulervalue = [objectreference].IsViewVertRuler

[objectreference].IsViewVertRuler = isviewvertrulervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewVertScrollBar property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWVERTSCROLLBAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewvertscrollbarvalue = [objectreference].IsViewVertScrollBar

[objectreference].IsViewVertScrollBar = isviewvertscrollbarvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

### **Word Pro: IsWordProChart property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISWORDPROCHART\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

iswordprochartvalue = [objectreference].IsWordProChart

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsWorkingDir property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISWORKINGDIR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if "Use working directory" is enabled in Word Pro Preferences.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

isworkingdirvalue = [objectreference].IsWorkingDir

[objectreference].IsWorkingDir =isworkingdirvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default value is False (0).

### **Usage**

Equivalent to "Use working directory" on the Locations panel of the Word Pro Preferences dialog box. If the value for this property is True, Word Pro continues to use the folder you chose when opening or saving a document. Word Pro uses this folder when it displays file names in the Open and Save As dialog boxes. If the value is False, Word Pro uses the default document directory.

### **Word Pro: Italic property**

{button ,AL('H\_FONT\_CLASS;H\_FONTMETRICS\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ITALIC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

italicvalue = [objectreference].Italic

[objectreference].Italic = italicvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## Word Pro: JoinCorners property

{button ,AL('H\_JOIN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_JOINCORNERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Defines which corner of a page, table layout, or frame container to apply a join type.

### Data Type

Variant (Enumerated)

JoinCornerPosition

### Syntax

joincornersvalue = [objectreference].JoinCorners

[objectreference].JoinCorners = joincornersvalue

### Legal values

\$LwpJoinCornerPositionAll (426) Allows you to apply a specific join type to all corners of a page, table layout, or frame container.

\$LwpJoinCornerPositionBottomleft (425) Allows you to apply a specific join type to the bottom left corner of a page, table layout, or frame container.

\$LwpJoinCornerPositionBottomright (424) Allows you to apply a specific join type to the bottom right corner of a page, table layout, or frame container.

\$LwpJoinCornerPositionTopleft (422) Allows you to apply a specific join type to top left corner of a page, table layout, or frame container.

\$LwpJoinCornerPositionTopright (423) Allows you to apply a specific join type to top right corner of a page, table layout, or frame container.

### Usage

## Word Pro: JoinHeight property

{button ,AL('H\_JOIN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_JOINHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Defines the height of the bounding rectangular box of a scalable join object.

### Data Type

Long

### Syntax

joinheightvalue = [objectreference].JoinHeight

[objectreference].JoinHeight = joinheightvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

A join object can be scalable or fixed.

#### Scalable join objects

You can set the ScaleMode property of a scaleable join object to scaling or no scaling. Setting the ScaleMode property to scaling for a scaleable join object causes the join object's width and height to be a function of the page, table layout, or frame container's height and width and the Percentage property. (For more information, see ScaleMode property and Percentage property.)

Setting the ScaleMode property to no scaling for a scaleable join object causes the width and height of a join object to be the same as the width and height properties of the join object. The width and height of the join object does not change as the container's width and height changes.

The width and height of a scaleable join with a no scaling setting are fixed. However, you can change the width and height properties of a join object.

#### Fixed join objects

A fixed join object has predefined Word Pro width and height properties. Therefore, you cannot change the width and height of a fixed join object.

## Word Pro: JoinType property

{button ,AL('H\_JOIN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_JOINTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The ID that defines the type or style of the join object.

### Data Type

Variant (Enumerated)

JoinType

### Syntax

jointypevalue = [objectreference].JoinType

[objectreference].JoinType = jointypevalue

### Legal values

A join type object can be scalable or fixed. For more information on scalable and fixed join type objects, see the Usage section below.

\$LwpJoinTypeDeco1 (543) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeDeco2 (544) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeDeco3 (549) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeDesktop (539) *Scalable*

\$LwpJoinTypeDogear (538) *Scalable*

\$LwpJoinTypeNone (530)

\$LwpJoinTypePin (546) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeRain (545) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeRect (534)

\$LwpJoinTypeRope (542) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeRose (547) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

\$LwpJoinTypeRounded (533) *Scalable*

\$LwpJoinTypeSunf (548) *Fixed* Word Pro by default sets the join type to no scaling. Setting a scaling property for the value causes it to malfunction.

### Usage

#### Scalable join type objects

You can set the ScaleMode property of a scalable join type object to scaling or no scaling. Setting the ScaleMode property to scaling for a scalable join type object causes the join type object's width and height to be a function of the page, table layout, or frame container's height and width and the Percentage property. (For more information, see ScaleMode property and/or Percentage property.)

Setting the ScaleMode property to no scaling for a scalable join type object causes the width and height of a join type object to be the same as the width and height properties of the join type object. The width and height of the join type object does not change as the container's width and height changes.

The width and height of a scalable join type object with a no scaling setting are fixed; however, you can change the width and height properties of a join type object.

#### Fixed join type objects

A fixed join type object has predefined Word Pro width and height properties. As a result, you cannot change the width and height of a fixed join object.



## Word Pro: JoinWidth property

{button ,AL('H\_JOIN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_JOINWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Defines the width of the bounding rectangular box of a scaleable join object.

### Data Type

Long

### Syntax

joinwidthvalue = [objectreference].JoinWidth

[objectreference].JoinWidth = joinwidthvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

Use the JoinWidth property to define the width of the bounding rectangular box of a scalable join object. A join object can be scalable or fixed.

#### Scalable join objects

You can set the ScaleMode property of a scaleable join object to scaling or no scaling. Setting the ScaleMode property to scaling for a scaleable join object causes the join object's width and height to be a function of the container's (page, table layout, or frame) height and width and the Percentage property. (For more information, see ScaleMode property and Percentage property.)

Setting the ScaleMode property to no scaling for a scaleable join object causes the width and height of a join object to be the same as the width and height properties of the join object. The width and height of the join object does not change as the container's width and height changes.

The width and height of a scaleable join with a no scaling setting are fixed; however, you can change the width and height properties of a join object.

#### Fixed join objects

A fixed join object has predefined Word Pro width and height properties. Therefore, you cannot change the width and height of a fixed join object.

## Word Pro: Justifiable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_JUSTIFIABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Expands the layout object to match the width of the parent content area, not the content itself.

## Data Type

[Integer](#)

## Syntax

justifiablevalue = [objectreference].Justifiable

[objectreference].Justifiable = justifiablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False.

## Usage

## **Word Pro: KeepWithNext property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_KEEPWITHNEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

keepwithnextvalue = [objectreference].KeepWithNext

[objectreference].KeepWithNext = keepwithnextvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: KeepWithPrev property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_KEEPWITHPREV\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

keepwithprevvalue = [objectreference].KeepWithPrev

[objectreference].KeepWithPrev = keepwithprevvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## Word Pro: KeyboardLanguage property

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_KEYBOARDLANGUAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The current text language being used by Word Pro. Word Pro uses the language specified in this property to set the language for text you type. The value of this property does not indicate the language setting for text in the focus.

### Data Type

[Integer](#)

### Syntax

keyboardlanguagevalue = [objectreference].KeyboardLanguage

[objectreference].KeyboardLanguage = keyboardlanguagevalue

### Legal values

You can change the value of this property but you must use one of the Windows 95 language codes listed in the table below. Note that this table contains the codes twice: first sorted by name on the left and then sorted by code on the right.

<u>Language Codes By Name</u>		<u>Language Codes by Number</u>	
AFRIKAANS	1078	0	SYSTEM
ALBANIAN	1052	10241	ARABIC SYRIA
AMERICAN	1033	1025	ARABIC
ARABIC EGYPT	3073	10250	SPANISH PERU
ARABIC MOROCCO	6145	1029	CZECH
ARABIC	1025	1037	HEBREW
ARABIC BAHRAIN	15361	1038	HUNGARIAN
ARABIC JORDAN	11265	1030	DANISH
ARABIC QATAR	16385	1039	ICELANDIC
ARABIC IRAQ	2049	1040	ITALIAN
ARABIC SYRIA	10241	1026	BULGARIAN
ARABIC TUNISIA	7169	1031	GERMAN
ARABIC LEBANON	12289	1032	GREEK
ARABIC LIBYA	4097	1027	CATALAN
ARABIC OMAN	8193	1033	AMERICAN
ARABIC KUWAIT	13313	1034	SPANISH
ARABIC ALGERIA	5121	1028	CHINESE TRADITIONAL
ARABIC YEMEN	9217	1035	FINNISH
ARABIC UAE	14337	1036	FRENCH
AUSTRALIAN	3081	1041	JAPANESE
BASQUE	1069	1042	KOREAN
BRAZILIAN	1046	1043	DUTCH
BRITISH	2057	1044	NORWEGIAN
BRITISHMEDIZE	35849	1045	POLISH
BRMEDICAL	34825	1046	BRAZILIAN
BULGARIAN	1026	1047	RHAETO ROMAN
BYELORUSSIAN	1059	1048	ROMANIAN
CATALAN	1027	1049	RUSSIAN

CHINESE PRCHINA	2052	1053	SWEDISH
CHINESE HONGKONG	3076	1050	CROATIAN
CHINESE SINGAPORE	4100	1051	SLOVAK
CHINESE TRADITIONAL	1028	1052	ALBANIAN
CROATIAN CYRILLIC	2074	1054	THAI
CROATIAN SERBIAN	3098	1055	TURKISH
CROATIAN	1050	1056	URDU
CZECH	1029	1057	INDONESIAN
DANISH	1030	1058	UKRAINIAN
DUTCH BELGIAN	2067	1059	BYELORUSSIAN
DUTCH	1043	1060	SLOVENE
ENGLISH CANADIAN	4105	1061	ESTONIAN
ENGLISH NEWZEALAND	5129	1062	LATVIAN
ENGLISH IRELAND	6153	1063	LITHUANIAN
ENGLISH SAFRICA	7177	1065	FARSI
ENGLISH JAMAICA	8201	1069	BASQUE
ENGLISH CARRIBEAN	9225	1070	SORBIAN
ESTONIAN	1061	1078	AFRIKAANS
FAEROESE	1080	1080	FAEROESE
FARSI	1065	11265	ARABIC JORDAN
FINNISH	1035	11274	SPANISH ARGENTINA
FRENCH BELGIAN	2060	12289	ARABIC LEBANON
FRENCH CANADIAN	3084	12298	SPANISH ECUADOR
FRENCH SWISS	4108	13313	ARABIC KUWAIT
FRENCH LUXEMBOURG	5132	13322	SPANISH CHILE
FRENCH	1036	14337	ARABIC UAE
GERMAN SWISS	2055	16394	SPANISH BOLIVIA
GERMAN AUSTRIAN	3079	14346	SPANISH URUGUAY
GERMAN	1031	15361	ARABIC SAUDI
GERMAN LIECHTENSTEIN	5127	15370	SPANISH PARAGUAY
GERMAN LUXEMBOURG	4103	16385	ARABIC QATAR
GREEK	1032	2049	ARABIC IRAQ
HEBREW	1037	2052	CHINESE PRCHINA
HUNGARIAN	1038	2055	GERMAN SWISS
ICELANDIC	1039	2057	BRITISH
INDONESIAN	1057	2058	SPANISH MEXICAN
ITALIAN SWISS	2064	2060	FRENCH BELGIAN
ITALIAN	1040	2064	ITALIAN SWISS
JAPANESE	1041	2066	KOREAN JOHAB
KOREAN	1042	2067	DUTCH BELGIAN
KOREAN JOHAB	2066	2068	NYNORSK
LATVIAN	1062	2070	PORTUGUESE
LITHUANIAN	1063	2074	CROATIAN CYRILLIC
MEDICAL	33801	3073	ARABIC EGYPT
NORWEGIAN	1044	3076	CHINESE HONGKONG

NYNORSK	2068	3079	GERMAN AUSTRIAN
POLISH	1045	3081	AUSTRALIAN
PORTUGUESE	2070	3082	SPANISH MODERN
RHAETO ROMAN	1047	3084	FRENCH CANADIAN
ROMANIAN	1048	3091	VOORKEUR
RUSSIAN	1049	3098	CROATIAN SERBIAN
RUSSIANIO	33817	33801	MEDICAL
SLOVAK	1051	33817	RUSSIANIO
SLOVENE	1060	34825	BRMEDICAL
SORBIAN	1070	35849	BRITISHMEDIZE
SPANISH ECUADOR	12298	4100	CHINESE SINGAPORE
SPANISH DOMINICAN	7178	5130	SPANISH COSTARICA
SPANISH MODERN	3082	4108	FRENCH SWISS
SPANISH ARGENTINA	11274	65535	UNIVERSAL
SPANISH PANAMA	6154	4106	SPANISH GUATEMALA
SPANISH PERU	10250	6154	SPANISH PANAMA
SPANISH VENEZUELA	8202	4103	GERMAN LUXEMBOURG
SPANISH MEXICAN	2058	7169	ARABIC TUNISIA
SPANISH CHILE	13322	4105	ENGLISH CANADIAN
SPANISH COLOMBIA	9226	6145	ARABIC MOROCCO
SPANISH BOLIVIA	16394	5121	ARABIC ALGERIA
SPANISH URUGUAY	14346	4097	ARABIC LIBYA
SPANISH COSTARICA	5130	5127	GERMAN LIECHTENSTEIN
SPANISH	1034	5129	ENGLISH NEWZEALAND
SPANISH GUATEMALA	4106	5132	FRENCH LUXEMBOURG
SPANISH PARAGUAY	15370	6153	ENGLISH IRELAND
SWEDISH	1053	7177	ENGLISH SAFRICA
SYSTEM	0	7178	SPANISH DOMINICAN
THAI	1054	8193	ARABIC OMAN
TURKISH	1055	8201	ENGLISH JAMAICA
UKRAINIAN	1058	8202	SPANISH VENEZUELA
UNIVERSAL	65535	9217	ARABIC YEMEN
URDU	1056	9225	ENGLISH CARRIBEAN
VOORKEUR	3091	9226	SPANISH COLOMBIA

## Usage

When you type text in a Word Pro document, Word Pro assigns a language to that text. Any text you type is assigned the language shown in this property. If you change the language code in this property, all text you type is assigned the new language, regardless of where you type the text. When you perform a Spell Check, Word Pro uses the text's language to select the appropriate dictionary.

Word Pro will not use this property for language settings, unless you specify this option in the Word Pro Preferences dialog box by choosing File - User Setup - Word Pro Preferences. In the Word Pro Preferences dialog box, click the General tab and specify "language sets text's language" in the Keyboard options.

## **Word Pro: Keywords property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_KEYWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The user-defined keywords associated with a document.

### **Data Type**

String

### **Syntax**

keywordvalue = [objectreference].Keywords

[objectreference].Keywords = keywordvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: LandscapeMode property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_LANDSCAPEMODE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the width of the layout object to its height and set the height of the layout object to its width.

### Data Type

Integer

### Syntax

```
landscapemodevalue = [objectreference].LandscapeMode
```

```
[objectreference].LandscapeMode = landscapemodevalue
```

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: LastChild property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTCHILD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

lastchildvalue = [objectreference].LastChild

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: LastDocPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTDOCPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The path (drive and directory) Word Pro uses if the "Use working directory" option is enabled in Word Pro Preferences.

### **Data Type**

[String](#)

### **Syntax**

lastdocpathvalue = [objectreference].LastDocPath

[objectreference].LastDocPath = lastdocpathvalue

### **Legal values**

### **Usage**

This property is only used if the value for the IsWorkingDir property is True. Contains the last path that was saved to or opened from.

### **Word Pro: LastEditorName property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTEDITORNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

lasteditornamevalue = [objectreference].LastEditorName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: LastGraphicPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTGRAPHICPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The path (drive and directory) that was last chosen when importing a picture.

### **Data Type**

[String](#)

### **Syntax**

lastgraphicpathvalue = [objectreference].LastGraphicPath

[objectreference].LastGraphicPath = lastgraphicpathvalue

### **Legal values**

### **Usage**

Equivalent to the path in the Import Picture dialog box. Word Pro always displays the last path chosen to import a picture when it opens the Import Picture dialog box.

## **Word Pro: LastGraphicType property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTGRAPHICTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The last type of graphic that Word Pro imported

### **Data Type**

[String](#)

### **Syntax**

lastgraphicstypevalue = [objectreference].LastGraphicType

[objectreference].LastGraphicType = lastgraphicstypevalue

### **Legal values**

### **Usage**

Equivalent to the "Files of type" box in the Import Picture dialog box. Although both the file type and the file extension are listed in this box, this property contains only the file type, not the file extension (for example, GIF or Windows Bitmap, not \*.GIF or \*.BMP).

## **Word Pro: LastMacroPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTMACROPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The path (drive and directory) that was last chosen when running a script or setting a startup script.

### **Data Type**

String

### **Syntax**

lastmacropathvalue = [objectreference].LastMacroPath

[objectreference].LastMacroPath = lastmacropathvalue

### **Legal values**

A valid path including drive and directory.

### **Usage**

Equivalent to the path that you use when you choose Edit - Script & Macros, and then either Run - Browse, or Set Startup Scripts - Browse. It is the path that is available in the Lotus Word Pro - Choose Script dialog box.

**Word Pro: LastName property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

lastnamevalue = [objectreference].LastName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: LastPage property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

lastpagevalue = [objectreference].LastPage

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts and Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: LastUsedDateFormula property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LASTUSEDDATEFORMULA\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The last format of a date that was inserted into a document.

### **Data Type**

String

### **Syntax**

lastuseddateformulavalue = [objectreference].LastUsedDateFormula

[objectreference].LastUsedDateFormula = lastuseddateformulavalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to the Insert Date/Time dialog box and the "Insert current date" icon. When you click the icon, Word Pro inserts today's date at the cursor, using the format that was last used to insert a date.

**Word Pro: Last property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LAST\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the ClickHere block which is last in the division (uses Tab order.)

**Data Type**

[String](#)

**Syntax**

lastvalue = [objectreference].Last

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: LayoutName property

{button ,AL(^H\_CLICKHERE\_CLASS;H\_DIVISIONINFO\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_INDEXSECTION\_CLASS;H\_OLEOBJECT\_CLASS;H\_SECTION\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LAYOUTNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The content class of the layout that is associated with the text object.

### Data Type

String

### Syntax

layoutnamevalue = [objectreference].LayoutName

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage



**Word Pro: LeaderDotDashChar property**

{button ,AL('H\_CHARACTERSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LEADERDOTDASHCHAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

leaderdotdashcharvalue = [objectreference].LeaderDotDashChar

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: LeaderDotDotChar property**

{button ,AL('H\_CHARACTERSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LEADERDOTDOTCHAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

leaderdotdotcharvalue = [objectreference].LeaderDotDotChar

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: LeaderDotType property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_USERINTERFACEPREFS_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_LEADERDOTTYPE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The type of dot used as a leader in a layout object.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

leaderdotypevalue = [objectreference].LeaderDotType

[objectreference].LeaderDotType = leaderdotypevalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpLeaderDotTypeNone (526)	Setting this value prevents leader dots from being inserted in a layout object.
\$LwpLeaderDotTypeDots (527)	Setting this value inserts dots as the type of leader dot used in a layout object.
\$LwpLeaderDotTypeDashes (528)	Setting this value inserts dashes as the type of leader dot used in a layout object.
\$LwpLeaderDotTypeUnderscores (529)	Setting this value inserts underscores as the type of leader dot used in a layout object.

## Usage

You can set the value of this property to affect individual layout objects, or you can set the value of this property to act as a default for all layout objects.

To set the type of leader dot in a particular layout object, set the value of this property for that layout object.

To set your default preference for a type of leader dot throughout Word Pro, set the value of the LeaderDotType property on the UserInterfacePrefs object. Equivalent to choosing the "Leader" box in the Set Tabs dialog box.

**Word Pro: LeaderDotUnderscoreChar property**

{button ,AL('H\_CHARACTERSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LEADERDOTUNDERSCORECHAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

leaderdotunderscorecharvalue = [objectreference].LeaderDotUnderscoreChar

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: LeadingText property**

{button ,AL(^H\_ENDNOTEDIVISIONGROUPNUM\_CLASS;H\_ENDNOTEDIVISIONNUM\_CLASS;H\_ENDNOTEDOC  
NUM\_CLASS;H\_FOOTNOTENUMBERING\_CLASS;H\_FOOTNOTENUMOPT\_CLASS';0)} [See list of classes](#)

{button ,AL(^H\_LEADINGTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Adds a string of characters before an endnote reference.

### **Data Type**

[String](#)

### **Syntax**

leadingtextvalue = [objectreference].LeadingText

[objectreference].LeadingText = leadingtextvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Assigning this property to an endnote object is equivalent to choosing Create - Footnote/Endnote, clicking Options, selecting "Enclosed by text before," and inserting text in the "Text before" box located on the Numbering panel.

## Word Pro: LeftExternalMargin property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_LEFTEXTERNALMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the width of the left external margin of a layout object.

### Data Type

Long

### Syntax

leftexternalmarginvalue = [objectreference].LeftExternalMargin

[objectreference].LeftExternalMargin = leftexternalmarginvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

This property cannot be set individually for FrameLayout objects within Word Pro. It is combined with all external margin values in the "Padding around border" setting, located on the Size & Margins panel of the InfoBox.

### Word Pro: LeftIntArea property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_LEFTINTAREA_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Not implemented.

### Data Type

Long

### Syntax

leftintareavalue = [objectreference].LeftIntArea

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## Word Pro: LeftTopCellId property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_LEFTTOPCELLID_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates the column ID of the top left cell when cells are connected in a layout object. In an unconnected cell, this property returns the current cell's column ID.

## Data Type

Integer

## Syntax

lefttopcellidvalue = [objectreference].LeftTopCellId

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

The column ID that is returned is zero based, which means that the first column in a table has a column ID value of zero.



## Word Pro: Left property

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LEFT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[StatusBar]

The left position of the status bar on the workspace.

[ApplicationWindow]

The position of the window (the origin of the window) relative to the left of the screen.

[Window]

[DocWindow]

## Data Type

Long

## Syntax

leftvalue = [objectreference].Left

[objectreference].Left = leftvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

[IconBarManager]

This property is not valid for IconBarManager.

[StatusBar]

Use this property to set the left position of the status bar on the workspace.

[ApplicationWindow]

Use this property to set the left position of the application window on the workspace.

[Window]

[DocWindow]

## **Word Pro: Length property**

{button ,AL(^H\_BAG\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LENGTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

lengthvalue = [objectreference].Length

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: Level property**

{button ,AL('H\_KINSOKU\_CLASS;H\_NUMBERING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

levelvalue = [objectreference].Level

[objectreference].Level = [objectreference].Level

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: LineLocation property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_LINELOCATION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to determine the placement of a line around or within a layout object.

## Data Type

Integer

## Syntax

linelocationvalue = [objectreference].LineLocation

[objectreference].LineLocation = linelocationvalue

## Legal values

<u>Value</u>	<u>Effect</u>
0	Setting this value places a line around the margins of a layout object.
1	Setting this value places a line in the middle of a layout object.
2	Setting this value places a line around the edge of a layout object.
3	Setting this value allows you to specify the distance of a line from the edge of a layout object.

## Usage

Equivalent to the "Line placement" setting, located on the Lines & Colors panel of the InfoBox for certain layout objects.

## Word Pro: LineMix property

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINEMIX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-Write) Allows you to return or set a variety of line styles to a table cell.

### Data Type

Variant (Enumerated)

TableMix

### Syntax

linemix value = [objectreference].LineMix

[objectreference].LineMix = Linemix value

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpTableMixAllmixed (1893)	Setting this value allows you to assign a different line style to each line of a table cell.
\$LwpTableMixBottommixed (1890)	Setting this value allows you to assign a line style to the bottom line of a table cell that is different from the line style of the top, right, and left line of a table cell.
\$LwpTableMixLeftmixed (1891)	Setting this value allows you to assign a line style to the left line of a table cell that is different from the line style of the top, right, and bottom line of a table cell.
\$LwpTableMixRightmixed (1892)	Setting this value allows you to assign a line style to the right line of a table cell that is different from the line style of the top, bottom, and left line of a table cell.
\$LwpTableMixTopmixed (1889)	Setting this value allows you to assign a line style to the top line of a table cell that is different from the line style of the bottom, right, and left line of a table cell.

### Usage

You can use this property to set more than one type of Word Pro defined styles for lines in a table cell. For example, you could set the top and bottom lines of a table cell with a double wavy pattern, and set the right and left lines of a table cell with a long dash pattern. You could also set a different line style to each line in a table cell. For example, you could set the top table cell line with a wavy pattern, bottom table cell line with a star pattern, the right table cell line with a long dash pattern, and the left table cell line with a dotted pattern.

You can use this property to return the different types of line styles assigned to a table cell.

## **Word Pro: LinePlacement property**

{button ,AL('H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINEPLACEMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer (Enumerated Bitmask)

LinePlacement

### **Syntax**

lineplacementvalue = [objectreference].LinePlacement

[objectreference].LinePlacement = lineplacementvalue

### **Legal values**

LwpLinePlacementAllsides (&HF)

LwpLinePlacementBottom (&H8)

LwpLinePlacementLeft (&H1)

LwpLinePlacementRight (&H2)

LwpLinePlacementTop (&H4)

### **Usage**

## **Word Pro: LinesSpacedEveryNthUnit property**

{button ,AL('H\_LINENUMBEROPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINESPACEDEVERYNTHUNIT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

linesspacedeverynthunitvalue = [objectreference].LinesSpacedEveryNthUnit

[objectreference].LinesSpacedEveryNthUnit = linesspacedeverynthunitvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: LineValid property**

{button ,AL('H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINEVALID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer (Enumerated Bitmask)

LinePlacement

### **Syntax**

lineinvalidvalue = [objectreference].LineValid

[objectreference].LineValid = lineinvalidvalue

### **Legal values**

LwpLinePlacementAllsides (&HF)

LwpLinePlacementBottom (&H8)

LwpLinePlacementLeft (&H1)

LwpLinePlacementRight (&H2)

LwpLinePlacementTop (&H4)

### **Usage**



**Word Pro: LinkAvailable property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINKAVAILABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

linkavailablevalue = [objectreference].LinkAvailable

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: LinkDisplayNameFileLength property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINKDISPLAYNAMEFILELENGTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Long](#)

**Syntax**

linkdisplaynamefilelengthvalue = [objectreference].LinkDisplayNameFileLength

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: LinkDisplayName property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINKDISPLAYNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

linkdisplaynamevalue = [objectreference].LinkDisplayName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: LinkedFileName property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINKEDFILENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

linkedfilenamevalue = [objectreference].LinkedFileName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: Linked property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINKED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

linkedvalue = [objectreference].Linked

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: LinkGraphic property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINKGRAPHIC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

linkgraphicvalue = [objectreference].LinkGraphic

[objectreference].LinkGraphic = linkgraphicvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: ListCount property**

{button ,AL('H\_SMARTFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LISTCOUNT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

listcountvalue = [objectreference].ListCount

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: LoadFilesMaximized property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOADFILESMAXIMIZED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if the "Load files maximized" option is enabled in Word Pro Preferences.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

loadfilesmaximizedvalue = [objectreference].LoadFilesMaximized

[objectreference].LoadFilesMaximized = loadfilesmaximizedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "Load files maximized" option on the General panel of the Word Pro Preferences dialog box. If the value is True (-1), Word Pro loads files maximized. If the value is False (0), Word Pro loads files tiled and minimized.



## **Word Pro: Location property**

{button ,AL(^H\_APPLICATION\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LOCATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A path that specifies either the current working directory for Word Pro or the location of a document.

### **Data Type**

[String](#)

### **Syntax**

locationvalue = [objectreference].Location

### **Legal values**

The value of this property cannot be set by a script.

### **Usage**

When you call this property from WPAApplication, you get the current working directory for Word Pro.

When you call this property from a TextDocument object, you get the location of that document.

If you use LotusScript to create a file without specifying a directory for that file, Word Pro will store that file in the current working directory.

Under Windows 3.1, you could see and specify a working directory in the Properties dialog box for a particular executable. Windows 95 does not offer a user interface for working directories, but the concept of a working directory is still in use. In Word Pro, the working directory is always the same as the directory in which the Word Pro executable is installed.

## **Word Pro: Locked property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOCKED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

lockedvalue = [objectreference].Locked

[objectreference].Locked = lockedvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: LockForNotesUserName property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOCKFORNOTESUSERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Not implemented.

**Data Type**

[String](#)

**Syntax**

lockfornotesusernamevalue = [objectreference].LockForNotesUserName

[objectreference].LockForNotesUserName = lockfornotesusernamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: Locks property

{button ,AL('H\_EDITOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOCKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Allows or restricts the tasks an editor can perform in a document.

### Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these constants when you want Word Pro to combine the features listed below. Use the OR operator to combine constants.

### Syntax

locksvalue = [objectreference].Locks

### Legal values

<u>Value</u>	<u>Effect</u>
LwpEditLocksNoCopyAndNoSaveas (&H4)	Prevents an editor from copying any part of the document to the Clipboard, using drag and drop on any part of the document, or using File - Save As to save the document with a new name.
LwpEditLocksNoEditNamedStyles (&H2)	Prevents an editor from modifying any named styles in the document.
LwpEditLocksNoLocks (&H0)	Allows an editor to make any changes to a document.
LwpEditLocksNoPrinting (&H8)	Prevents an editor from printing the document.
LwpEditLocksNoVersionOrReview (&H1)	Prevents an editor from creating new versions or seeing any other versions.
LwpEditLocksRevmarkOnly (&H10)	Allows an editor to edit the document. However, all edits appear as markups. The editor cannot accept or reject edits using the Review bar. For example, when the editor deletes text, it does not disappear from the document, but appears with the editor's deleted text attributes.

### Usage

You can use one or a combination of the values above to determine which tasks an editor can perform.

## **Word Pro: LowerCaseAscent property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOWERCASEASCENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Points](#)

### **Syntax**

lowercaseascentvalue = [objectreference].LowerCaseAscent

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: LowerCase property**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LOWERCASE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

lowercasevalue = [objectreference].LowerCase

[objectreference].LowerCase = lowercasevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: Hwnd property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LWPBASECTRL\_LWPCUSTOMDIALOG\_HWND\_PROPERTY\_EXSCRIPT',1)} [See example](#)  
(Read-write)

### **Data Type**

Long

### **Syntax**

hwndvalue = [objectreference].Hwnd

[objectreference].Hwnd = hwndvalue

### **Legal values**

### **Usage**

## **Word Pro: MacroName property**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACRO\_NAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

macronamevalue = [objectreference].MacroName

[objectreference].MacroName = macronamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



**Word Pro: MacroPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACROPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the default path (drive and directory) for Word Pro scripts

**Data Type**

String

**Syntax**

macropathvalue = [objectreference].MacroPath

[objectreference].MacroPath = macropathvalue

**Legal values**

A valid path including drive and directory.

**Usage**

Equivalent to the "Scripts" field on the Locations panel in the Word Pro Preferences dialog box. In the Word Pro interface, the "Scripts" field can contain multiple paths. You can use this property to clear all paths before setting the default or first script path, or you can use the property, MacroPaths, to read multiple paths entered by the user.

## **Word Pro: MacroStatus property**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACROSTATUS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Integer (Enumerated Bitmask)

MacroStatus

### **Syntax**

macrostatusvalue = [objectreference].MacroStatus

### **Legal values**

LwpMacroStatusIsPaused (&H1)

LwpMacroStatusIsQuickRecord (&H4)

LwpMacroStatusIsRecording (&H2)

LwpMacroStatusIsRunning (&H1)

### **Usage**

**Word Pro: MailRoutingPtr property**

{button ,AL('H\_MAILROUTING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAILROUTINGPTR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An internal processor that points to the document's internal data structure.

**Data Type**

[Long](#)

**Syntax**

mailroutingptrvalue = [objectreference].MailRoutingPtr

**Legal values****Usage**

This property identifies the location in memory.

## Word Pro: MaintainAspectRatio property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOOUT_CLASS;H_DROPCAPLAYOOUT_CLASS;H_ENDNOTELAYOOUT_CLASS;H_FOOTERLAYOOUT_CLASS;H_FOOTNOTELAYOOUT_CLASS;H_FRAMEGROUPLAYOOUT_CLASS;H_FRAMELAYOOUT_CLASS;H_GROUPLAYOOUT_CLASS;H_HEADERLAYOOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOOUT_CLASS;H_PAGELAYOOUT_CLASS;H_ROWGROUPLAYOOUT_CLASS;H_ROWLAYOOUT_CLASS;H_RUBYLAYOOUT_CLASS;H_SUPERTABLEGROUPLAYOOUT_CLASS;H_SUPERTABLELAYOOUT_CLASS;H_TABLEHEADINGLAYOOUT_CLASS;H_TABLELAYOOUT_CLASS;H_TOCSUPERTABLELAYOOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MAINTAINASPECTRATIO_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to specify whether an image scales proportionately within a layout object.

### Data Type

Integer

### Syntax

maintainaspectratiovalue = [objectreference].MaintainAspectRatio

[objectreference].MaintainAspectRatio = maintainaspectratiovalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Equivalent to the "Scale proportionately" setting, located on the Misc panel of the InfoBox for certain layout objects.

**Word Pro: MaintainEditor property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAINTAINEDITOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

**Data Type**

[Integer](#)

**Syntax**

[objectreference].MaintainEditor = maintaineditorvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: MarginBottom property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERBORDER_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTE LAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS ;H_NOTELAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYO UT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPÉ RTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MARGINBOTTOM_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines the distance from the border to the bottom edge of the contents in a layout object.

### Data Type

Long

### Syntax

marginbottomvalue = [objectreference].MarginBottom

[objectreference].MarginBottom = marginbottomvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

Equivalent to the "Bottom margin" setting, located on the Size & Margins panel of the InfoBox for certain layout objects.

## Word Pro: MarginLeft property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERBORDER_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTE LAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CL ASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS ;H_NOTELAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYO UT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPÉ RTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MARGINLEFT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines the distance from the border to the left edge of the contents in a layout object.

## Data Type

Long

## Syntax

marginleftvalue = [objectreference].MarginLeft

[objectreference].MarginLeft = marginleftvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

Equivalent to the "Left margin" setting, located on the Size & Margins panel of the InfoBox for certain layout objects.

## Word Pro: MarginRight property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERBORDER_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTE LAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CL ASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS ;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYO UT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPÉ RTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MARGINRIGHT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines the distance from the border to the right edge of the contents in a layout object.

## Data Type

Long

## Syntax

marginrightvalue = [objectreference].MarginRight

[objectreference].MarginRight = marginrightvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

Equivalent to the "Right margin" setting, located on the Size & Margins panel of the InfoBox for certain layout objects.



## Word Pro: MarginTop property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERBORDER_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTE LAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CL ASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS ;H_NOTELAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYO UT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPÉ RTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MARGIN_TOP_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines the distance from the border to the top edge of the contents in a layout object.

## Data Type

Long

## Syntax

margintopvalue = [objectreference].MarginTop

[objectreference].MarginTop = margintopvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

Equivalent to the "Top margin" setting, located on the Size & Margins panel of the InfoBox for certain layout objects.

**Word Pro: MarkCharacter property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MARKCHARACTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

markcharactervalue = [objectreference].MarkCharacter

[objectreference].MarkCharacter = markcharactervalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: MarkerClass property

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MARKERCLASS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Variant (Enumerated)

MarkerType

### Syntax

markerclassvalue = [objectreference].MarkerClass

[objectreference].MarkerClass = markerclassvalue

### Legal values

\$LwpMarkerTypeBookmark (590)

\$LwpMarkerTypeClickhere (593)

\$LwpMarkerTypeDde (591)

\$LwpMarkerTypeDefault (589)

\$LwpMarkerTypeField (592)

\$LwpMarkerTypeRuby (594)

### Usage

### **Word Pro: MarkerName property**

{button ,AL('H\_BOOKMARK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MARKERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The internal (hexidecimal) representation of a bookmark name. This value cannot be changed.

### **Data Type**

[String](#)

### **Syntax**

markernamevalue = [objectreference].MarkerName

### **Legal values**

### **Usage**

If you don't name the bookmark yourself, Word Pro always assigns an internal (hexidecimal) name to a bookmark. This name cannot be changed.

## **Word Pro: MarkPosition property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MARKPOSITION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MarkPosition

### **Syntax**

markpositionvalue = [objectreference].MarkPosition

[objectreference].MarkPosition = markpositionvalue

### **Legal values**

\$LwpMarkPositionBothSides (585)

\$LwpMarkPositionLeft (583)

\$LwpMarkPositionRight (584)

### **Usage**

## **Word Pro: MarkType property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MARKTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MarkType

### **Syntax**

marktypevalue = [objectreference].MarkType

[objectreference].MarkType = marktypevalue

### **Legal values**

\$LwpMarkTypeBars (587)

\$LwpMarkTypeChar (588)

\$LwpMarkTypeNone (586)

### **Usage**

## Word Pro: MasterName property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MASTERNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns the name of the layout unless the layout is part of a complex layout, in which case, it returns the name of the complex layout object.

## Data Type

String

## Syntax

masternamevalue = [objectreference].MasterName

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

A complex layout is one which uses separate layout objects for left and right pages.

## **Word Pro: MatchType property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MATCHTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to narrow a search in Find & Replace.

### **Data Type**

Variant (Enumerated)

FindMatch

### **Syntax**

matchtypevalue = [objectreference].MatchType

[objectreference].MatchType = matchtypevalue

### **Legal values**

\$LwpFindMatchOnBeginningOfWord (283) Sets the option to match words beginning with the find text.

\$LwpFindMatchOnEndingOfWord (284) Sets the option to match words ending with the find text.

\$LwpFindMatchOnWholeWord (282) Sets the option to match the find text to whole words only.

\$LwpFindMatchWithinAWord (285) Sets the option to match words containing the find text.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text, and choosing an option in the "Find & Replace" box.



### **Word Pro: MaxBottomBorder property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXBOTTOMBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the bottom border of a row in a table object.

### **Data Type**

Long

### **Syntax**

maxbottombordervalue = [objectreference].MaxBottomBorder

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: MaxBottomGutter property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXBOTTOMGUTTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the bottom gutter of a row in a table.

### **Data Type**

Long

### **Syntax**

maxbottomguttervalue = [objectreference].MaxBottomGutter

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## Word Pro: MaxContentHeight property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXCONTENTHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the maximum height the content of a container can expand and rotate.

### Data Type

Long

### Syntax

maxcontentheightvalue = [objectreference].MaxContentHeight

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## Word Pro: MaxContentWidth property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASSES;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXCONTENTWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the maximum width the content of a container can expand and rotate.

### Data Type

Long

### Syntax

maxcontentwidthvalue = [objectreference].MaxContentWidth

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## **Word Pro: MaxHorzPaneDistance property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXHORIZPANEDISTANCE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

maxhorzpanedistancevalue = [objectreference].MaxHorzPaneDistance

[objectreference].MaxHorzPaneDistance = maxhorzpanedistancevalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: MaxHyphLines property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXHYPHLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

maxhyphlinesvalue = [objectreference].MaxHyphLines

[objectreference].MaxHyphLines = maxhyphlinesvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: NumberEveryNthLine property**

{button ,AL('H\_LINENUMBEROPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBЕРЕVERYNTHLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

numbereverynthlinevalue = [objectreference].NumberEveryNthLine

[objectreference].NumberEveryNthLine = numbereverynthlinevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: NumberFound property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBERFOUND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables a count of words found that corresponded to the find text at the completion of Find & Replace.

### **Data Type**

Long

### **Syntax**

numberfoundvalue = [objectreference].NumberFound

[objectreference].NumberFound = numberfoundvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text and proceeding with Find & Replace. When finished, Word Pro displays a dialog box with a count of the words found that matched the find text.



## **Word Pro: NumberingPosition property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBERINGPOSITION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

numberingpositionvalue = [objectreference].NumberingPosition

[objectreference].NumberingPosition = numberingpositionvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumberingStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBERINGSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

numberingstylevalue = [objectreference].NumberingStyleName

[objectreference].NumberingStyleName = numberingstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumberOfDataFields property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBEROFDATAFIELDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

numberofdatafieldsvalue = [objectreference].NumberOfDataFields

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumberOfReplacements property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBEROFREPLACEMENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables a count of words that used the replace with text at the completion of Find & Replace.

### **Data Type**

[Long](#)

### **Syntax**

numberofreplacementsvalue = [objectreference].NumberOfReplacements

[objectreference].NumberOfReplacements = numberofreplacementsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text and proceeding with Find & Replace. When finished, Word Pro displays a dialog box with a count of the words that used the replace with text.

## **Word Pro: NumberOfRevisions property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBEROFREVISIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

numberofrevisionsvalue = [objectreference].NumberOfRevisions

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumberSequenceName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBERSEQUENCENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

numbersequencenamevalue = [objectreference].NumberSequenceName

[objectreference].NumberSequenceName = numbersequencenamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumberWhichLines property**

{button ,AL('H\_LINENUMBEROPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBERWHICHLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

LineNumberOpts

### **Syntax**

numberwhichlinesvalue = [objectreference].NumberWhichLines

[objectreference].NumberWhichLines = numberwhichlinesvalue

### **Legal values**

\$LwpLineNumberOptsNone (554)

\$LwpLineNumberOptsSpecifiedLines (555)

\$LwpLineNumberOptsTextLinesOnly (556)

### **Usage**

### **Word Pro: Number property**

{button ,AL('H\_FOOTNOTE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

numberpropertyvalue = [objectreference].Number

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



**Word Pro: NumCharsInDoc property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMCHARSINDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read only) Returns the number of characters in a document.

**Data Type**

Long

**Syntax**

numcharsindocvalue = [objectreference].NumCharsInDoc

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: NumCharsInParagraph property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMCHARSINPARAGRAPH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The number of characters in the current paragraph.

### **Data Type**

Long

### **Syntax**

numcharsinparagraphvalue = [objectreference].NumCharsInParagraph

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: NumColsSpannedOneCell property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_NUMCOLSSPANNEDONECELL_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The number of table columns spanned by a connected cell.

## Data Type

[Integer](#)

## Syntax

numcolsspannedonecellvalue = [objectreference].NumColsSpannedOneCell

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

In an unconnected cell, this property will contain a value of 1.

## Word Pro: NumCols property

```
{button ,AL('H_BASSETABLE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DOCWINDOW_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNOTETABLE_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GLOSSARY_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARALLELOLUMNS_CLASS;H_POWERFIELD_CLASS;H_PRINTSETTINGS_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WI  
NVIEWPREFS_CLASS',0)} See list of classes
```

```
{button ,AL('H_NUMCOLS_PROPERTY_EXSCRIPT',1)} See example
```

[Layout]

(Read-write) Allows you to control the number of columns in a layout object.

[Table]

(Read-Only) Indicates the number of columns in a table.

## Data Type

Integer

## Syntax

numcolsvalue = [objectreference].NumCols

[objectreference].NumCols = numcolsvalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

[Layout]

Equivalent to the "Number of newspaper columns" setting, located on the Columns panel of the InfoBox for certain layout objects.

### **Word Pro: NumContainers property**

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCO  
LSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMCONTAINERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of containers currently selected together.

### **Data Type**

Integer

### **Syntax**

numcontainersvalue = [objectreference].NumContainers

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumDecimalPlaces property**

{button ,AL('H\_NUMERICFORMAT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMDECIMALPLACES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

numdecimalplacesvalue = [objectreference].NumDecimalPlaces

[objectreference].NumDecimalPlaces = numdecimalplacesvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumFields property**

{button ,AL('H\_DOCINFOFIELDMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMFIELDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The number of document fields that currently exist in a document.

### **Data Type**

[Integer](#)

### **Syntax**

numfieldsvalue = [objectreference].NumFields

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: FootnoteContSep class**

Refers to a continued footnote separator.

**Base Classes**

BaseObject\FooterSepOpt

**Derived Classes**

None

**Contained by**

FootnoteOptions in the FootnoteContSep Property

**Usage**



**Word Pro: FootnoteLayoutCollection class**

A collection of footnote layouts objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the FootnoteLayouts Property

**Usage**

Use this collection to access any of the footnote layout objects in the foundry of a specific division.

## **Word Pro: FootnoteLayout class**

The layout for a footnote object.

### **Base Classes**

BaseObject\Layout\TableLayout\EndnoteLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: FootnoteNumbering class**

Refers to the numbering order for footnote objects in a division.

**Base Classes**

BaseObject\FootnoteNumOpt

**Derived Classes**

None

**Contained by**

[FootnoteOptions](#) in the [FootnoteNumbering](#) Property

**Usage**

## **Word Pro: FootnoteNumOpt class**

Refers to the numbering options for a footnote in a division.

### **Base Classes**

BaseObject

### **Derived Classes**

[EndnoteDivisionGroupNum](#)

[EndnoteDivisionNum](#)

[EndnoteDocNum](#)

[FootnoteNumbering](#)

### **Contained by**

### **Usage**

**Word Pro: FootnoteOptions class**

Refers to the footnote options displayed in the Footnote dialog box.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

Division in the FootnoteOptions Property

TextDocument in the FootnoteOptions Property

**Usage**

**Word Pro: FootnoteSeparator class**

Refers to the line that separates the footnote from the body text in the document.

**Base Classes**

BaseObject\FooterSepOpt

**Derived Classes**

None

**Contained by**

FootnoteOptions in the FootnoteSeparator Property

**Usage**

**Word Pro: FootnoteSepOpt class**

Refers to the footnote separator options displayed in the Separator panel in the Footnote and Endnote Options dialog box.

**Base Classes**

BaseObject

**Derived Classes**

FootnoteContSep

FootnoteSeparator

**Contained by****Usage**

A footnote separator is the line that separates the footnote from the body text.

**Word Pro: FootnoteTable class**

A footnote in a table object.

**Base Classes**

BaseObject\Content\BaseTable

**Derived Classes**

None

**Contained by****Usage**



**Word Pro: FormatCheckPref class**

Refers to the format check options displayed in the Format Check Options dialog box which is accessed from the Format Check bar.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

WPApplication in the FormatCheckPreferences Property

**Usage**

You can check your document for consistent use of spacing between sentences, correct bulleted lists, and the appearance of acronyms in a paragraph. The format check also replaces incorrect characters and common typing mistakes.

## **Word Pro: FormatPreferences class**

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

WPApplication in the Format Property

### **Usage**

## **Word Pro: Formula class**

The content of a table cell. Whenever a table cell contains text, numbers, or formulas, Word Pro sees the cell contents as a Formula object. If a table cell contains a graphic, Word Pro sees the graphic as a Graphic object.

### **Base Classes**

BaseObject\Content

### **Derived Classes**

None

### **Contained by**

### **Usage**

The Content property in a CellLayout object is defined with the Variant data type so that Word Pro can store text or graphics in a cell. Word Pro treats all text as a Formula object and stores the Formula object in the Content property of the CellLayout object. The Content property always holds either a Formula object or a Graphic object. To access a Formula object in a table cell, use the following statement:

```
[objectreference.]CellLayout.Content[.memberreference]
```

Since the Content property contains the Formula object, you would add a Formula class member name to the end of this statement to achieve your desired result. With the members of the Formula class, you can create or change formulas to add, subtract, multiply, and use sums and percents for numbers in a table. You can also set the alignment, font, and other properties of the Formula object.

## Word Pro: Foundry class

A place in which Word Pro creates, stores, and provides access to other Word Pro objects. Word Pro maintains several Foundry objects and each one has a special use as described below under Usage. However, in most cases, you will use the `Division.Foundry` object to access objects in a document and the `WPAApplication.TempFoundry` object to create, store, and move objects from one `Division.Foundry` to another.

### Base Classes

`BaseObject`

### Derived Classes

None

### Contained by

`Division` in the `Foundry` Property

`WPAApplication` in the `TempFoundry` Property

`WPAApplication` in the `Foundry` Property

`WPAApplication` in the `AppFoundry` Property

### Usage

#### WPAApplication.AppFoundry

`AppFoundry` is a property on the `WPAApplication` object (always stored in the `CurrentApplication` variable). It contains a Foundry object which Word Pro uses as the Clipboard. This is the same Clipboard used when you copy or cut items in a Word Pro document. When you cut or copy a selection, Word Pro takes all the objects from your selection and places them in their respective collection objects in the Foundry object stored in the `AppFoundry` property.

For example, if you select some text and a table and choose Edit - Copy, Word Pro places all the objects which comprise that text and table into their respective collection objects in `AppFoundry`. This means that all the Layout objects are stored in the corresponding layout collection objects. All `CharacterStyle` objects are stored in the `CharacterStyleCollection` object. All `CellEngine` objects are stored in the `CellCollection` object. Text objects are stored in the `TextCollection` object. When you choose Edit - Paste, all of these objects are reassembled in their original form and displayed in the document at the insertion point.

Because Word Pro uses the Foundry object in `AppFoundry` as its Clipboard, you must exercise caution when working with `AppFoundry`. Any objects you place in `AppFoundry` will be included in the next Paste operation. Any objects you remove from `AppFoundry` will be excluded from the next Paste operation, and may adversely affect your ability to paste from the Clipboard.

You can get an object from `AppFoundry` and store it in a variable using the following statement:

```
myobject = CurrentApplication.AppFoundry.collectionpropertyname(itemreference)
```

In this statement, *myobject* is the variable in which you want to store the object; `CurrentApplication` is a global variable that always contains the `WPAApplication` object; *collectionpropertyname* is the name of the property that contains the collection object where the object you want is stored; *itemreference* is the index that specifies the object you want. For more information about collection classes, see [Overview: Word Pro LotusScript Collection Classes](#).

**Note** While you may retrieve objects from `AppFoundry`, you should not use LotusScript to place objects in the `AppFoundry` collections. This could interfere with normal user operations, such as Cut and Copy. When creating and storing your own Word Pro objects, use the Foundry object in the `TempFoundry` property.

#### WPAApplication.TempFoundry

`TempFoundry` is a property on `WPAApplication`. `TempFoundry` contains a Foundry object which Word Pro uses to temporarily store objects that are part of a Drag and Drop operation. You can use `TempFoundry` in much the same way. You can use the collection objects in `TempFoundry` as a staging area for any Word Pro LotusScript objects you create and manipulate.

For example, when you want to move an object or objects from one document to another, you can store those objects temporarily in the `TempFoundry` collection objects. The `TempFoundry` property is always available, regardless of which document is active, so you always have access to the contents of its collections. This makes it an ideal place for temporarily storing items that you want to use or move.

**Note** You must [clear](#) `TempFoundry` after each use. Any objects left in any of `TempFoundry`'s collections can reappear during Drag and Drop and other operations, and result in unpredictable behavior.

#### Division.Foundry

In addition to AppFoundry and TempFoundry, Word Pro maintains one Foundry object for each Division object. These Division foundries are stored in the Foundry property on each Division object. The Foundry object in a division provides access to all the objects in that division, including Layouts, Text, Graphics, Markers, Tables, Footnotes, and so on. As seen in the AppFoundry example above, you can access all the objects in a division through the appropriate collection in the Division foundry.

You can access objects in any division's Foundry property by going through the division collection object stored in the Divisions property on WPAApplication, as follows:

```
myobject =  
CurrentApplication.Divisions(DivisionName).Foundry.collectionpropertyname  
(itemreference)
```

In this example, *DivisionName* is the name of the division as seen in the Division tab; *collectionpropertyname* is the name of the Foundry object property in which the collection object you seek is stored; *itemreference* is a name or reference to a specific object in the collection.

### **WPAApplication.Foundry**

The Foundry property in WordPro provides a shortcut to the currently active division's Foundry object. The Foundry object in WordPro.Foundry changes as the focus changes from one Division object to another.

For example, if you had a document with one division named "Overview" and another division named "Summary," the contents of the Foundry property on WordPro would change as you moved the focus from Overview to Summary. While the focus was on the Overview division, this property would contain the Foundry object for the Division object named Overview. When the focus changed to the Summary division, this property would also change to contain the Foundry object for the Division object named Summary.

To access a collection object in the currently active division's Foundry, you can use the following statement:

```
myobject = CurrentApplication.Foundry.collectionpropertyname(itemreference)
```

### **TextDocument.Foundry**

The Foundry object stored in TextDocument.Foundry is not used by Word Pro. Do not use this Foundry object in your scripts.

## **Word Pro: FrameContainer class**

The container object for frames. This object only exists for one frame at a time and only when there is a frame within the focus. When a FrameContainer object is present, it is stored in the Frame property on the WPApplication object.

### **Base Classes**

BaseObject\BaseContainer

### **Derived Classes**

[NoteContainer](#)

[RubyContainer](#)

### **Contained by**

[WPApplication](#) in the [Frame](#) Property

### **Usage**

The primary use for a FrameContainer object is to provide quick and easy access to the FrameLayout object for the currently active frame. A FrameContainer object always represents the frame that currently has the focus. Thus, if you assign a FrameContainer object to a variable, you can use that variable to access the currently active frame. However, you must remember that the frame referenced by the variable will change as the focus moves from one frame to another. This is because the variable references the FrameContainer object, and the FrameContainer object always represents the frame that has the focus. If there is no frame within the focus, there is no FrameContainer object. Thus, a variable that stores a FrameContainer object will have a null value whenever the focus does not contain a frame. There is never more than one FrameContainer object at any time.

For more information about container objects, see [BaseContainer](#).

## **Word Pro: FrameGroupLayout class**

The layout for a frame group.

### **Base Classes**

BaseObject\Layout\FrameLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: FrameLayoutCollection class**

A collection of frame layout objects in the foundry of a specific division.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

Foundry in the Frames Property

Foundry in the FrameStyles Property

### **Usage**

Use this collection to access any of the frame layout objects in the foundry of a specific division.



## **Word Pro: FrameLayout class**

A frame layout for a frame object. This class inherits most of its members from the Layout class.

### **Base Classes**

BaseObject\Layout

### **Derived Classes**

FrameGroupLayout

### **Contained by**

### **Usage**

The FrameLayout class provides you with a way to access and modify the format and appearance of FrameLayout objects within your document.

Since the FrameLayout class is derived from the Layout class, FrameLayout objects can be stored within properties of the Layout type. For instance, the Layout property within the FrameContainer class is of the Layout type. However, this property often stores objects of the FrameLayout type. The Layout property is implemented in this way so that objects of other derived layout class types can be stored there as well. The Layout property within the FrameContainer class, for instance, may also contain objects of the NoteLayout type.

FrameLayout objects within a division are stored together in a collection. You can use the collection to access all FrameLayout objects in the collection, or you can reference a particular FrameLayout object in the collection. For example, by using the FrameLayouts collection, you could modify each FrameLayout object in the collection to be of a certain height. For more information on how to work with collections, see [Overview: Word Pro LotusScript Collection Classes](#).

At many locations within your document, multiple layouts are available. For instance, your cursor may be within a frame within a page. In this case, the frame and the page both have associated layout objects. These layout objects may be combined with other objects into related groups known as containers. For more information on containers and their associated layouts, see the help topic titled [Word Pro: BaseContainer class](#).

**Word Pro: GlossaryCollection class**

A collection of glossary objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Glossarys Property

**Usage**

Use this collection to access any of the glossary objects in the foundry of a specific division.

**Word Pro: Glossary class**

A glossary that stores frequently used information that you can insert in a document.

**Base Classes**

BaseObject\Content\BaseTable\ParallelColumns

**Derived Classes**

None

**Contained by****Usage**

A glossary stores frequently used information, such as text, tables, frames, and so on. You can then insert the information in a document.

## **Word Pro: Grammar class**

A grammatical proofing tool that analyzes a document for possible errors, and supplies suggestions and examples for incorrect sentences.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

UserInterfacePrefs in the GrammarOptions Property

### **Usage**

You can proofread and edit a document for grammar, style, and mechanics. You can also display document and readability statistics for every document you check. Equivalent to choosing Edit - Check Grammar, and then using the default grammar options or customizing grammar options for proofing the document.

The Grammar class uses many Apply properties, some of which will only be found in specific language versions of Word Pro.

**Word Pro: GraphicCollection class**

A collection of graphic objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Graphics Property

**Usage**

Use this collection to access any of the graphic objects in the foundry of a specific division.

**Word Pro: GraphicOleObjectCollection class**

A collection of graphic and OLE objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by****Usage**

Use this collection to access any of the graphic and OLE objects in the foundry of a specific division.

## **Word Pro: GraphicOleObject class**

An OLE graphic object in a document; the virtual base class for a graphic and OLE object.

### **Base Classes**

BaseObject\Content

### **Derived Classes**

Graphic

OleObject

### **Contained by**

WPApplication in the GraphicOleObject Property

### **Usage**

**Word Pro: Graphic class**

A graphic object in a document.

**Base Classes**

BaseObject\Content\GraphicOleObject

**Derived Classes**

None

**Contained by**

WPAApplication in the Graphic Property

**Usage**



**Word Pro: GroupLayoutCollection class**

A collection of group layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Groups Property

**Usage**

Use this collection to access any of the group layout objects in the foundry of a specific division.

## **Word Pro: GroupLayout class**

The layout for a group of objects.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: Gutter class**

A line (border) in the middle of the gap defined by the ColumnGap property.

**Base Classes**

BaseObject\BorderLines

**Derived Classes**

None

**Contained by**

Layout in the Gutter Property

**Usage**

**Word Pro: HeaderLayoutCollection class**

A collection of header layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Headers Property

**Usage**

Use this collection to access any of the header layout objects in the foundry of a specific division.

## **Word Pro: HeaderLayout class**

The layout for a header object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: HyphenationOptions class**

Hyphenation options displayed in the Options panel of the Document Properties dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

DivisionOptions in the HyphenationOptions Property

### **Usage**

**Word Pro: IconBarCollection class**

A collection of icon bar objects in the IconBarManager class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

[IconBarManager](#) in the [IconBars](#) Property

**Usage**

Use this collection to access any of the icon bar objects in the IconBarManager class.

## **Word Pro: IconBarManager class**

A tool for managing icon bar objects in the application. Keeps and manages the list of icon bar objects in the document. You must go through the IconBarManager before using the IconBar property.

### **Base Classes**

BaseObjectWindow

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the IconBarManager Property

### **Usage**

Used in conjunction with IconBar and IconBarCollection objects. You can use the IconBarManager to select, find, add, or remove icon bar objects. Word Pro keeps a list of icon bar sets in the IconBarCollection object.



## **Word Pro: IconBar class**

A bar containing a set of icons (small symbols) that represent shortcuts for Word Pro functions, commands, and scripts.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**

You can use icon objects to represent shortcuts and menu IDs, and add, edit, and change their display. Although you cannot change Word Pro's standard icons, you can create new icon objects and add macros to them.

## **Word Pro: Indent class**

The indentation of text from the right or left margins.

### **Base Classes**

BaseObject

### **Derived Classes**

RelativeIndent

### **Contained by**

ClickHere in the Indent Property

Formula in the Indent Property

ParagraphStyle in the Indent Property

Text in the Indent Property

TextMarker in the Indent Property

### **Usage**

**Word Pro: IndexSection class**

A section that contains the index for a document.

**Base Classes**

BaseObject\Section

**Derived Classes**

None

**Contained by****Usage**

## **Word Pro: Index class**

An index for an entire document, for a section, division, or selected text in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: Join class**

The rectangular bounding area that connects lines surrounding a frame, page, or table layout.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

Layout in the Join Property

**Usage**

A join connects line styles at each corner of a page, frame, or table. Page, frame, and table layouts all contain join and line objects. Each line object has a set of join objects it can use. For a list of join objects and the corresponding line objects, see JoinType property.

## **Word Pro: Kinsoku class**

The page layout object for the Asian version of Word Pro.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [Kinsoku](#) Property

[ParagraphStyle](#) in the [Kinsoku](#) Property

[Text](#) in the [Kinsoku](#) Property

[TextMarker](#) in the [Kinsoku](#) Property

### **Usage**

## Word Pro: Language class

### Base Classes

BaseObject

### Derived Classes

None

### Contained by

[CharacterStyle](#) in the [Language](#) Property

[ClickHere](#) in the [Language](#) Property

[DivisionOptions](#) in the [Language](#) Property

[FindAndReplace](#) in the [ReplaceLanguage](#) Property

[FindAndReplace](#) in the [SearchLanguage](#) Property

[FormatPreferences](#) in the [Language](#) Property

[ParagraphStyle](#) in the [Language](#) Property

[Text](#) in the [Language](#) Property

[TextMarker](#) in the [Language](#) Property

### Usage

**Word Pro: LayoutCollection class**

A collection of layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Layouts Property

Layout in the ChildLayouts Property

**Usage**

Use this collection to access any of the layout objects in the foundry of a specific division.



## **Word Pro: LayoutOverride class**

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

DivisionInfo in the LayoutOverride Property

### **Usage**

## Word Pro: Layout class

An abstract class which defines properties and methods that are common to all Word Pro layout objects. An explanation of layout objects is provided below under Usage. This information applies to all layout objects in Word Pro, but each layout object may exhibit minor differences. These differences are noted in the documentation of each specific layout class.

### Base Classes

[BaseObject](#)

### Derived Classes

[CellGroupLayout](#)

[CellLayout](#)

[ConnectedLayout](#)

[EndnoteLayout](#)

[FooterLayout](#)

[FootnoteLayout](#)

[FrameGroupLayout](#)

[FrameLayout](#)

[GroupLayout](#)

[HeaderLayout](#)

[NoteLayout](#)

[PageLayout](#)

[RowLayout](#)

[RubyLayout](#)

[SuperTableGroupLayout](#)

[SuperTableLayout](#)

[TableHeadingLayout](#)

[TableLayout](#)

[TOCSuperTableLayout](#)

### Contained by

[BaseContainer](#) in the [Layout](#) Property

[BaseTable](#) in the [Layout](#) Property

[BaseTable](#) in the [CurrentColumn](#) Property

[Layout](#) in the [Footer](#) Property

[Layout](#) in the [Style](#) Property

[Layout](#) in the [RightPage](#) Property

[Layout](#) in the [Header](#) Property

[Layout](#) in the [LeftPage](#) Property

[Marker](#) in the [Layout](#) Property

[WPApplication](#) in the [Layout](#) Property

[WPApplication](#) in the [CurrentColumn](#) Property

### Usage

The Layout class provides the foundation of formatting attributes for a number of derived classes. An object created from any of these derived layout classes is known as a layout object. A layout object gets its properties, methods, and events from its derived layout class.

For example, a frame is one type of layout object. It gets its properties, methods, and events from the FrameLayout class. Other layout objects include: Cells (CellLayout), Endnotes (EndnoteLayout), Footers (FooterLayout), Footnotes (FootnoteLayout), Headers (Header Layout), Comment Notes (NoteLayout), Pages (PageLayout), Rows

(RowLayout), and Tables (TableLayout). These derived classes allow you to access and modify physical appearance attributes of these layout objects.

### **Default layouts and the Style property**

Each time you create one of these layout objects, Word Pro gets that object's property values from a default layout object. A default layout object is represented by a style in Word Pro. For example, when you look at the Style panel in the InfoBox, you will see a Default Table style for table objects, a Default Frame style for frame objects, and so on. A layout object's default property values are always accessible through the Style property of that layout object.

### **Layout objects and collections**

Layout objects which are created from the same derived layout class are stored together in collections. You can use these collections to access all layout objects in the collection, or you can reference a particular layout object in the collection. For example, by using the FrameLayouts collection, you could modify all FrameLayout objects in the collection to be of a certain height. For more information on working with collections, see [Overview: Word Pro LotusScript Collection Classes](#)

### **Layout objects and containers**

At many locations within your document, the current focus will include multiple layouts. For example, your cursor may be in a cell in a table on a page. In this case, the cell, the table, and the page all have associated layout objects. These layout objects may be combined with other objects into related groups, known as containers. For more information on containers and their associated layouts, see the Help topic titled [Word Pro: BaseContainer class](#).

## **Word Pro: LineNumberOptions class**

The line number options displayed in the Line Numbering dialog box in the Options panel of the Document Properties dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the LineNumberOptions Property

TextDocument in the LineNumberOptions Property

### **Usage**

**Word Pro: LongCollection class**

A collection of Long values. This collection is currently not being used in the object model.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by****Usage**

**Word Pro: LWPTimer class**

Allows you to schedule a specific event every x number of seconds.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by****Usage**

**Word Pro: Macro class**

An object that automates tasks in the application.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

ApplicationWindow in the Macro Property

**Usage**

**Word Pro: MailRouting class**

Directs the distribution of e-mail messages in Word Pro. Each document has an internal data structure to which the MailRouting Class points.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

TextDocument in the MailRouting Property

**Usage**

Contains the name of the route and the names of people in the route. The Word Pro document must be open for MailRouting to function. Values are 0 when starting a new route and 1 when editing a route.



**Word Pro: MarkerCollection class**

A collection of marker objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Markers Property

**Usage**

Use this collection to access any of the marker objects in the foundry of a specific division.

## **Word Pro: Marker class**

Hidden objects used to attach some data or functionality to a specific location in the document.

### **Base Classes**

BaseObject

### **Derived Classes**

[ClickHere](#)

[PowerField](#)

[RubyMarker](#)

[TableMarker](#)

[TextMarker](#)

### **Contained by**

### **Usage**

**Word Pro: MenuItemCollection class**

A collection of menu item objects in the MenuItem class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

MenuItem in the Items Property

**Usage**

Use this collection to access any of the menu item objects in the MenuItem class.

## **Word Pro: MenuItem class**

Any menu or item on a menu.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the LwpMenuBar Property

ApplicationWindow in the RightMouseMenus Property

ApplicationWindow in the FreeMenus Property

### **Usage**

Allows you to return or set your own menu items. You can get the current values of LWP menu items but you cannot change them. To get control over LWP menu items, you must create your own menu item and replace the LWP menu item with your own. The menu items you create can cause a specific script function to execute, or emulate some other predefined Word Pro menu ID when the item is selected. For more information, see the Action property or the NewItem method.

## **Word Pro: MergeOptions class**

The merge options displayed in the Merge Assistant dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

TextDocument in the MergeOptions Property

### **Usage**

## **Word Pro: NoteContainer class**

The container object for notes. This object only exists for one note at a time and only when there is a note within the focus.

### **Base Classes**

BaseObject\BaseContainer\FrameContainer

### **Derived Classes**

None

### **Contained by**

Not contained in a property of any object.

### **Usage**

The primary use for a NoteContainer object is to provide quick and easy access to the NoteLayout object for the currently active note. A NoteContainer object always represents the note that currently has the focus. Thus, if you assign a NoteContainer object to a variable, you can use that variable to access the currently active note. However, you must remember that the note referenced by the variable will change as the focus moves from one note to another. This is because the variable references the NoteContainer object, and the NoteContainer object always represents the note that has the focus. If there is no note within the focus, there is no NoteContainer object. Thus, a variable that stores a NoteContainer object will have a null value whenever the focus does not contain a note. There is never more than one NoteContainer object at any given time.

For more information about container objects, see [BaseContainer](#).

**Word Pro: NoteLayoutCollection class**

A collection of note layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the NoteLayouts Property

**Usage**

Use this collection to access any of the note layout objects in the foundry of a specific division.

## **Word Pro: NoteLayout class**

The layout for a note object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**



## **Word Pro: Numbering class**

Numbering pages in a document or numbering lines of text or blank lines on a page in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [Numbering](#) Property

[ParagraphStyle](#) in the [Numbering](#) Property

[Text](#) in the [Numbering](#) Property

[TextMarker](#) in the [Numbering](#) Property

### **Usage**

## **Word Pro: NumericFormatSubset class**

A secondary format for numbers in a table cell object.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

NumericFormat in the AnyNumber Property

NumericFormat in the Negative Property

NumericFormat in the Zero Property

### **Usage**

**Word Pro: NumericFormat class**

The format for numbers used in a table cell object.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

Layout in the NumericFormat Property

**Usage**

**Word Pro: OleObjectCollection class**

A collection of Ole objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the OleObjects Property

**Usage**

Use this collection to access any of the Ole objects in the foundry of a specific division.

**Word Pro: OleObject class**

An OLE object within the application.

**Base Classes**

BaseObject\Content\GraphicOleObject

**Derived Classes**

None

**Contained by**

WPAApplication in the OleObject Property

**Usage**

## **Word Pro: Options class**

The options displayed in a dialog box in a division or a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the DocOptions Property

TextDocument in the Options Property

TextDocument in the DocOptions Property

### **Usage**

## **Word Pro: OutlineSeqCollection class**

A collection of OutlineStyleSequence objects. The scope of this collection is usually limited to a single Division object.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

Foundry in the OutlineStyleSequences Property

### **Usage**

## **Word Pro: OutlineSeqItemCollection class**

A collection of OutSeqItem objects. The scope of this collection is usually limited to a single TextDocument object.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

OutlineStyleSequence in the OutlineSeqItems Property

### **Usage**



## **Word Pro: FootnoteNumOpt class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

LeadingText AS String

Name AS String

Parent AS BaseObject

ResetWhen AS ResetOption

StartingNumber AS Integer

TrailingText AS String

UseSuperscriptReferenceNum AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: FootnoteOptions class members**

### **Properties**

Application AS WPAApplication

ContinuedFromAlignment AS AlignmentType

ContinuedFromMessage AS String

ContinuedOnAlignment AS AlignmentType

ContinuedOnMessage AS String

Description AS String

EndnoteDivisionGroupNum AS EndnoteDivisionGroupNum

EndnoteDivisionNum AS EndnoteDivisionNum

EndnoteDocNum AS EndnoteDocNum

FootnoteContSep AS FootnoteContSep

FootnoteNumbering AS FootnoteNumbering

FootnoteSeparator AS FootnoteSeparator

IsContinuedFrom AS Integer (Boolean)

IsContinuedOn AS Integer (Boolean)

IsRepeat AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: FootnoteSeparator class members**

### **Properties**

Application AS WPAApplication

BorderLines AS BorderLines

CustomLength AS Long (measured in Twips)

Description AS String

IndentFromLeft AS Long (measured in Twips)

IsFixedLength AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

SpaceAbove AS Long (measured in Twips)

SpaceBelow AS Long (measured in Twips)

UseSeparatorLine AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: FootnoteSepOpt class members**

### **Properties**

Application AS WPAApplication

BorderLines AS BorderLines

CustomLength AS Long (measured in Twips)

Description AS String

IndentFromLeft AS Long (measured in Twips)

IsFixedLength AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

SpaceAbove AS Long (measured in Twips)

SpaceBelow AS Long (measured in Twips)

UseSeparatorLine AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## Word Pro: FootnoteTable class members

### Properties

Application AS WPAApplication  
CanEmbed AS Integer (Boolean)  
CellLayouts AS StringCollection  
ClassName AS String  
ColumnLayouts AS StringCollection  
ContentLayouts AS StringCollection  
ContentType AS ContentType  
CurrentCell AS CellLayout  
CurrentColumn AS Layout  
CurrentRow AS RowLayout  
DefCellStyleName AS String  
DefColWidth AS Long (measured in Twips)  
DefRowHeight AS Long (measured in Twips)  
Description AS String  
EndingColOfSelection AS Integer  
EndingRowOfSelection AS Integer  
IsAutoGrow AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsParagraphNumberingDown AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsResetParagraphNumber AS Integer (Boolean)  
IsSizingViaMouse AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxBottomBorder AS Long (measured in Twips)  
MaxBottomGutter AS Long (measured in Twips)  
MaxLeftBorder AS Long (measured in Twips)  
MaxLeftGutter AS Long (measured in Twips)  
MaxNumColsAllowed AS Integer  
MaxNumRowsAllowed AS Integer  
MaxRightBorder AS Long (measured in Twips)  
MaxRightGutter AS Long (measured in Twips)  
MaxSplitCols AS Integer  
MaxSplitRows AS Integer  
MaxTopBorder AS Long (measured in Twips)  
MaxTopGutter AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
Parent AS BaseObject  
RowLayouts AS StringCollection  
SelectionType AS SelectionType  
SingleCellSelected AS Integer (Boolean)  
StartingColOfSelection AS Integer

StartingColStringOfSelection AS String

StartingRowOfSelection AS Integer

TableFill AS TableFill

TableLine AS TableLine

VersionID AS Long

### **Methods**

CellLayout

Connect

Copy

DeleteTable

DisconnectCells

DoesMarkerNameMatch

FindCellLayout

GetMarkerName

GoToTableCell

InsertRowOrColumn

Mark

NextItem

PreviousItem

SelectTableItem

Split

### **Events**

None

## **Word Pro: FormatCheckPref class members**

### **Properties**

Application AS WPAApplication

Description AS String

FixAcronymns AS Integer (Boolean)

FixBullets AS Integer (Boolean)

FixMargins AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Options AS FormatCheckOptions

Parent AS BaseObject

Replacements AS ReplacementChoices

Typos AS TypoChoices

UseConsistentSpaceBetweenSentences

AS Integer (Boolean)

UseTwoSpacesBetweenSentences AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: FormatPreferences class members**

### **Properties**

Application AS WPAApplication

Attributes AS Attributes

Description AS String

FastFormatType AS FastFormatType

Font AS Font

IsValid AS Integer (Boolean)

Language AS Language

Name AS String

Parent AS BaseObject

StyleName AS String

VersionID AS Long

### **Methods**

None

### **Events**

None



## **Word Pro: Formula class members**

### **Properties**

Alignment AS Alignment

Application AS WPApplication

CanEmbed AS Integer (Boolean)

ClassName AS String

ContentType AS ContentType

Description AS String

Font AS Font

Formula AS String

Indent AS Indent

IsEmpty AS Integer (Boolean)

IsReplaceable AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

ParagraphStyle AS ParagraphStyle

Parent AS BaseObject

Spacing AS Spacing

StyleName AS String

VersionID AS Long

### **Methods**

EmbedFormula

Mark

### **Events**

None

## Word Pro: Foundry class members

### Properties

[Application](#) AS [WPApplication](#)  
[Bags](#) AS [BagCollection](#)  
[CellEngines](#) AS [CellCollection](#)  
[CellLayouts](#) AS [CellLayoutCollection](#)  
[CellLayoutStyles](#) AS [CellLayoutCollection](#)  
[CharacterStyles](#) AS [CharacterStyleCollection](#)  
[ClickHeres](#) AS [ClickHereCollection](#)  
[ConnectedLayouts](#) AS [ConnectedLayoutCollection](#)  
[Contents](#) AS [ContentCollection](#)  
[ContinuedFromStory](#) AS String  
[ContinuedOnStory](#) AS String  
[Description](#) AS String  
[DropCaps](#) AS [DropCapLayoutCollection](#)  
[DropCapStyles](#) AS [DropCapLayoutCollection](#)  
[Endnotes](#) AS [EndnoteLayoutCollection](#)  
[Enumeration](#) AS EnumScope  
[Footers](#) AS [FooterLayoutCollection](#)  
[FootnoteLayouts](#) AS [FootnoteLayoutCollection](#)  
[Footnotes](#) AS [FootnoteCollection](#)  
[Frames](#) AS [FrameLayoutCollection](#)  
[FrameStyles](#) AS [FrameLayoutCollection](#)  
[Glossarys](#) AS [GlossaryCollection](#)  
[Graphics](#) AS [GraphicCollection](#)  
[Groups](#) AS [GroupLayoutCollection](#)  
[Headers](#) AS [HeaderLayoutCollection](#)  
[IsContents](#) AS Integer (Boolean)  
[IsUndoOn](#) AS Integer (Boolean)  
[IsValid](#) AS Integer (Boolean)  
[Layouts](#) AS [LayoutCollection](#)  
[Markers](#) AS [MarkerCollection](#)  
[Name](#) AS String  
[NoteLayouts](#) AS [NoteLayoutCollection](#)  
[OleObjects](#) AS [OleObjectCollection](#)  
[OutlineStyleSequences](#) AS [OutlineSeqCollection](#)  
[Pages](#) AS [PageLayoutCollection](#)  
[PageStyles](#) AS [PageLayoutCollection](#)  
[ParagraphStyles](#) AS [ParagraphStyleCollection](#)  
[ParallelColumns](#) AS [ParallelColsCollection](#)  
[Parent](#) AS [BaseObject](#)  
[PowerFields](#) AS [PowerFieldCollection](#)  
[Rows](#) AS [RowLayoutCollection](#)  
[RubyLayouts](#) AS [RubyLayoutCollection](#)  
[Sections](#) AS [SectionCollection](#)  
[SilverBullets](#) AS [SilverBulletCollection](#)

[SuperTableLayouts](#) AS [SuperTableLayoutCollection](#)  
[SuperTables](#) AS [SuperTableCollection](#)  
[TableHeadingLayouts](#) AS [TableHeadingLayoutCollection](#)  
[TableHeadings](#) AS [TableHeadingCollection](#)  
[TableLayouts](#) AS [TableLayoutCollection](#)  
[TableMarkers](#) AS [TableMarkerCollection](#)  
[Tables](#) AS [TableOnlyCollection](#)  
[TableStyles](#) AS [TableLayoutCollection](#)  
[TextMarkers](#) AS [TextMarkerCollection](#)  
[Texts](#) AS [TextCollection](#)  
[TextStyles](#) AS [TextStyleCollection](#)  
[VersionID](#) AS Long

**Methods**

[Clear](#)  
[Copy](#)  
[Create](#)  
[Paste](#)  
[Purge](#)  
[Release](#)

**Events**

None

## **Word Pro: FrameContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## Word Pro: FrameGroupLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer



UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: FrameLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: FrameLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp



## **Word Pro: GlossaryCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: Glossary class members

### Properties

Application AS WPAApplication  
CanEmbed AS Integer (Boolean)  
CellLayouts AS StringCollection  
ClassName AS String  
ColumnLayouts AS StringCollection  
ContentLayouts AS StringCollection  
ContentType AS ContentType  
CurrentCell AS CellLayout  
CurrentColumn AS Layout  
CurrentRow AS RowLayout  
DefaultLeftColumnStyleName AS String  
DefaultRightColumnStyleName AS String  
DefCellStyleName AS String  
DefColWidth AS Long (measured in Twips)  
DefRowHeight AS Long (measured in Twips)  
Description AS String  
EndingColOfSelection AS Integer  
EndingRowOfSelection AS Integer  
IsAutoGrow AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsParagraphNumberingDown AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsResetParagraphNumber AS Integer (Boolean)  
IsSizingViaMouse AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxBottomBorder AS Long (measured in Twips)  
MaxBottomGutter AS Long (measured in Twips)  
MaxLeftBorder AS Long (measured in Twips)  
MaxLeftGutter AS Long (measured in Twips)  
MaxNumColsAllowed AS Integer  
MaxNumRowsAllowed AS Integer  
MaxRightBorder AS Long (measured in Twips)  
MaxRightGutter AS Long (measured in Twips)  
MaxSplitCols AS Integer  
MaxSplitRows AS Integer  
MaxTopBorder AS Long (measured in Twips)  
MaxTopGutter AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
Parent AS BaseObject  
RowLayouts AS StringCollection  
SelectionType AS SelectionType

SingleCellSelected AS Integer (Boolean)

StartingColOfSelection AS Integer

StartingColStringOfSelection AS String

StartingRowOfSelection AS Integer

TableFill AS TableFill

TableLine AS TableLine

VersionID AS Long

### **Methods**

CellLayout

Connect

Copy

CopyMeaning

DeleteTable

DisconnectCells

DoesMarkerNameMatch

EnumerateTerm

ExtractText

FindCellLayout

FindTerm

GetMarkerName

Glossarize

GoToTableCell

InsertRowOrColumn

Mark

NextItem

PreviousItem

SelectTableItem

Split

### **Events**

None

## Word Pro: Grammar class members

### Properties

Application AS WPAApplication  
ApplyAdjectivePos AS Integer (Boolean)  
ApplyAdjectNounPart AS Integer (Boolean)  
ApplyAgreementWithHereThere AS Integer (Boolean)  
ApplyAnglicisms AS Integer (Boolean)  
ApplyArchaicExpressions AS Integer (Boolean)  
ApplyArticleAgreement AS Integer (Boolean)  
ApplyBadComparatives AS Integer (Boolean)  
ApplyBadInflection AS Integer (Boolean)  
ApplyBadNoun AS Integer (Boolean)  
ApplyBadNounGender AS Integer (Boolean)  
ApplyBadPlural AS Integer (Boolean)  
ApplyBadPrepositions AS Integer (Boolean)  
ApplyBelgianExpression AS Integer (Boolean)  
ApplyBorrowedForeign AS Integer (Boolean)  
ApplyBureuaJargon AS Integer (Boolean)  
ApplyCalque AS Integer (Boolean)  
ApplyCapitalizationCheck AS Integer (Boolean)  
ApplyClauseErrors AS Integer (Boolean)  
ApplyCliches AS Integer (Boolean)  
ApplyColloquialExpression AS Integer (Boolean)  
ApplyCommonlyConfusedWords AS Integer (Boolean)  
ApplyCommonMisspell AS Integer (Boolean)  
ApplyComplexWords AS Integer (Boolean)  
ApplyConfusedEasy AS Integer (Boolean)  
ApplyConfusedEnglish AS Integer (Boolean)  
ApplyConfusedHard AS Integer (Boolean)  
ApplyConfusedMedium AS Integer (Boolean)  
ApplyConfusedVerb AS Integer (Boolean)  
ApplyConsecutiveNouns AS Integer  
ApplyContractions AS Integer (Boolean)  
ApplyDerogatory AS Integer (Boolean)  
ApplyDifferentPrep AS Integer (Boolean)  
ApplyDoubleNegative AS Integer (Boolean)  
ApplyDoublePlural AS Integer (Boolean)  
ApplyDoubleWordCheck AS Integer (Boolean)  
ApplyElision AS Integer (Boolean)  
ApplyEnglishDerived AS Integer (Boolean)  
ApplyEnglishWords AS Integer (Boolean)  
ApplyExotic AS Integer (Boolean)  
ApplyExtraPrepositionCheck AS Integer (Boolean)  
ApplyFalseFriend AS Integer (Boolean)  
ApplyFemaleOccupation AS Integer (Boolean)  
ApplyFixedExpression AS Integer (Boolean)

ApplyForeignWord AS Integer (Boolean)  
ApplyFormalTerms AS Integer (Boolean)  
ApplyFormatErrors AS Integer (Boolean)  
ApplyGallicisms AS Integer (Boolean)  
ApplyGenderExpressions AS Integer (Boolean)  
ApplyGermanisms AS Integer (Boolean)  
ApplyHomoGraphs AS Integer (Boolean)  
ApplyHomonyms AS Integer (Boolean)  
ApplyHomonymsEasy AS Integer (Boolean)  
ApplyHomonymsHard AS Integer (Boolean)  
ApplyHomoPhone1 AS Integer (Boolean)  
ApplyHomoPhone2 AS Integer (Boolean)  
ApplyHomoPhones AS Integer (Boolean)  
ApplyIncorrectPlural AS Integer (Boolean)  
ApplyInformalExpressions AS Integer (Boolean)  
ApplyJargonWords AS Integer (Boolean)  
ApplykSplitInfinitives AS Integer  
ApplyLowercaseAdjective AS Integer (Boolean)  
ApplyLowercaseColor AS Integer (Boolean)  
ApplyLowercaseNumbers AS Integer (Boolean)  
ApplyLowercasePhrases AS Integer (Boolean)  
ApplyLowercasePronouns AS Integer (Boolean)  
ApplyMassVsCount AS Integer (Boolean)  
ApplyMisspelledExpressions AS Integer (Boolean)  
ApplyMisspelledForeignExpressions  
AS Integer (Boolean)  
ApplyMisspelledItalian AS Integer (Boolean)  
ApplyMisspelledWords AS Integer (Boolean)  
ApplyMisusedWords AS Integer (Boolean)  
ApplyNonStandardExpression AS Integer (Boolean)  
ApplyNonStandardModifiers AS Integer (Boolean)  
ApplyNoudModifierOrderCheck AS Integer (Boolean)  
ApplyNounConsistency AS Integer (Boolean)  
ApplyNounPhraseAgree AS Integer (Boolean)  
ApplyNSAdjective AS Integer (Boolean)  
ApplyNSClause AS Integer (Boolean)  
ApplyNSCompare AS Integer (Boolean)  
ApplyNSContract AS Integer (Boolean)  
ApplyNSGeography AS Integer (Boolean)  
ApplyNSInflection AS Integer (Boolean)  
ApplyNSNegation AS Integer (Boolean)  
ApplyNSPrep AS Integer (Boolean)  
ApplyNSPronoun AS Integer (Boolean)  
ApplyNSSpell AS Integer (Boolean)  
ApplyNSUsage AS Integer (Boolean)  
ApplyNSVerbForm AS Integer (Boolean)  
ApplyOpenClosedSpelling AS Integer (Boolean)

ApplyOpenUsage AS Integer (Boolean)  
ApplyOverusedPhrases AS Integer (Boolean)  
ApplyPassiveVerbErrors AS Integer (Boolean)  
ApplyPostClitAgree AS Integer (Boolean)  
ApplyPrepExpression AS Integer (Boolean)  
ApplyPrepositionalPhrases AS Integer  
ApplyPretentiousWords AS Integer (Boolean)  
ApplyPronounErrors AS Integer (Boolean)  
ApplyPunctuationErrors AS Integer (Boolean)  
ApplyRedundantExpressions AS Integer (Boolean)  
ApplyRegionalExpression AS Integer (Boolean)  
ApplyRelatedWord AS Integer (Boolean)  
ApplySensitiveExp AS Integer (Boolean)  
ApplySexistExpressions AS Integer (Boolean)  
ApplySpellStandard AS Integer (Boolean)  
ApplyStockPhrase AS Integer (Boolean)  
ApplyStyleParameters AS Integer (Boolean)  
ApplySubjectVerbAgreement AS Integer (Boolean)  
ApplySwedishGender AS Integer (Boolean)  
ApplySwedishNegation AS Integer (Boolean)  
ApplySwedishUsage AS Integer (Boolean)  
ApplyTrite AS Integer (Boolean)  
ApplyTwoGender AS Integer (Boolean)  
ApplyTypicalMisspell AS Integer (Boolean)  
ApplyUnGrammaticalExpressions AS Integer (Boolean)  
ApplyVagueQuantifiers AS Integer (Boolean)  
ApplyVerbGroupConsistency AS Integer (Boolean)  
ApplyWeakModifiers AS Integer (Boolean)  
ApplyWordChoice AS Integer (Boolean)  
ApplyWordCompoundingCheck AS Integer (Boolean)  
ApplyWordConfusion AS Integer (Boolean)  
ApplyWordGender AS Integer (Boolean)  
ApplyWordParts AS Integer (Boolean)  
ApplyWordyPhraseCheck AS Integer (Boolean)  
Description AS String  
GrammarFormalityLevel AS Integer  
GrammarProofLevel AS Integer  
IsValid AS Integer (Boolean)  
MaxIdenticalConsecSentOpens AS Integer  
MaxIdenticalSentOpensWithin10 AS Integer  
MaximumWordsinaSentence AS Integer  
Name AS String  
Parent AS BaseObject  
ShowStatistics AS Integer (Boolean)  
SpacesBetweenSentences AS Integer  
VersionID AS Long

**Methods**

None

**Events**

None

## **Word Pro: GraphicCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: GraphicOleObjectCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: GraphicOleObject class members**

### **Properties**

Alignment AS Alignment

Application AS WPAApplication

CanEmbed AS Integer (Boolean)

ClassName AS String

ContentType AS ContentType

Description AS String

Font AS Font

Height AS Long (measured in Twips)

IsActive AS Integer (Boolean)

IsEmpty AS Integer (Boolean)

IsReplaceable AS Integer (Boolean)

IsScalable AS Integer (Boolean)

IsValid AS Integer (Boolean)

LayoutName AS String

MetafilePict AS Long

Name AS String

OrigHeight AS Long (measured in Twips)

OrigWidth AS Long (measured in Twips)

Parent AS BaseObject

PositionXOnPage AS Long (measured in Twips)

PositionYOnPage AS Long (measured in Twips)

Section AS String

Text AS Text

VersionID AS Long

Width AS Long (measured in Twips)

### **Methods**

DeleteContent

GetCopyFormatCategories

Mark

TheoreticalScaledSize

### **Events**

None

## Word Pro: Graphic class members

### Properties

Afid AS Variant  
AfidClassName AS String  
Alignment AS Alignment  
Application AS WPAApplication  
AtBeginningOfStream AS Integer (Boolean)  
AtEndOfStream AS Integer (Boolean)  
AutomaticLink AS Integer (Boolean)  
CanEmbed AS Integer (Boolean)  
ClassName AS String  
ContentType AS ContentType  
Description AS String  
EqnFontHeight AS Long (measured in Twips)  
ExternalFileID AS String  
ExternalFileName AS String  
ExternallyControlledUndo AS Integer (Boolean)  
FirstSpellString AS String  
Font AS Font  
GetAfidHelpFileName AS String  
GetAfidHelpInfo AS Integer  
GetUndoWhatDesc AS String  
Height AS Long (measured in Twips)  
InfoBoxSelectionText AS String  
IsActive AS Integer (Boolean)  
IsChartLink AS Integer (Boolean)  
IsDraw AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsEquation AS Integer (Boolean)  
IsExternalFile AS Integer (Boolean)  
IsLotusChart AS Integer (Boolean)  
IsNotCopyImage AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsScalable AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
IsWordProChart AS Integer (Boolean)  
LayoutName AS String  
LinkAvailable AS Integer (Boolean)  
LinkedFileName AS String  
MetafilePict AS Long  
Name AS String  
NextSpellString AS String  
OrigHeight AS Long (measured in Twips)  
OrigWidth AS Long (measured in Twips)  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)

PositionYOnPage AS Long (measured in Twips)

Prompt AS String

RightMousePropId AS Integer

RightMousePropText AS String

Section AS String

ServerFormat AS String

SpellString AS String

Text AS Text

VersionID AS Long

WaterMarkName AS String

Width AS Long (measured in Twips)

### **Methods**

BreakLink

ConvertToClass

CreateFromBitmap

CreateFromClipBrd

CreateFromDataObject

CreateFromMetafile

CreateNew

DataObjectGetData

DataObjectGetDataHere

DeleteContent

DoneWithRightContextMenu

EnumerateChartLinks

GetCopyFormatCategories

GetLinkDisplayNameFileLength

GetLinkName

GetLinkSourceName

GetPasteFormatCategories

GetRightContextMenu

GetSource

GetUserClassNameFull

GetUserClassNameShort

ImportPicture

Mark

ProcessAccelKey

SaveData

SaveSnapshot

SetLinkSource

TheoreticalScaledSize

Undo

UpdateLink

### **Events**

None

## **Word Pro: GroupLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: GroupLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer



UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: Gutter class members**

### **Properties**

AllBorders AS Border

Application AS WPApplication

BottomBorder AS Border

Description AS String

IsValid AS Integer (Boolean)

LeftBorder AS Border

LinePlacement AS LinePlacement

LineValid AS LinePlacement

Name AS String

Parent AS BaseObject

RightBorder AS Border

TopBorder AS Border

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: HeaderLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: HeaderLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout



KeyStroke

MouseDown

MouseUp

## **Word Pro: HyphenationOptions class members**

### **Properties**

Application AS WPAApplication

AutoHyphenate AS Integer (Boolean)

Description AS String

HyphenateLastWordInColumnOrPage

AS Integer (Boolean)

HyphenateLastWordInPara AS Integer (Boolean)

HyphZoneAfter AS Integer

HyphZoneBefore AS Integer

IgnoreSoftHyphens AS Integer (Boolean)

IsValid AS Integer (Boolean)

MaxHyphLines AS Integer

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: IconBarCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: IconBarManager class members**

### **Properties**

Application AS WPAApplication

AreDisabledIconsGrayed AS Integer (Boolean)

Description AS String

IconBars AS IconBarCollection

IconBarSets AS StringCollection

IconHelpText AS String

IconScript AS String

IconSize AS IconSize

IsActionOnButtonDown AS Integer (Boolean)

IsBubbleHelp AS Integer (Boolean)

IsHoverHelp AS Integer (Boolean)

IsIconDepressible AS Integer (Boolean)

IsShowing AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

Configure

HideIconBars

SelectCustomIcon

SelectStandardIcon

ShowIconBars

### **Events**

None

## Word Pro: IconBar class members

### Properties

Application AS WPAApplication

Description AS String

Height AS Long

IconBarPositionState AS IconBarPositionState

IconSetName AS String

ID AS Long

IsEnabled AS Integer (Boolean)

IsGrabBar AS Integer (Boolean)

IsSame AS Integer (Boolean)

IsSave AS Integer (Boolean)

IsScalableBorder AS Integer (Boolean)

IsShowing AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

PositionType AS IconBarPos

ScreenPositionX AS Long

ScreenPositionY AS Long

SetContextOfBar AS Integer

ShowInContext AS Integer (Boolean)

VersionID AS Long

Width AS Long

### Methods

AddIcon

Configure

HideIconBar

Show

### Events

None

## **Word Pro: Indent class members**

### **Properties**

All AS Long (measured in Twips)

Application AS WPApplication

BodyOnly AS Integer (Boolean)

Description AS String

First AS Long (measured in Twips)

Hang AS Integer

IsBothSidesEqual AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Relative AS IndentProperty

Rest AS Long (measured in Twips)

Right AS Long (measured in Twips)

UseRelative AS Integer (Boolean)

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: IndexSection class members**

### **Properties**

Application AS WPAApplication

Color AS Color

Description AS String

IndexAlphabeticSeparator AS Integer (Boolean)

IndexDivision AS String

IndexIndentType AS Integer (Boolean)

IndexParent AS String

IndexRange AS String

IndexSection AS String

IndexSource AS GenerateFrom

IsIndex AS Integer (Boolean)

IsIndex AS Integer (Boolean)

IsValid AS Integer (Boolean)

LayoutName AS String

Name AS String

NextSection AS String

Parent AS BaseObject

ShowTabs AS Integer (Boolean)

UserName AS String

VersionID AS Long

### **Methods**

DeleteSection

GoToSection

### **Events**

None

## **Word Pro: Index class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Offset AS Long

Parent AS BaseObject

Path AS String

Size AS Long

SnapshotOffset AS Long

SnapshotPath AS String

SnapshotSize AS Long

VersionID AS Long

### **Methods**

None

### **Events**

None



## **Word Pro: Join class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

JoinCorners AS JoinCornerPosition

JoinHeight AS Long (measured in Twips)

JoinType AS JoinType

JoinWidth AS Long (measured in Twips)

Name AS String

Parent AS BaseObject

Percentage AS Integer

ScaleMode AS JoinScaleType

VersionID AS Long

### **Methods**

None

### **Events**

None

## Word Pro: Kinsoku class members

### Properties

Application AS WPAApplication

Description AS String

IsEnabled AS Integer (Boolean)

IsHangover AS Integer (Boolean)

IsSqueeze AS Integer (Boolean)

IsValid AS Integer (Boolean)

Level AS Integer

Name AS String

Parent AS BaseObject

VersionID AS Long

### Methods

RevertToStyle

### Events

None

## **Word Pro: Language class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Language AS Languages

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: LayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: LayoutOverride class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddLayoutOverride

### **Events**

None

## Word Pro: Layout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer



UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: LineNumberOptions class members**

### **Properties**

Application AS WPAApplication

CountBlankLines AS Integer (Boolean)

Description AS String

DistanceFromLeftMargin AS Long (measured in Twips)

IsValid AS Integer (Boolean)

LinesSpacedEveryNthUnit AS Long (measured in Twips)

Name AS String

NumberEveryNthLine AS Integer

NumberWhichLines AS LineNumberOpts

Parent AS BaseObject

ResetOnEachPage AS Integer (Boolean)

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: LongCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: LWPTimer class members**

### **Properties**

Application AS WPApplication

Description AS String

Enabled AS Integer (Boolean)

Interval AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

TimerTick

## **Word Pro: Macro class members**

### **Properties**

Application AS WPApplication

DebugVariable AS Integer

Description AS String

IsDebug AS Integer (Boolean)

IsValid AS Integer (Boolean)

MacroName AS String

MacroStatus AS MacroStatus

Name AS String

Parent AS BaseObject

ResumePausedMacro AS Integer (Boolean)

VersionID AS Long

### **Methods**

AppendMacro

GetPowerFieldValue

RemovePowerField

Run

SaveMacro

SetPowerFieldValue

### **Events**

None

## **Word Pro: MailRouting class members**

### **Properties**

Application AS WPAApplication

CanEditProperty AS Integer (Boolean)

Description AS String

InitializeRoute AS Integer (Boolean)

Initialize AS Integer (Boolean)

IsDocumentInRoute AS Integer (Boolean)

IsLastStop AS Integer (Boolean)

IsValid AS Integer (Boolean)

MailRouteingPtr AS Long

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: MarkerCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: Marker class members**

### **Properties**

Application AS WPApplication  
Description AS String  
DivisionInfo AS DivisionInfo  
IsChanged AS Integer (Boolean)  
IsMarkerValid AS Integer (Boolean)  
IsRegistered AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MarkerClass AS MarkerType  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
PageNumber AS Integer  
PageNumberAsText AS String  
Parent AS BaseObject  
StartColumns AS Integer  
StartRow AS Integer  
StateID AS Long  
VersionID AS Long  
WPDataSets AS WPDataSetCollection

### **Methods**

DeleteContents  
DeleteMarker  
GetContents  
GetMarkedText  
GetNamedProperty  
GetParagraphNumber  
GoTo  
HasNamedProperty  
InsertMarker  
RegisterWPDataSet  
RemoveNamedProperty  
SetNamedProperty  
UnregisterWPDataSet

### **Events**

None

## **Word Pro: MenuItemCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: MenuItem class members**

### **Properties**

Action AS String

Application AS WPApplication

Caption AS String

Checked AS Integer (Boolean)

Description AS String

Enabled AS Integer (Boolean)

ID AS Long

IsValid AS Integer (Boolean)

Items AS MenuItemCollection

Name AS String

Parent AS BaseObject

ParentMenuHWND AS Long

VersionID AS Long

### **Methods**

CopyItem

DeleteItem

ExchangeItem

MoveItem

NewItem

### **Events**

None

## **Word Pro: MergeOptions class members**

### **Properties**

Application AS WPAApplication

DataFileFieldNames AS String

DataFileName AS String

Description AS String

DescriptionFileName AS String

FilterName AS String

IsAmiProTableImport AS Integer

IsValid AS Integer (Boolean)

MergeFileType AS Integer

MergeInfoPtr AS Long

MergeStepNumber AS MergeStepNumber

MergeToFile AS String

Name AS String

NumberOfDataFields AS Integer

NumberOfMergeConditions AS Integer

Options AS MergeOptFlg

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddACondition

Clear

RemoveDataFile

### **Events**

None

## **Word Pro: NoteContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: NoteLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: NoteLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
Color AS Color  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
InitialsForFilters AS String  
IsAdopted AS Integer (Boolean)



IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)  
IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS Layout  
Layer AS String  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)

LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer  
LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NameForFilters AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
Text AS Text  
TextOrient AS TextOrient

Tile AS Integer (Boolean)  
Time AS Long  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer  
UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

#### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Hide  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
Open  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange

RevisionCancelLayoutChange

SetAllMargins

SetMinimumOrigin

SetNamedProperty

UnregisterWPDataSet

Update

**Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: Numbering class members**

### **Properties**

Application AS WPAApplication

Description AS String

Heading AS Integer

IsValid AS Integer (Boolean)

Level AS Integer

Name AS String

Parent AS BaseObject

Position AS Integer

SmartLevel AS CommandState

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: NumericFormatSubset class members**

### **Properties**

Application AS WPAApplication

Color AS Color

ColorOverride AS Integer (Boolean)

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Prefix AS String

Suffix AS String

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: NumericFormat class members**

### **Properties**

AnyNumber AS NumericFormatSubset

Application AS WPApplication

Description AS String

FormatType AS NumberFormat

IsValid AS Integer (Boolean)

Name AS String

Negative AS NumericFormatSubset

NumDecimalPlaces AS Integer

Parent AS BaseObject

VersionID AS Long

Zero AS NumericFormatSubset

### **Methods**

Reset

### **Events**

None

## **Word Pro: OleObjectCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: OleObject class members**

### **Properties**

Alignment AS Alignment  
Application AS WPAApplication  
AutomaticLink AS Integer (Boolean)  
CanEmbed AS Integer (Boolean)  
ClassId AS String  
ClassName AS String  
ContentType AS ContentType  
Description AS String  
DisplayAsIcon AS Integer (Boolean)  
FileFormat AS Long  
Font AS Font  
Height AS Long (measured in Twips)  
IDispatch AS Variant  
InDocument AS Integer (Boolean)  
IsActive AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsScalable AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
LayoutName AS String  
LinkAvailable AS Integer (Boolean)  
LinkDisplayName AS String  
LinkDisplayNameFileLength AS Long  
Linked AS Integer (Boolean)  
MetafilePict AS Long  
Name AS String  
Object AS Variant  
OleObjectSize AS Long  
OrigHeight AS Long (measured in Twips)  
OrigWidth AS Long (measured in Twips)  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
ProgID AS String  
Section AS String  
Text AS Text  
UserClassNameApp AS String  
UserClassNameFull AS String  
UserClassNameShort AS String  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

ActivateAs  
AddVerbMenu

BreakLink

ConvertTo

DeleteContent

GetCopyFormatCategories

Mark

Open

SetLinkSource

TheoreticalScaledSize

UpdateLink

**Events**

None

## **Word Pro: Options class members**

### **Properties**

Application AS WPAApplication  
CharacterSetName AS String  
ClickHerePrompts AS Integer  
Description AS String  
EmbedFonts AS Integer  
EncryptPassword AS String  
EncryptPassword2 AS String  
GrammarSetName AS String  
IncludeInitialsInNotes AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
MarkCharacter AS Integer  
MarkPosition AS MarkPosition  
MarkType AS MarkType  
Name AS String  
PairKerning AS Integer (Boolean)  
Parent AS BaseObject  
PowerField AS Integer  
UseEncrypt AS Integer (Boolean)  
VersionID AS Long  
WidowOrphan AS Integer (Boolean)

### **Methods**

None

### **Events**

None

## **Word Pro: OutlineSeqCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: OutlineSeqItemCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: OutlineStyleSequence class members**

### **Properties**

Application AS WPApplication

Description AS String

Heading AS Integer

IsValid AS Integer (Boolean)

Name AS String

OutlineSeqItems AS OutlineSeqItemCollection

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddOutlineSequenceItem

Clear

### **Events**

None

## **Word Pro: OutSeqItem class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Position AS Integer

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: PageContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide



IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: PageLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: PageLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## Word Pro: ParagraphBorder class members

### Properties

Application AS WPAApplication

BorderLines AS BorderLines

Description AS String

IsBorder AS Integer (Boolean)

IsValid AS Integer (Boolean)

MarginBottom AS Long (measured in Twips)

MarginLeft AS Long (measured in Twips)

MarginRight AS Long (measured in Twips)

MarginTop AS Long (measured in Twips)

Name AS String

Parent AS BaseObject

Shadow AS Shadow

TypeAbove AS ParaBorderWidth

TypeBelow AS ParaBorderWidth

TypeRight AS ParaBorderWidth

VersionID AS Long

WidthAbove AS Long (measured in Twips)

WidthBelow AS Long (measured in Twips)

### Methods

RevertToStyle

### Events

None



## **Word Pro: ParagraphStyleCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: ParagraphStyle class members

### Properties

Alignment AS Alignment  
AlignStyleName AS String  
Amikake AS Amikake  
AmikakeName AS String  
Application AS WPApplication  
AttrStyleName AS String  
BorderStyleName AS String  
Breaks AS Breaks  
BreaksStyleName AS String  
Bullet AS Bullet  
BulletStyleName AS String  
CharacterBorder AS CharacterBorder  
CharacterBorderName AS String  
Definition AS Long  
Description AS String  
DocumentLevel AS Integer  
FaceStyleName AS String  
Font AS Font  
FontStyleName AS String  
HasTabs AS Integer (Boolean)  
Heading AS Integer  
Indent AS Indent  
IndentStyleName AS String  
InUseCount AS Long  
IsCumulative AS Integer (Boolean)  
IsLesser AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsPrivate AS Integer (Boolean)  
IsTemp AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Kinsoku AS Kinsoku  
Language AS Language  
Name AS String  
Numbering AS Numbering  
NumberingPosition AS Integer  
NumberingStyleName AS String  
ParagraphBorder AS ParagraphBorder  
Parent AS BaseObject  
RelativeIndent AS RelativeIndent  
SizeStyleName AS String  
Spacing AS Spacing  
SpacingStyleName AS String  
TabRack AS TabRack  
TextAttributes AS Attributes

Type AS ParaStyleType

VersionID AS Long

WPDataSets AS WPDataSetCollection

**Methods**

Clear

IsTemporary

OutlineStyleSequence

RegisterWPDataSet

UnregisterWPDataSet

Update

**Events**

None

## **Word Pro: ParallelColsCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: ParallelColsContainer class members

### Properties

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### Methods

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: FrameStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FRAMESTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the FrameLayoutCollection class. This object provides access to FrameLayout objects which are used as frame styles.

### **Data Type**

[FrameLayoutCollection](#)

### **Syntax**

framestylesvalue = [objectreference].FrameStyles

### **Legal values**

Always contains an instance of the FrameLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to the FrameLayout objects contained in that Division object which are used as frame styles.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to the FrameLayout objects contained in the Word Pro Clipboard which are used as frame styles.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to the FrameLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to the FrameLayout objects contained in the currently active Division object which are used as frame styles.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## **Word Pro: Frames property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FRAMES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the FrameLayoutCollection class. This object provides access to FrameLayout objects including those used as frame styles.

### **Data Type**

[FrameLayoutCollection](#)

### **Syntax**

framesvalue = [objectreference].Frames

### **Legal values**

Always contains an instance of the FrameLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the FrameLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the FrameLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the FrameLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the FrameLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#).

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).



**Word Pro: Frame property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FRAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the FrameContainer class. This is a current context property which only contains an object when the focus of Word Pro includes a frame. If there is no frame in the focus, this property is empty.

**Data Type**

[FrameContainer](#)

**Syntax**

framevalue = [objectreference].Frame

**Legal values**

An instance of the FrameContainer class.

**Usage**

When the focus includes a frame, this property contains the FrameContainer object which groups together the objects that comprise the frame which has the focus. You can use this property to access the layout or other objects related to that frame.

**Word Pro: FreeMenus property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_FREEMENUS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) MenuItem(s) object(s) created by the script writer from the MenuItem class.

**Data Type**

[MenuItem](#)

**Syntax**

freemenusvalue = [objectreference].FreeMenus

**Legal values**

Always contains an instance of the MenuItem class.

**Usage**

Use this property as a holding area for menu items until you implement them in Word Pro.

## **Word Pro: GlossaryDataFiles property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYDATAFILES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Multiple names for the Word Pro Glossary file.

### **Data Type**

[StringCollection](#)

### **Syntax**

glossarydatafilesvalue = [objectreference].GlossaryDataFiles

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "Default glossary file(s)" field on the Default files panel of the Word Pro Preferences dialog box. In the Word Pro interface, the "Default glossary file(s)" field can contain multiple file names. You can use this property to read these multiple file names, including the primary (default) file name stored in the GlossaryDataFileName property.

## **Word Pro: GlossaryDataPaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYDATAPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Multiple paths (drive and directory) where Word Pro looks for the Glossary file.

### **Data Type**

[StringCollection](#)

### **Syntax**

glossarydatapathsvalue = [objectreference].GlossaryDataPaths

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "Glossaries" field on the Locations panel of the Word Pro Preferences dialog box. The "Glossaries" field can contain multiple datafile paths. You can use this property to read these multiple paths, including the primary (default) path stored in the GlossaryPath property.

## **Word Pro: Glossary's property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GLOSSARYS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the GlossaryCollection class. This object provides access to Glossary objects.

### **Data Type**

[GlossaryCollection](#)

### **Syntax**

glossarysvalue = [objectreference].Glossarys

### **Legal values**

Always contains an instance of the GlossaryCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Glossary objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the Glossary objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the Glossary objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the Glossary objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

**Word Pro: GrammarOptions property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAMMAROPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the Grammar class. This object holds a pointer to the grammar options available on the Grammar Check bar, Options dialog box.

**Data Type**

[Grammar](#)

**Syntax**

grammaroptionsvalue = [objectreference].GrammarOptions

**Legal values**

Always contains an instance of the Grammar class.

**Usage**

Currently, the Word Pro Grammar Check contains 128 options. These options include on or off for all Grammar Check rules, and the values for other Grammar Check options, such as split infinitives, consecutive nouns, maximum number of words, and so on.

**Word Pro: GraphicExports property**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHICEXPORTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

graphicexportsvalue = [objectreference].GraphicExports

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

**Word Pro: GraphicImports property**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHICIMPORTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

graphicimportsvalue = [objectreference].GraphicImports

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**



**Word Pro: GraphicOleObject property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHICOLEBJECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains the Graphic or OleObject object which is uppermost in the focus.

**Data Type**

[GraphicOleObject](#)

**Syntax**

graphicoleobjectvalue = [objectreference].GraphicOleObject

**Legal values**

Always contains an instance of the GraphicOleObject class.

**Usage**

Use this property when you want to access a graphic or OLE object, regardless of whether the object is seen by LotusScript as a Graphic object or an OleObject object.

## **Word Pro: Graphics property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHICS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the GraphicCollection class. This object provides access to Graphic objects.

### **Data Type**

[GraphicCollection](#)

### **Syntax**

graphicsvalue = [objectreference].Graphics

### **Legal values**

Always contains an instance of the GraphicCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Graphic objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the Graphic objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the Graphic objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the Graphic objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: Graphic property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GRAPHIC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains the Graphic object which is uppermost in the focus.

**Data Type**

[Graphic](#)

**Syntax**

graphicvalue = [objectreference].Graphic

**Legal values**

Always contains an instance of the Graphic class.

**Usage**

Use this property when you want to access the Graphic object which currently has the focus. If you want to access a graphic that is an OLE object, use the OleObject property on WPAApplication. If you are not sure if a graphic is an OLE object, use the GraphicOleObject property that is capable of containing both Graphic objects and OleObject objects.

## **Word Pro: Groups property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_GROUPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the GroupLayoutCollection class. This object provides access to GroupLayout objects.

### **Data Type**

[GroupLayoutCollection](#)

### **Syntax**

groupsvalue = [objectreference].Groups

### **Legal values**

Always contains an instance of the GroupLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the GroupLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the GroupLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the GroupLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the GroupLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## Word Pro: Gutter property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_GUTTER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The border settings between text columns. This border is always positioned in the center of the column gap.

## Data Type

Gutter

## Syntax

guttervalue = [objectreference].Gutter

[objectreference].Gutter = guttervalue

## Legal values

Always contains an instance of the Gutter class.

## Usage

Equivalent to the "Line style," "Line width," and "Line color" settings on the Columns panel of the InfoBox for certain layout objects.

## **Word Pro: Headers property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HEADERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the HeaderLayoutCollection class. This object provides access to HeaderLayout objects.

### **Data Type**

[HeaderLayoutCollection](#)

### **Syntax**

headersvalue = [objectreference].Headers

### **Legal values**

Always contains an instance of the HeaderLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the HeaderLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the HeaderLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the HeaderLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the HeaderLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## Word Pro: Header property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_HEADER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns a header object in a layout object.

## Data Type

[Layout](#)

## Syntax

headervalue = [objectreference].Header

## Legal values

Always contains an instance of the Layout class.

## Usage

Use this property to access the header layout object of the current layout. Not all layout objects have header layouts. You can check the UseHeader property of a layout object to see whether there is a header layout object currently available.

## **Word Pro: HiLiteColor property**

{button ,AL('H\_EDITOR\_CLASS;H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HILITECOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The color that is used to highlight text in a document.

### **Data Type**

[Color](#)

### **Syntax**

hilitecolorvalue = [objectreference].HiLiteColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

[Editor]

This property is equivalent to the "Highlighter/comment color" option, which is located in the Markup Options dialog. The Markup Options dialog can be opened by pressing the Markup Options button, which is located in the General panel of the Word Pro Preferences dialog.



**Word Pro: HorzRuler property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HORZRULER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains a ruler object that indicates tab settings, indents, margins, and columns.

**Data Type**

[Ruler](#)

**Syntax**

horzrulervalue = [objectreference].HorzRuler

**Legal values**

Always contains an instance of the Ruler class.

**Usage**

Use this property to display the horizontal ruler when no document is open.

## **Word Pro: HyphenationOptions property**

{button ,AL('H\_DIVISIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HYPHENATIONOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[HyphenationOptions](#)

### **Syntax**

hyphenationoptionsvalue = [objectreference].HyphenationOptions

### **Legal values**

Always contains an instance of the HyphenationOptions class.

### **Usage**

**Word Pro: IconBarManager property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONBARMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The IconBarManager object for the current application window

**Data Type**

[IconBarManager](#)

**Syntax**

iconbarmanagervalue = [objectreference].IconBarManager

**Legal values**

Always contains an instance of the IconBarManager class.

**Usage**

You must go through this property to get to the IconBar for the application window. The application window can have multiple iconbars, but only one IconBarManager. You can use the IconBarManager to select, find, add, or remove icon bar objects.

**Word Pro: IconBarSets property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONBARSETS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enumerates all icon bar sets by name, whether or not they are currently showing. The icon bar sets are listed by icon bar name, not by file name.

**Data Type**

[StringCollection](#)

This field is not used for properties with CLASS data types.

**Syntax**

iconbarsetsvalue = [objectreference].IconBarSets

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

If you know the name of the icon bar set, you can select it through the IconBar class. You can then display it, hide it, add and remove icons from the set, and so on.

### **Word Pro: IconBars property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONBARS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enumerates all icon bars that are currently showing, by icon bar name. The user interface control is the file button on the solid color bar adjacent to an icon bar object.

### **Data Type**

[IconBarCollection](#)

### **Syntax**

iconbarsvalue = [objectreference].IconBars

### **Legal values**

Always contains an instance of the IconBarCollection class.

### **Usage**

If you click the file button adjacent to an icon bar object, a list of icon bar sets that can show in the current context will display. The icon bar names with check marks next to them are the ones currently showing. This property returns a String value.

### **Word Pro: IconPaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains multiple paths (drive and directory) for SmartIcons.

### **Data Type**

[StringCollection](#)

### **Syntax**

iconpathsvalue = [objectreference].IconPaths

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "SmartIcons" field on the Locations panel of the Word Pro Preferences dialog box. The "SmartIcons" field can contain multiple document paths. You can use this property to read these multiple document paths, including the primary (default) document path that is stored in the IconPath property.

### **Word Pro: IconScript property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONSCRIPT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Sets a script or macro for a custom icon.

### **Data Type**

[StringCollection](#)

This field is not used for properties with CLASS data types.

### **Syntax**

iconscript = [objectreference].IconScript

[objectreference].IconScript = iconscriptvalue

### **Legal values**

String

### **Usage**

Before you can write a script for an icon, you must first select the icon using the SelectCustomIcon method.

**Word Pro: IdentifierColor property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IDENTIFIERCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Color](#)

**Syntax**

identifiercolorvalue = [objectreference].IdentifierColor

**Legal values**

Always contains an instance of the Color class.

**Usage**



### **Word Pro: Indent property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_T  
EXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INDENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Indent

### **Syntax**

indentvalue = [objectreference].Indent

### **Legal values**

Always contains an instance of the Indent class.

### **Usage**

## **Word Pro: InsertFont property**

{button ,AL(^H\_EDITOR\_CLASS;H\_REVISIONDISPLAY\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INSERTFONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The collection of attributes associated with a font object that marks inserted text in a document.

### **Data Type**

[Font](#)

### **Syntax**

insertfontvalue = [objectreference].InsertFont

### **Legal values**

Always contains an instance of the Font class.

### **Usage**

[Editor]

This property is equivalent to the "Markup for insertions" option, which is located in the Markup Options dialog. The Markup Options dialog can be opened by pressing the Markup Options button, which is located in the General panel of the Word Pro Preferences dialog.

## **Word Pro: Items property**

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ITEMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A collection item used by the current parent menu item to hold submenu items.

### **Data Type**

[MenuItemCollection](#)

### **Syntax**

itemsvalue = [objectreference].Items

### **Legal values**

Always contains an instance of the MenuItemCollection class.

### **Usage**

This property allows you to access submenu items for a specific parent menu item. For example, the Word Pro main menu resides in the LWPMenuBar property in the ApplicationWindow class. If you want to access the Word Pro File menu, you must use the Items property to return the File menu in the LWPMenuBar. Menu items in the Items property are enumerated by their Caption property.

## Word Pro: Join property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_JOIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Allows you to access the join object for a specific layout object.

## Data Type

[Join](#)

## Syntax

joinvalue = [objectreference].Join

## Legal values

Always contains an instance of the Join class.

## Usage

### **Word Pro: KeywordColor property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_KEYWORDCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Color](#)

### **Syntax**

keywordcolorvalue = [objectreference].KeywordColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

### **Word Pro: Kinsoku property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',  
0)} [See list of classes](#)

{button ,AL(^H\_KINSOKU\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Kinsoku](#)

### **Syntax**

kinsokuvalue = [objectreference].Kinsoku

### **Legal values**

Always contains an instance of the Kinsoku class.

### **Usage**

## Word Pro: Language property

```
{button ,AL('H_APPLICATION_CLASS;H_CHARACTERSTYLE_CLASS;H_CLICKHERE_CLASS;H_DIVISIONOPTI  
ONS_CLASS;H_FORMATPREFERENCES_CLASS;H_LANGUAGE_CLASS;H_PARAGRAPHSTYLE_CLASS;H_  
TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_LANGUAGE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) An instance of the Language class.

### Data Type

[Language](#)

### Syntax

languagevalue = [objectreference].Language

### Legal values

Always contains an instance of the Language class.

### Usage

This property is available on the following types of objects:

- CharacterStyle
- ClickHere
- DivisionOptions
- FormatPreferences
- ParagraphStyle
- Text
- TextMarker

The Language object stored in this property is determined by the object from which you call this property. For example, if you call this property from a ClickHere object, you get the Language object for that ClickHere. The scope and use of the Language object in this property is also determined by the object from which you call this property.

**Word Pro: LayoutOverride property**

{button ,AL(^H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LAYOUTOVERRIDE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[LayoutOverride](#)

**Syntax**

layoutoverridevalue = [objectreference].LayoutOverride

**Legal values**

Always contains an instance of the LayoutOverride class.

**Usage**



## Word Pro: Layouts property

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the LayoutCollection class. This object provides access to all types of Layout objects. Layout objects are those objects that are created from one of the derived Layout classes listed below:

- CellLayout
- ConnectedLayout
- EndnoteLayout
- FooterLayout
- FootnoteLayout
- FrameLayout
- GroupLayout
- HeaderLayout
- NoteLayout
- PageLayout
- RowLayout
- RubyLayout
- SuperTableLayout
- TableHeadingLayout
- TableLayout

All of these classes are derived from the same Layout class and share the common set of Layout class members. The objects created from these classes are all related through their common parent class, Layout. That is why we say they are Layout objects. One of the benefits of related classes of objects is the ability to store related objects in a variable that has the parent class data type.

For example, a variable of type CellLayout could only hold an object created from the CellLayout class. But a variable of type Layout could hold any object created from one of the Layout-derived classes listed above.

When you use the LayoutCollection object in this property, you have access to all types of Layout objects.

## Data Type

[LayoutCollection](#)

## Syntax

layoutsvalue = [objectreference].Layouts

## Legal values

Always contains an instance of the LayoutCollection class.

## Usage

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Layout objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the Layout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the Layout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the Layout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## Word Pro: Shadow property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASSES;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_LAYOUT_PARAGRAPHBORDER_SHADOW_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The shadow object for a layout or paragraph border object.

## Data Type

[Shadow](#)

## Syntax

Shadowvalue = [objectreference.]Shadowvalue

## Legal values

Always contains an instance of the Shadow class.

## Usage

## Word Pro: Layout property

{button ,AL(^H\_BASECONTAINER\_CLASS;H\_BASSETABLE\_CLASS;H\_CELLCONTAINER\_CLASS;H\_CLICKHERE\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_GLOSSARY\_CLASS;H\_MARKER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_PARALLELOLUMNS\_CLASS;H\_POWERFIELD\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_RUBYSMARKER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLE\_CLASS;H\_TABLECONTAINER\_CLASS;H\_TABLEHEADING\_CLASS;H\_TABLEMARKER\_CLASS;H\_TABLEONLYCONT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LAYOUT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the layout object of any container.

## Data Type

[Layout](#)

## Syntax

layoutvalue = [objectreference].Layout

## Legal values

Always contains an instance of the Layout class or one of its derived classes.

## Usage

When called from WPAApplication, this property contains the Layout object for the container object that is uppermost in the focus.

When called from a container class, this property contains a layout object which corresponds to the container object's type. For example, the layout property of a frame container object contains a frame layout object. The layout property of a cell container object contains a cell layout object.

**Word Pro: LeftBorder property**

{button ,AL(^H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LEFTBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Border](#)

**Syntax**

leftbordervalue = [objectreference].LeftBorder

**Legal values**

Always contains an instance of the Border class.

**Usage**

### Word Pro: LeftPage property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_LEFTPAGE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The left page of a complex layout. A complex layout contains separate layouts for left and right pages.

### Data Type

[PageLayout](#)

### Syntax

leftpagevalue = [objectreference].LeftPage

### Legal values

Always contains an instance of the PageLayout class.

### Usage

**Word Pro: LineNumberOptions property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINENUMBEROPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[LineNumberOptions](#)

**Syntax**

linenumbersvalue = [objectreference].LineNumberOptions

**Legal values**

Always contains an instance of the LineNumberOptions class.

**Usage**

## **Word Pro: LwpMenuBar property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LWPMENUBAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The main Word Pro application menu bar object, created from the MenuItem class.

### **Data Type**

[MenuItem](#)

### **Syntax**

lwpmenubarvalue = [objectreference].LwpMenuBar

### **Legal values**

Always contains an instance of the MenuItem class.

### **Usage**

The MenuItem class is used to set and get your own menu items. You can get the current values of LWP menu items, but you cannot change them. To change LWPMenuItems, you must create your own menu item and replace the LWPMenuItem with the new menu items.

## **Word Pro: MacroPaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACROPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores multiple paths (drives and directories) for Word Pro scripts.

### **Data Type**

[StringCollection](#)

### **Syntax**

macropathvalue = [objectreference].MacroPaths

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "Scripts" field on the Locations panel of the Word Pro Preferences dialog box. The "Scripts" field can contain multiple paths. You can use this property to read these multiple paths, including the default or first script path that is also stored in the MacroPath property.



**Word Pro: Macro property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACRO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the macro class for the current application window.

**Data Type**

[Macro](#)

**Syntax**

macrovalue = [objectreference].Macro

**Legal values**

Always contains an instance of the Macro class.

**Usage**

Use this property to run scripts and/or macros saved in another file, or to run scripts and/or macros when you do not have any open documents.

**Word Pro: MailRouting property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAILROUTING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[MailRouting](#)

**Syntax**

mailroutingvalue = [objectreference].MailRouting

**Legal values**

Always contains an instance of the MailRouting class.

**Usage**

## **Word Pro: MarginColor property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MARGINCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores the color of a document's margin as seen in Layout view when "Show margin in color" is enabled.

### **Data Type**

[Color](#)

### **Syntax**

margincolorvalue = [objectreference].MarginColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

## **Word Pro: Markers property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MARKERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the MarkerCollection class. This object provides access to Marker objects.

### **Data Type**

[MarkerCollection](#)

### **Syntax**

markersvalue = [objectreference].Markers

### **Legal values**

Always contains an instance of the MarkerCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Marker objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the Marker objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the Marker objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the Marker objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

**Word Pro: Master property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MASTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[TextDocument](#)

**Syntax**

mastervalue = [objectreference].Master

**Legal values**

Always contains an instance of the TextDocument class.

**Usage**

## **Word Pro: MergeOptions property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGEOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[MergeOptions](#)

### **Syntax**

mergeoptionsvalue = [objectreference].MergeOptions

### **Legal values**

Always contains an instance of the MergeOptions class.

### **Usage**

**Word Pro: Negative property**

{button ,AL(^H\_NUMERICFORMAT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEGATIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[NumericFormatSubset](#)

**Syntax**

negativevalue = [objectreference].Negative

**Legal values**

Always contains an instance of the NumericFormatSubset class.

**Usage**

**Word Pro: NoteColor property**

{button ,AL(^H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NOTECOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The default color for comment note marks.

**Data Type**

[Color](#)

**Syntax**

notecolorvalue = [objectreference].NoteColor

**Legal values**

Always contains an instance of the Color class.

**Usage**

Equivalent to the "Highlighter/comment color" box in the Markup Options for current editor dialog box.



## **Word Pro: NoteLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NOTELAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the NoteLayoutCollection class. This object provides access to NoteLayout objects.

### **Data Type**

[NoteLayoutCollection](#)

### **Syntax**

notelayoutsvalue = [objectreference].NoteLayouts

### **Legal values**

Always contains an instance of the NoteLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the NoteLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the NoteLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the NoteLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the NoteLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

### **Word Pro: Numbering property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',  
0)} [See list of classes](#)

{button ,AL(^H\_NUMBERING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Numbering](#)

### **Syntax**

numberingvalue = [objectreference].Numbering

### **Legal values**

Always contains an instance of the Numbering class.

### **Usage**

### **Word Pro: NumericFormat property**

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_NUMERICFORMAT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns a numeric format object in a specific layout object.

### **Data Type**

[NumericFormat](#)

### **Syntax**

numericformatvalue = [objectreference].NumericFormat

### **Legal values**

Always contains an instance of the NumericFormat class.

### **Usage**

## **Word Pro: OleObjects property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLEOBJECTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the OleObjectCollection class. This object provides access to OleObject objects.

### **Data Type**

[OleObjectCollection](#)

### **Syntax**

oleobjectsvalue = [objectreference].OleObjects

### **Legal values**

Always contains an instance of the OleObjectCollection class.

### **Usage**

Use this property to determine if any OLE objects exist in a particular object.

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the OleObject objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the OleObject objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the OleObject objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the OleObject objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: OleObject property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLEOBJECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The OleObject object which is uppermost in the focus of the currently active document.

**Data Type**

[OleObject](#)

**Syntax**

oleobjectvalue = [objectreference].OleObject

**Legal values**

Always contains an instance of the OleObject class.

**Usage**

Use this property when you want to access the OLE object that currently has the focus. If you want to access a graphic that is not an OLE object, use the Graphic property on WPAApplication. If you are not sure if a graphic is an OLE object, use the GraphicOleObject property that is capable of containing both Graphic objects and OleObject objects.

## **Word Pro: OutlineBorderLines property**

{button ,AL(^H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_OUTLINEBORDERLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-Write) Allows you to return or set the style of the outside border line of a table object.

### **Data Type**

[BorderLines](#)

### **Syntax**

outlineborderlinesvalue = [objectreference].OutlineBorderLines

[objectreference].OutlineBorderLines = outlineborderlinesvalue

### **Legal values**

Always contains an instance of the BorderLines class.

### **Usage**

**Word Pro: OutlineSeqItems property**

{button ,AL('H\_OUTLINESTYLESEQUENCE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINESEQITEMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[OutlineSeqItemCollection](#)

**Syntax**

outlineseqitemsvalue = [objectreference].OutlineSeqItems

**Legal values**

Always contains an instance of the OutlineSeqItemCollection class.

**Usage**

## **Word Pro: OutlineStyleSequences property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINESTYLESEQUENCES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the OutlineSeqCollection class. This object provides access to OutlineStyleSequence objects.

### **Data Type**

[OutlineSeqCollection](#)

### **Syntax**

outlinestylesequencesvalue = [objectreference].OutlineStyleSequences

### **Legal values**

Always contains an instance of the OutlineSeqCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the OutlineStyleSequence objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the OutlineStyleSequence objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the OutlineStyleSequence objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the OutlineStyleSequence objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.



## **Word Pro: PageStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGESTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the PageLayoutCollection class. This object provides access to PageLayout objects which are used as page styles.

### **Data Type**

[PageLayoutCollection](#)

### **Syntax**

pagestylesvalue = [objectreference].PageStyles

### **Legal values**

Always contains an instance of the PageLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the PageLayout objects which are used as page styles and contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the PageLayout objects which are used as page styles and contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the PageLayout objects which are used as page styles and placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the PageLayout objects which are used as page styles and contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: Pages property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the PageLayoutCollection class. This object provides access to PageLayout objects.

### **Data Type**

[PageLayoutCollection](#)

### **Syntax**

pagesvalue = [objectreference].Pages

### **Legal values**

Always contains an instance of the PageLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the PageLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the PageLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the PageLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the PageLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

**Word Pro: Page property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the PageContainer class. This is a current context property which only contains an object when the focus of Word Pro includes a page. If there is no page in the focus, this property is empty.

**Data Type**

[PageContainer](#)

**Syntax**

pagevalue = [objectreference].Page

**Legal values**

An instance of the PageContainer class.

**Usage**

When the focus includes a page, this property contains the PageContainer object which groups together the objects that comprise the page that has the focus. You can use this property to access the Layout or other objects related to that page.

**Word Pro: PaneColor property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PANECOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the Color class which represents the color of the application window's background.

**Data Type**

[Color](#)

**Syntax**

panecolorvalue = [objectreference].PaneColor

**Legal values**

Always contains an instance of the Color class.

**Usage**

Word Pro does not use this property.

## **Word Pro: PaperNames property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAPERNAME\_PROPERTIES\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[StringCollection](#)

### **Syntax**

papernamesvalue = [objectreference].PaperNames

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

## **Word Pro: ParagraphBorder property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',  
0)} [See list of classes](#)

{button ,AL('H\_PARAGRAPHBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[ParagraphBorder](#)

### **Syntax**

paragraphbordervalue = [objectreference].ParagraphBorder

### **Legal values**

Always contains an instance of the ParagraphBorder class.

### **Usage**

## **Word Pro: ParagraphHasDropCap property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARAGRAPHHASDROPCAP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Integer

### **Syntax**

paragraphhasdropcapvalue = [objectreference].ParagraphHasDropCap

### **Legal values**

Boolean

### **Usage**

### **Word Pro: ParagraphHasText property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PARAGRAPHHASTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether there are any Text fribs present in the current paragraph.

### **Data Type**

Integer

### **Syntax**

paragraphbordervalue = [objectreference].ParagraphBorder

### **Legal values**

Always contains an instance of the ParagraphBorder class.

### **Usage**



## **Word Pro: ParagraphStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARAGRAPHSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the ParagraphStyleCollection class. This object provides access to ParagraphStyle objects.

### **Data Type**

[ParagraphStyleCollection](#)

### **Syntax**

paragraphstylesvalue = [objectreference].ParagraphStyles

### **Legal values**

Always contains an instance of the ParagraphStyleCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the ParagraphStyle objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the ParagraphStyle objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the ParagraphStyle objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the ParagraphStyle objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## **Word Pro: ParagraphStyle property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARAGRAPHSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[ParagraphStyle](#)

### **Syntax**

paragraphstylevalue = [objectreference].ParagraphStyle

### **Legal values**

Always contains an instance of the ParagraphStyle class.

### **Usage**

**Word Pro: ParallelColumns property**

{button ,AL(`H\_FOUNDRY\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_PARALLELCOLUMNS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The ParallelColumns object which is uppermost in the focus when this property is called.

**Data Type**

[ParallelColumns](#)

**Syntax**

parallelcolumnsvalue = [objectreference].ParallelColumns

**Legal values**

Always contains an instance of the ParallelColumns class.

**Usage**

## **Word Pro: Parent property**

{button ,AL('H\_BASEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The parent (or containing) object for the object from which you are calling the Parent property. The object stored in this property is determined by the object from which you call this property.

### **Data Type**

[BaseObject](#)

### **Syntax**

parentvalue = [objectreference].Parent

### **Legal values**

Data type for this property is BaseObject, which allows this property to contain any object derived directly or indirectly from the BaseObject class. However, this also means that you can only make use of the six properties inherited from BaseObject. For example, if the parent object in this property is a Text object, you can only access the six properties that text inherits from BaseObject.

### **Usage**

The Parent property allows you to access an object's parent object. This is useful when you need to get to the object that contains another object as a property.

For example, if you are working with a Color object and you are unsure of where the color object is contained, you can determine what object contains the color object by accessing that Color object's Parent property. Once you have the name of the object in the Parent property, you can assign that object to a variable with the same data type as the object. This will give you complete access to that object and all its members.

## **Word Pro: PowerFields property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POWERFIELDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the PowerFieldCollection class. This object provides access to PowerField objects.

### **Data Type**

[PowerFieldCollection](#)

### **Syntax**

powerfieldsvalue = [objectreference].PowerFields

### **Legal values**

Always contains an instance of the PowerFieldCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the PowerField objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the PowerField objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the PowerField objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the PowerField objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: Preferences property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PREFERENCES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The preferences object for the currently active session of Word Pro.

### **Data Type**

[Preferences](#)

### **Syntax**

preferencesvalue = [objectreference].Preferences

### **Legal values**

Always contains an instance of the Preferences class.

### **Usage**

The properties in this object contain the settings seen in the Word Pro Preferences dialog box. You can open this dialog box in Word Pro by choosing File - User Setup - Word Pro Preferences.

## **Word Pro: Presentation property**

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRESENTATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the presentation object of any container.

### **Data Type**

Presentation

### **Syntax**

presentationvalue = [objectreference].Presentation

### **Legal values**

Always contains an instance of the Presentation class.

### **Usage**

**Word Pro: PrintManager property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[PrintManager](#)

**Syntax**

printmanagervalue = [objectreference].PrintManager

**Legal values**

Always contains an instance of the PrintManager class.

**Usage**



**Word Pro: PrintSettings property**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTSETTINGS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[PrintSettings](#)

**Syntax**

printsettingsvalue = [objectreference].PrintSettings

**Legal values**

Always contains an instance of the PrintSettings class.

**Usage**

**Word Pro: Prompt property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PROMPT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The prompt text for a ClickHere object.

**Data Type**

[Text](#)

**Syntax**

promptvalue = [objectreference].Prompt

**Legal values**

Always contains an instance of the Text class.

**Usage**

### **Word Pro: RelativeIndent property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',  
0)} [See list of classes](#)

{button ,AL(^H\_RELATIVEINDENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

RelativeIndent

### **Syntax**

relativeindentvalue = [objectreference].RelativeIndent

### **Legal values**

Always contains an instance of the RelativeIndent class.

### **Usage**

**Word Pro: ReplaceAttributes property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACEATTRIBUTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enables the user to replace specific text attributes in Find & Replace.

**Data Type**

[Attributes](#)

**Syntax**

replaceattributesvalue = [objectreference].ReplaceAttributes

**Legal values**

Always contains an instance of the Attributes class. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Use this property to replace text attributes in Find & Replace. If set to True, replaces the text attributes that match the user setting. Equivalent to choosing Edit - Find & Replace Text, clicking Options, clicking the Font button in the "Replace options" section, and choosing text attributes.

## **Word Pro: ReplaceFont property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACEFONT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enables the user to replace a text font in Find & Replace.

### **Data Type**

[Font](#)

### **Syntax**

replacefontvalue = [objectreference].ReplaceFont

### **Legal values**

Always contains an instance of the Font class. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property to replace a font in Find & Replace. If True, replaces the font that matches the user setting. Equivalent to choosing Edit - Find & Replace Text, clicking Options, clicking the Font button in the "Replace options" section, and selecting a font in the "Font name" list box on the Replace with panel.

## **Word Pro: ReplaceLanguage property**

{button ,AL(^H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REPLACELANGUAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Enables the user to replace the language used in Find & Replace.

### **Data Type**

[Language](#)

### **Syntax**

replacelanguagevalue = [objectreference].ReplaceLanguage

### **Legal values**

Always contains an instance of the Language class.

### **Usage**

## **Word Pro: ReviewVersions property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_REVIEWVERSIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVIEWVERSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the ReviewVersions class which allows you to review versions and create new versions.

### **Data Type**

[ReviewVersions](#)

### **Syntax**

reviewversionsvalue = [objectreference].ReviewVersions

### **Legal values**

Always contains an instance of the ReviewVersions class.

### **Usage**

Use this property to compare different files and versions when no document is open.

## **Word Pro: RevisionDisplay property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISIONDISPLAY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[RevisionDisplay](#)

### **Syntax**

revisiondisplayvalue = [objectreference].RevisionDisplay

### **Legal values**

Always contains an instance of the RevisionDisplay class.

### **Usage**



## **Word Pro: RevisionMark property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REVISIONMARK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Revision](#)

### **Syntax**

revisionmarkvalue = [objectreference].RevisionMark

### **Legal values**

Always contains an instance of the Revision class.

### **Usage**

**Word Pro: RightBorder property**

{button ,AL(^H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RIGHTBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Border](#)

**Syntax**

rightbordervalue = [objectreference].RightBorder

**Legal values**

Always contains an instance of the Border class.

**Usage**

**Word Pro: RightContextMenu property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RIGHTMOUSEMENU\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Menus the user gets when he right clicks the mouse over a context sensitive area.

**Data Type**

[MenuItem](#)

**Syntax**

rightcontextmenuvalue = [objectreference].RightContextMenu

**Legal values**

Always contains an instance of the MenuItem class.

**Usage**

Use this class to access the right click menus when no document is open.

## Word Pro: RightPage property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_RIGHTPAGE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The right page of a complex layout. A complex layout contains separate layouts for left and right pages.

## Data Type

[PageLayout](#)

## Syntax

rightpagevalue = [objectreference].RightPage

## Legal values

Always contains an instance of the PageLayout class.

## Usage

**Word Pro: RowLayouts property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ROWLayouts\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) This object provides the names of all row layout objects within a table.

**Data Type**

[StringCollection](#)

**Syntax**

rowlayoutvalue = [objectreference].RowLayouts

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

Row layout objects exist for rows which contain non-virgin cells. A virgin cell is a cell for which a layout object has not yet been created. A cell layout object is created when the layout settings or contents of a cell are modified.

## **Word Pro: Rows property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ROWS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the RowLayoutCollection class. This object provides access to RowLayout objects.

### **Data Type**

[RowLayoutCollection](#)

### **Syntax**

rowsvalue = [objectreference].Rows

### **Legal values**

Always contains an instance of the RowLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the RowLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the RowLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the RowLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the RowLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: RubyLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUBYLAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the RubyLayoutCollection class. This object provides access to RubyLayout objects.

### **Data Type**

[RubyLayoutCollection](#)

### **Syntax**

rubylayoutsvalue = [objectreference].RubyLayouts

### **Legal values**

Always contains an instance of the RubyLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the RubyLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the RubyLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the RubyLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the RubyLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: Script property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SCRIPT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the Script class which is the Script Preferences available in the Script Editor.

**Data Type**

[Script](#)

**Syntax**

scriptvalue = [objectreference].Script

**Legal values**

Always contains an instance of the Script class.

**Usage**

Equivalent to the options available when you choose File - Script Preferences in the Script Editor.



**Word Pro: SearchAttributes property**

{button ,AL(^H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SEARCHATTRIBUTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Attributes](#)

**Syntax**

searchattributesvalue = [objectreference].SearchAttributes

**Legal values**

Always contains an instance of the Attributes class.

**Usage**

**Word Pro: SearchLanguage property**

{button ,AL(^H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SEARCHLANGUAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Language](#)

**Syntax**

searchlanguagevalue = [objectreference].SearchLanguage

**Legal values**

Always contains an instance of the Language class.

**Usage**

## **Word Pro: Sections property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SECTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the SectionCollection class. This object provides access to Section objects.

### **Data Type**

[SectionCollection](#)

### **Syntax**

sectionsvalue = [objectreference].Sections

### **Legal values**

Always contains an instance of the SectionCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Section objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the Section objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the Section objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the Section objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: SectionTabs property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SECTIONTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the SectionTabs class which manipulates divider tabs.

**Data Type**

[SectionTabs](#)

**Syntax**

sectiontabsvalue = [objectreference].SectionTabs

**Legal values**

Always contains an instance of the SectionTabs class.

**Usage**

Use this property to manipulate DividerTabs when a document is not open. For example, turn them on and off before you open a document.

## **Word Pro: SelectionBorderColor1 property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTIONBORDERCOLOR1\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Word Pro places bars and handles around any frame that you select. The bars and handles are made up of three colors that give the frame a shaded effect. SelectionBorderColor1 stores the outermost color.

### **Data Type**

[Color](#)

### **Syntax**

selectionbordercolor1value = [objectreference].SelectionBorderColor1

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

## **Word Pro: SelectionBorderColor2 property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTIONBORDERCOLOR2\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Word Pro places bars and handles around any frame that you select. The bars and handles are made up of three colors that give the frame a shaded effect. SelectionBorderColor2 stores the middle color.

### **Data Type**

[Color](#)

### **Syntax**

selectionbordercolor2value = [objectreference].SelectionBorderColor2

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

## **Word Pro: SelectionBorderColor3 property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTIONBORDERCOLOR3\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Word Pro places bars and handles around any frame that you select. The bars and handles are made up of three colors that give the frame a shaded effect. SelectionBorderColor3 stores the innermost color.

### **Data Type**

[Color](#)

### **Syntax**

selectionbordercolor3value = [objectreference].SelectionBorderColor3

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

**Word Pro: SetTabsDialog property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETTABS\_DIALOG\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the SetTabsDialog class which deals with the dialog box used to set tabs in a document.

**Data Type**

[SetTabsDialog](#)

**Syntax**

settabsdialogvalue = [objectreference].SetTabsDialog

**Legal values**

Always contains an instance of the SetTabsDialog class.

**Usage**

Use this property to manipulate the Set Tabs dialog box in conjunction with the Ruler objects created in HorzRuler and VertRuler. You can use this property to bring up the Set Tabs dialog box and select a tab.



## **Word Pro: SilverBullets property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SILVERBULLETS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the SilverBulletCollection class. This object provides access to SilverBullet objects.

### **Data Type**

[SilverBulletCollection](#)

### **Syntax**

silverbulletsvalue = [objectreference].SilverBullets

### **Legal values**

Always contains an instance of the SilverBulletCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the SilverBullet objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the SilverBullet objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the SilverBullet objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the SilverBullet objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: SilverBullet property**

{button ,AL('H\_BULLET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SILVERBULLET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[SilverBullet](#)

**Syntax**

silverbulletvalue = [objectreference].SilverBullet

**Legal values**

Always contains an instance of the SilverBullet class.

**Usage**

**Word Pro: SmartCorrects property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SMARTCORRECTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the SmartCorrectCollection class.

**Data Type**

[SmartCorrectCollection](#)

**Syntax**

smartcorrectsvalue = [objectreference].SmartCorrects

**Legal values**

Always contains an instance of the SmartCorrectCollection class.

**Usage**

## **Word Pro: SmartCorrect property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SMARTCORRECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The SmartCorrect object for the currently active session of Word Pro.

### **Data Type**

[SmartCorrect](#)

### **Syntax**

smartcorrectvalue = [objectreference].SmartCorrect

### **Legal values**

Always contains an instance of the SmartCorrect class.

### **Usage**

The properties in this SmartCorrect object contain the SmartCorrect settings for Word Pro.

**Word Pro: SnapShotSaveOptions property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SNAPSHOTSAVEOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

snapshotsoptionsvalue = [objectreference].SnapShotSaveOptions

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

**Word Pro: SortLevel1 property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTLEVEL1\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies the primary sort order for a multi-level sort.

**Data Type**

[SortKey](#)

**Syntax**

[objectreference].SortLevel1 = sortlevel1value

sortlevel1value = [objectreference].SortLevel1

**Legal values**

Always contains an instance of the SortKey class.

**Usage**

Equivalent to choosing Text - Sort and selecting any one of the options in the "Field/col.," "Type," "Order," or "Word" box in the first sort level.

**Word Pro: HidelconBars method**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIDEICONBARS\_METHOD\_EXSCRIPT',1)} [See example](#)

Temporarily closes/hides all SmartIcons bars that are currently showing, until the context changes.

**Syntax**

[objectreference].HidelconBars()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

The user interface for closing icon bar objects is found in the list that displays when you click on the file drawer adjacent to an icon bar object.

**Word Pro: HideStatusBar method**

{button ,AL('H\_STATUSBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIDESTATUSBAR\_METHOD\_EXSCRIPT',1)} [See example](#)

Hides the status bar.

**Syntax**

[objectreference].HideStatusBar()

**Parameters****Return value**

The return values for this method will always be -1 and 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## **Word Pro: HighlightToggle method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIGHLIGHTTOGGLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Turns the Review & Comment tools highlighter on or off. Equivalent to clicking the Highlighter icon on the Review & Comment tools icon bar.

### **Syntax**

[objectreference].HighlightToggle()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: Hit method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HIT\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].Hit()

**Parameters****Return value****Usage**

## **Word Pro: HourGlass method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_HOURLASS\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to show or hide the Windows hourglass cursor.

### **Syntax**

[objectreference].HourGlass(Show)

### **Parameters**

#### *Show*

A Numeric expression which allows you to specify whether or not you want the hourglass cursor to show. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. There is no default value for this parameter. A value of True will show the hourglass.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

Typically, you would use this method to hide the hourglass while your script is running.

## Word Pro: ImportGraphic method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IMPORTGRAPHIC\_METHOD\_EXSCRIPT',1)} [See example](#)

Imports a graphic into the current document. Unlike an OLE object, you cannot use the imported graphic to launch the application that created the graphic or edit the graphic in place. However, the graphic will be updated if you change the original file and resave the Word Pro file that contains the imported graphic.

### Syntax

[objectreference].ImportGraphic(FilePath, FileFormat, Link, ScratchOutFrame, [FrameStyle])

### Parameters

#### *FilePath*

A String expression which specifies the directory path and name of the file which is the source of the imported graphic.

#### *FileFormat*

A String expression which specifies the file format for the graphic you are importing. The string expression for each file format is unique and registered with Microsoft Windows 95. The values listed in the table below were valid at the time of publication.

**If you are importing this type of graphic: Use this value for the FileFormat parameter:**

WordProWMF	.wmf
WordProEPS	.eps
WordProPaint	.bmp
WordProImage	.tif
WordProDraw	.sdw
Boxes	.box
WordProEqn	.tex
WordProPCX	.pcx
WordProGIF	.gif
CGM	.cgm
HPGL	.plt
PCD	.pcd
DRW	.drw
CDR	.cdr
PIC	.pic
ImpWPG	.wpg
ExpSDW2WPG	.sdw
ExpWMF2WPG	.wmf
ExpBMP2WPG	.bmp
ImpWP2	.wp2
ExpSDW2WP2	.sdw
ExpWMF2WP2	.wmf
ExpBMP2WP2	.bmp
ImpJPG	.jpg
ExpJPG	.bmp

#### *Link*

An Integer value of -1 or 0 indicating whether the imported graphic will be receive updates from the original (-1) or will remain independent of the original (0). You can use the LotusScript constants of True (-1) and False (0) as the value for this parameter.

#### *ScratchOutFrame*

An Integer value which indicates whether you want to draw the new graphic frame by hand or let Word Pro draw the frame based on a frame style. If you want to draw the frame yourself, use the value of True (-1) for this parameter. If you want Word Pro to draw the frame based on an existing style, use a value of False (0) for this parameter.

*FrameStyle*

A String expression which specifies the frame style you want to use for the imported graphic's frame. Optional parameter. If the imported graphic is an equation and you do not specify a frame style, Word Pro will use the default equation frame style. All other imported graphics will be placed in the default GraphicOle frame style, unless you specify another frame style using this parameter.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: ImportPicture method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IMPORTPICTURE\_METHOD\_EXSCRIPT',1)} [See example](#)

Imports a graphic picture into the current document.

### **Syntax**

[objectreference].ImportPicture(Path, FileFormat, Link)

### **Parameters**

#### *Path*

Data type is String.

#### *FileFormat*

Data type is String.

#### *Link*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0).

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IndexAll method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXALL\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].IndexAll()

### **Parameters**

String Formula.

String ViceVersaFormula.

String MarkerName.

### **Return value**

Integer as Boolean.

### **Usage**

## Word Pro: InitFindAndReplace method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INITFINDANDREPLACE\_METHOD\_EXSCRIPT',1)} [See example](#)

Initializes the Find & Replace utility by clearing the registers which track the number of finds and replacements. This method also sets the options for the Find & Replace function.

### Syntax

[objectreference].InitFindAndReplace(UseUserSettings)

### Parameters

*UseUserSettings*

An Integer expression which allows you to choose between the options which were last set by the user (True) and the standard default options (False). Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False (0).

### Return value

None.

### Usage

The default Find & Replace options in Word Pro are the options that appear the first time you perform a Find & Replace during any Word Pro session. If you change these options to perform a find or replace, Word Pro sees your new options as "user settings." Your user settings remain in effect until you change them again, at which time, your new settings will take the place of your previous settings. These user settings are discarded each time you exit Word Pro.

If you provide a value of True for the UseUserSettings parameter, you are telling Word Pro to use the last options you set for Find & Replace. A value of False tells Word Pro to use the default Find & Replace settings, regardless of the options you may choose during the current session of Word Pro.



## Word Pro: InsertBreak method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTBREAK\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a break in a ClickHere block, a TextMarker, or Text object.

### Syntax

[objectreference].InsertBreak(BreakType)

### Parameters

#### *BreakType*

Data type is Variant. The value of this parameter must be one of the string values below or its integer equivalent.

\$LwpBreakTypeColumn (73)

\$LwpBreakTypeLine (74)

\$LwpBreakTypePage (72)

\$LwpBreakTypeStream (76)

\$LwpBreakTypeWord (75)

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: InsertBullet method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTBULLET\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts the bullet character you specify at the insertion point in the currently active document.

### **Syntax**

[objectreference].InsertBullet(FontName, BulletChar)

### **Parameters**

*FontName*

A String expression representing the name of the font from which you are getting the bullet character.

*BulletChar*

A String expression specifying the character used as the bullet.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: InsertClickHereLink method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTCLICKHERE\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens the Create Link dialog box so the user can insert a ClickHereLink. Equivalent to choosing Create - Click Here Block, choosing "Follow a Link" in "Behavior," and clicking Link.

### **Syntax**

[objectreference].InsertClickHereLink()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: InsertClickHere method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INSERTCLICKHERE\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a ClickHere block in a document.

**Syntax**

[objectreference].InsertClickHere()

**Parameters**

None

**Return value**

A String representing the name of the ClickHere object which was inserted.

**Usage**

## **Word Pro: InsertColumnBreak method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INSERTCOLUMNBREAK\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a column break in a document.

### **Syntax**

[objectreference].InsertColumnBreak()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

A column break breaks a page, a parallel column, or a table across a page.

## Word Pro: InsertDate method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTDATE\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a date at the insertion point. You can specify the date format with the Date parameter. Equivalent to choosing Text - Insert Other - Date/Time.

### Syntax

[objectreference].InsertDate(Date)

### Parameters

#### Date

A String expression representing the date formula. A date formula specifies the contents and format of the date you are inserting. You can use one of the preset date or time formulas (%D or %T), or you can create a custom formula (%FL or %FC). The table below illustrates the general composition of a preset Date formula, a preset Time formula, and two custom date formulas for a document that was first saved on Saturday, January 8, 1997 at 9:01:05 AM.

<u>This formula...</u>	<u>...yields this result</u>
"CreateDate %Da"	1/8/97
"CreateDate %T3"	09:01AM
"CreateDate %FLWeekday, M/D/YY, HH:mm ampm"	Saturday, 1/8/97, 09:01 AM
"CreateDate %FCWeekday, M/D/YY, HH:mm ampm"	SATURDAY, 1/8/97, 09:01 AM

Only one of the preset date formulas uses leading zeros. To get leading zeros on dates, you must create your own formula, using the "%FL" or "%FC." Both preset and custom formulas must begin with a reference to a specific date. The date includes the time.

<u>This date...</u>	<u>...yields this:</u>
Now()	The current system date and time.
CreateDate	The date and time the document was first saved.
EditDate	The date and time the document was last saved.

After choosing a date, you then add the date formula to indicate what parts of the date and time you want to display. You can use one preset date (%D) or one preset time (%T), but you cannot combine them. To display both date and time, you must use a custom date formula (%FL or %FC). When you use a custom date formula (%FC or %FL), you can include as many or as few of the custom elements as you like, in any order you like, and with any additional text or punctuation you want.

**Note** All custom formula elements are case-sensitive. Using a different case than that shown in the table above will yield unpredictable results.

All sample dates and times in the table below reflect the CreateDate for a document that was first saved on Saturday, January 8, 1997 at 9:01:05 AM.

<u>Pre-set Formula</u>	<u>Equivalent Custom Formula</u>	<u>Sample</u>
%Da	%FLM/D/YY	1/8/97
%Db	Month D, YYYY	January 8, 1997
%DB	Month D, YYYY	JANUARY 8, 1997
%Dc	D Month YYYY	8 January 1997
%DC	D Month YYYY	8 JANUARY 1997
%Dd	Weekday, Month D, YYYY	Saturday, January 8, 1997
%DD	Weekday, Month D, YYYY	SATURDAY, JANUARY 8, 1997
%De	Month D	January 8
%DE	Month D	JANUARY 8
%Df	Weekday, Month D	Saturday, January 8
%DF	Weekday, Month D	SATURDAY, JANUARY 8

%Dg	M/D	1/8
%Dh	M/D/YYYY	1/8/1997
%Di	D. Month	8. January
%DI	D. Month	8. JANUARY
%Dj	D. Month YYYY	8. January 1997
%DJ	D. Month YYYY	8. JANUARY 1997
%Dk	YYYY Month D	1997 January 8
%DK	YYYY Month D	1997 JANUARY 8
%DI	Month, YYYY	January, 1997
%DL	Month, YYYY	JANUARY, 1997
%Dm	DD/MM/YYYY	08/01/1997
%T1	HH:mm	09:01 (leading zero on hour)
%T2	H:mm ampm	9:01AM
%T3	HH:mm ampm	09:01AM (leading zero on hour)
%T4	(none)	9:01A
%T5	(none)	09:01A (leading zero on hour)
%T6	H:mm ampm	9:01am
%T7	HH:mm ampm	09:01am (leading zero on hour)
%T8	(none)	9:01a
%T9	(none)	09:01a (leading zero on hour)

**Custom formula element**

<b>Description</b>	<b>Sample</b>
%FL	Begins a mixed case date formula (after a specific date, such as CreateDate). Saturday, January 1, 1997
%FC	Begins an uppercase date formula (after a specific date, such as CreateDate). SATURDAY, JANUARY 1, 1997
SystemShortDate	Returns the date in the Short Date format which is specified in the Date panel of the Regional Settings control panel. (See the Date panel in your Regional Settings control panel)
SystemLongDate	Returns the date in the Long Date format which is specified in the Date panel of the Regional Settings control panel. (See the Date panel in your Regional Settings control panel)
ISODate1	Returns the date in the ISODate1 format. 1997/01/08
ISODate2	Returns the date in the ISODate2 format. 1997/01/08 09:01:05
Month	Returns the entire month name. January
Mn	Returns an abbreviated month name. Jan
Weekday	Returns the entire weekday name. Saturday

Wday	Returns an abbreviated weekday name.	Sat
D	Returns the day digit.	8
DD	Returns the day digit with leading zero.	08
M	Returns the month digit.	1
MM	Returns the month digit with leading zero.	01
YY	Returns the abbreviated year.	97
YYYY	Returns the entire year.	1997
H	Returns the hour.	9
HH	Returns the hour with leading zero.	09
m	Returns the minutes.	1
mm	Returns the minutes with leading zero.	01
S	Returns the seconds.	5
SS	Returns the seconds with leading zero.	05
ampm	Uses the 12-hour clock and appends the AM/PM indicator.	AM

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage



## Word Pro: InsertDocInfo method

```
{button ,AL('H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_INSERTDOCINFO_METHOD_EXSCRIPT',1)} See example
```

Inserts a document information field at the insertion point in the currently active document. This method operates in the same way, regardless of which object you call it from. The document information fields can be seen on the Fields panel of the Document Properties dialog box (choose File - Document Properties - Document).

### Syntax

```
[objectreference].InsertDocInfo(Type,[FieldName])
```

### Parameters

#### Type

The type of document information field you want to insert. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). There is no default value.

- \$LwpDocVarAllversionnames (210) The names of the different versions of the currently active document.
- \$LwpDocVarCreatedby (2046) The name of the user who created the currently active document.
- \$LwpDocVarDatecreated (196) The date the currently active document was created.
- \$LwpDocVarDatelastrevision (197) The date the currently active document was last revised.
- \$LwpDocVarDescription (195) The contents of the "Description" field for the currently active document.
- \$LwpDocVarDivisionname (203) The name of the active division in the currently active document.
- \$LwpDocVarDoccategory (212) The value of the "Document category" field.
- \$LwpDocVarDocsize (202) The size of the currently active document.
- \$LwpDocVarField (191) Use this value with the FieldName parameter to insert a custom Doc.Field which you create.
- \$LwpDocVarFilename (192) The name of the currently active document.
- \$LwpDocVarKeywords (215) The keywords listed in the "Keywords" field for the currently active document.
- \$LwpDocVarLasteditor (2047) The initials of the last user who edited the currently active document.
- \$LwpDocVarNumchars (201) The number of characters stored in the currently active document.
- \$LwpDocVarNumpages (199) The number of pages in the currently active document.
- \$LwpDocVarNumversions (209) The number of versions of the currently active document.
- \$LwpDocVarNumwords (200) The number of words in the currently active document.
- \$LwpDocVarOthereditors (2048) The initials of all the users who have edited the currently active document.
- \$LwpDocVarOtherversioneditors (207) The other editors for the currently active version of this document.
- \$LwpDocVarPath (193) The location of the currently active document.
- \$LwpDocVarSectionname (204) The name of currently active section. If the insertion point is not in a named section, this value yields no result.
- \$LwpDocVarStylesheet (194) The name of the SmartMaster used for the currently active document.
- \$LwpDocVarTotaledittime (198) The total amount of time that the currently active document has been open for editing.
- \$LwpDocVarVersioncreatedate (206) The date on which this version of the currently active document was created.
- \$LwpDocVarVersioncreatedby (205) The name of the user who created the currently active version of the active document.
- \$LwpDocVarVersionlasteditdate (213) The date on which the currently active version was last edited.
- \$LwpDocVarVersionlasteditedby (214) The name of the user who last edited the currently active version of the active document.
- \$LwpDocVarVersionname (208) The name of the currently active version of the active document.
- \$LwpDocVarVersionnumrevisions (2049) The number of revisions made to the currently active version of the active document.

\$LwpDocVarVersionremarks (211) The contents of the "Version Remarks" field for the currently active document.

*FieldName*

An optional String expression representing the name of the custom DocField that you want to insert. To insert a custom DocField, you must use \$LwpDocVarField as the value for the Type parameter.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You can call this method from the following types of objects:

- WPAplication

- ClickHere

- Text

- TextMarker

## Word Pro: InsertDocument method

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_INSERTDOCUMENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a document in the Word Pro application.

### Syntax

[objectreference].InsertDocument([FilePath,] [FileType,] [Password,] [AddToLastFileOpenList,] [Restore])

### Parameters

#### *FilePath*

An optional String expression which specifies the name and location of the document you want to insert.

#### *FileType*

An optional String expression representing the file type of the document you want to insert.

#### *Password*

An optional String expression representing the password of the document you want to insert.

#### *AddToLastFileOpenList*

Allows you to add the inserted document to the list of recently opened files. This list appears in the File menu in Word Pro. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False.

#### *Restore*

Allows you to restore the original position of the insertion point. A value of True places your insertion point at the beginning of the inserted document. A value of False leaves insertion point at end of the inserted document. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is True.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

### **Word Pro: InsertField method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTFIELD\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a Power Field in a document.

### **Syntax**

[objectreference].InsertField(Formula)

### **Parameters**

*Formula*

A String expression representing the Power Field instructions.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: InsertFootnote method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTFOOTNOTE\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a footnote at the insertion point in the currently active document.

### Syntax

[objectreference].InsertFootnote([FootnoteType])

### Parameters

#### *FootnoteType*

Use one of the values listed below to specify which type of footnote you want to insert. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpFnTypeAnyposition (289) Allows the footnote to flow with the footnote anchor.

\$LwpFnTypeAtBottomOfPage (290) Places the footnote at the bottom of the page.

\$LwpFnTypeAtEndOfDiv (293) Places the footnote at the end of the division which contains the footnote anchor.

\$LwpFnTypeAtEndOfDivisionSepDiv (294) Places the footnote in a separate division at the end of the division which contains the footnote anchor.

\$LwpFnTypeAtEndOfDivisionGroup (295) Places the footnote at the end of the division group which contains the footnote anchor.

\$LwpFnTypeAtEndOfDivGroupSepDiv (296) Places the footnote in a separate division at the end of the division group which contains the footnote anchor.

\$LwpFnTypeAtEndOfDoc (291) Places the footnote at the end of the document which contains the footnote anchor.

\$LwpFnTypeAtEndOfDocSepDiv (292) Places the footnote in a separate division at the end of the document which contains the footnote anchor.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## **Word Pro: InsertFrame method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTFRAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a frame using the dimensions you provide in the parameters.

### **Syntax**

[objectreference].InsertFrame(Width, Height, X, Y)

### **Parameters**

#### *Width*

A Numeric expression which specifies the width of the frame in Twips. Data type is Long.

#### *Height*

A Numeric expression which specifies the height of the frame in Twips. Data type is Long.

#### *X*

The position of the frame's upper left corner on the X (horizontal) axis. Data type is Long; measured in Twips.

#### *Y*

The position of the frame's upper left corner on the Y (vertical) axis. Data type of Long; measured in Twips.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: InsertHardSpace method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTHARDSPACE\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a hard space in a specified location of the document.

### **Syntax**

[objectreference].InsertHardSpace()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: InsertIndex method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTINDEX\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts an index in the currently active document. By default, this method inserts an index at the end of the document in a separate division, based on the default SmartMaster (default.mwp). The default index derives its entries from the entire document.

### Syntax

[objectreference].InsertIndex([IndexGeneration,] [IndexLocation,] [UseSeparateDivision,] [SmartMasterName])

### Parameters

#### *IndexGeneration*

Specifies the scope of the new index in terms of where the index will look for its index entries. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is \$LwpGenerateAcrossEntireDoc, which generates an index for the entire document.

\$LwpGenerateAcrossCurrentDiv (414) Generates an index using entries found in the current division.

\$LwpGenerateAcrossCurrentSect (415) Generates an index using entries found in the current section.

\$LwpGenerateAcrossEntireDoc (412) Generates an index using all the entries in the currently active document.

\$LwpGenerateAcrossGroupedDivs (413) Generates an index using entries found in the currently active group of divisions.

\$LwpGenerateAcrossSelectedText (416) Generates an index using entries found in the current selection.

#### *IndexLocation*

Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is \$LwpIndexLocationEndofdoc, which places the new index at the end of the document.

\$LwpIndexLocationEndofdivision (418) Places the new index at the end of the currently active division.

\$LwpIndexLocationEndofdoc (417) Places the new index at the end of the currently active document.

\$LwpIndexLocationEndofgroup (419) Places the new index at the end of the currently active group of divisions.

\$LwpIndexLocationEndofsection (421) Places the new index at the end of the currently active section.

\$LwpIndexLocationInsertionpoint (420) Places the new index at the insertion point.

#### *UseSeparateDivision*

A Numeric expression which allows you to place the new index in a separate division. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is True which places the new index in its own division.

#### *SmartMasterName*

An optional String expression representing the name of the SmartMaster used for the index division.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage



## **Word Pro: InsertLink method**

{button ,AL('H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTLINK\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a Dde link into a document.

### **Syntax**

[objectreference].InsertLink(MarkerName, Server, Topic, Item)

### **Parameters**

#### *MarkerName*

A String expression representing the internal name of the marker designating the link. You must create the marker before using this method. Required parameter.

#### *Server*

A String expression representing the specific server where you want to insert the link. Required parameter.

#### *Topic*

A String expression representing the link topic. Required parameter.

#### *Item*

A String expression representing the name of the item to be linked. Required parameter.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This method always uses a rich text format. Update data only value is always False (0). If you want to insert a link, the AddDdeLink method is recommended instead of this method because the additional parameters found in the AddDdeLink method give you more control.

### **Word Pro: InsertMarker method**

{button ,AL(`H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_INSERTMARKER\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts an object in a document that displays as a symbol indicating functions such as page break, column break or an inserted page layout.

### **Syntax**

[objectreference].InsertMarker(MarkerName)

[objectreference].InsertMarker(MarkerName)

### **Parameters**

*MarkerName*

A String expression representing the name of the marker you want to insert.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: InsertNote method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INSERTNOTE\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a note at the insertion point. The focus is left inside the note so the user can start typing. Equivalent to choosing Create - Comment Note.

### **Syntax**

[objectreference].InsertNote()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: InsertNumber method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTNUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a number (integer) into a document.

### Syntax

[objectreference].InsertNumber(NumberingStyle, Char, Level, [Start])

### Parameters

#### *NumberingStyle*

Data type is Variant which must be one of the following numbering styles listed below. You can use the string or its integer equivalent as the value of this parameter.

\$LwpNumberingStyleBasic (1588)

\$LwpNumberingStyleChar (1593)

\$LwpNumberingStyleFullpitchbasic (1594)

\$LwpNumberingStyleFullpitchlowercase (1597)

\$LwpNumberingStyleFullpitchuppercase (1596)

\$LwpNumberingStyleFullpitchverbasic (1595)

\$LwpNumberingStyleLowercaseletters (1590)

\$LwpNumberingStyleLowercaseroman (1592)

\$LwpNumberingStyleNone (1587)

\$LwpNumberingStyleUppercaseletters (1589)

\$LwpNumberingStyleUppercaseroman (1591)

#### *Char*

A Numeric expression representing the character you want to insert. You can use an integer as the numeric expression.

#### *Level*

A Numeric expression indicating which level to assign the page number. You can use an integer as the numeric expression.

#### *Start*

An optional Numeric expression that allows you to start numbering from 0. Default is 0. You must use an Integer as the numeric expression.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: InsertOleDivision method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTOLEDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts an OLE division in a Word Pro document. Equivalent to choosing Create - Division and clicking Create OLE Division.

### Syntax

[objectreference].InsertOleDivision(DivisionLocation, OleAction, ClassName, Path[, Parent][,Neighbor])

### Parameters

#### *DivisionLocation*

Specifies where the OLE division will be inserted. Data type is Variant so you can use either one of the string values or its numeric equivalent (in parentheses) as the value of this parameter.

\$LwpDivLocInsertAfterCurrentdiv (185) Inserts the OLE division before the current division.

\$LwpDivLocInsertAtInsertionPt (186) Inserts the OLE division at the insertion point, splitting the current division.

\$LwpDivLocInsertBeforeCurrentdiv (184) Inserts the OLE division after the current division.

#### *OleAction*

Allows you to specify how you want to create the OLE division contents. Data type is Variant so you can use one of the string expressions listed below or its numeric equivalent (in parentheses) as the value of this parameter.

\$LwpOleActionClipboardembedded (1604) Creates an embedded OLE object using the current contents of the Clipboard.

\$LwpOleActionClipboardlink (1605) Creates a linked OLE object using the current contents of the Clipboard.

\$LwpOleActionCreateembedded (1603) Creates an embedded OLE object using the file you specify in the Path parameter.

\$LwpOleActionCreatelink (1602) Creates a linked OLE object using the file you specify in the Path parameter.

\$LwpOleActionCreatenew (1601) Creates a new OLE object using the server application you specify in the ClassName parameter.

#### *ClassName*

A String expression which specifies the type of OLE object you are creating. You must provide this information so Word Pro knows how to create the OLE object. The type of object is expressed as the ClassID or ProgID for the application which creates that type of object. (The application used to create an OLE object is often referred to as the server application.) For example, a Lotus Freelance 96 Presentation has a ClassID of "{CF746000-94FB-101B-8C12-02608C454BFF}" and a ProgID of "FLW3Presentation."

Here are the server application IDs for some other SmartSuite application objects:

#### **1-2-3 Worksheet**

ClassID = {00045295-0000-0000-C000-000000000046}

ProgID = 123Worksheet

Launches 1-2-3 and opens an untitled worksheet.

#### **Lotus Approach 96 Report**

ClassID = {00028703-0000-0000-C000-000000000046}

ProgID = ApproachReport

Launches Approach 96 and prompts the user to select an existing database from which to create the report. Once the database is open, the Report Assistant opens and waits for the user to create the report.

#### **Lotus Approach 96 Application**

ClassID = {00028701-0000-0000-C000-000000000046}

ProgID = ApproachApplication

Launches Approach 96 and prompts the user to select an existing database.

#### **Lotus Freelance 96 Presentation**

ClassID = {CF746000-94FB-101B-8C12-02608C454BFF}

ProgID = FLW3Presentation

Launches Freelance 96 and prompts the user with the New Presentation dialog box.

#### **Lotus Freelance 96 Drawing**

ClassID = {CF746001-94FB-101B-8C12-02608C454BFF}

ProgID = FLW3Drawing

Launches Freelance 96 and opens a new presentation with one blank page.

#### **Lotus ScreenCam Movie 2.1**

ClassID = {00041920-0000-0000-C000-000000000046}

ProgID = ScreenCamMovie2

Launches ScreenCam 2.1 and displays the ScreenCam control panel for the user to start a recording.

You can find the ClassIDs and ProgIDs for other server applications in the Windows Registry for Windows 3.1 and Windows 95.

#### *Path*

A String expression which specifies the name and path of the source file you want to use for the OLE division. Use this parameter when you use \$LwpOleActionCreateembedded (1603) or \$LwpOleActionCreatelink (1602) as the value for the OleAction parameter. If you use any other value for OleAction, you must use a Null string ("") as the value for Path.

#### *Parent*

A String expression which allows you to specify the name of the division which you want to be a parent for the OLE division. If you include this parameter, Word Pro inserts the OLE division as a child division to the division you name here.

#### *Neighbor*

A String expression which allows you to specify the name of the division which you want to be a neighbor for the OLE division. If you include this parameter, Word Pro inserts the OLE division next to the division you name here.

#### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

#### **Usage**

An OLE division is seen by Word Pro as a part of the Word Pro document. However, when you click the division tab for an OLE division, Word Pro launches the server application for the division's source file.

## Word Pro: InsertOne method

{button ,AL('H\_TABRACK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTONE\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a tab in a document and allows you to indicate the following tab properties: twips position, tab type, tab leader, and relative tab type.

### Syntax

[objectreference].InsertOne(Position, TabType, LeaderType, RelativeType, AlignChar)

### Parameters

#### *Position*

Data type is Twips. Position of the tab measured in twips.

#### *TabType*

Data type is Variant. One of the four types of tabs listed below. You can use the string or its code as the value of this parameter.

\$LwpTabTypeCenter (1864)

\$LwpTabTypeLeft (1863)

\$LwpTabTypeNumeric (1866)

\$LwpTabTypeRight (1865)

#### *LeaderType*

Data type is Variant. One of the three tab leader types listed below. You can use the string or its code as the value of this parameter.

\$LwpTabLeaderDot (1857)

\$LwpTabLeaderHyphen (1856)

\$LwpTabLeaderLine (1858)

\$LwpTabLeaderNone (1855)

#### *RelativeType*

Data type is Variant. One of the three relative tab types listed below. You can use the string or its code as the value of this parameter.

\$LwpTabRelativeCenter (1862)

\$LwpTabRelativeLeft (1860)

\$LwpTabRelativeRight (1861)

#### *AlignChar*

A Numeric expression representing the tab alignment character. You must use an integer as the numeric expression.

### Return value

### Usage

## **Word Pro: InsertPageBreak method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_INSERTPAGEBREAK\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a page break in the document.

### **Syntax**

[objectreference].InsertPageBreak()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**



## Word Pro: InsertPageLayout method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTPAGELAYOUT\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a page layout at the insertion point. Use the LayoutStyleName parameter to specify the page style and the StartType parameter to specify where the new page begins.

### Syntax

[objectreference].InsertPageLayout(LayoutStyleName, [UsePrevHeaderText,] [UsePrevFooterText,] [StartType])

### Parameters

#### *LayoutStyleName*

A String expression representing the name of the page style you want to use for the new page layout.

#### *UsePrevHeaderText*

A Numeric expression indicating whether or not you want the new page to use previous header text. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False, which does not use the header text from the previous page layout.

#### *UsePrevFooterText*

A Numeric expression indicating whether or not you want the new page to use previous footer text. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False, which does not use the footer text from the previous page layout.

#### *StartType*

Specifies where you want to start the new page layout. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is \$LwpStartTypeThispage.

\$LwpStartTypeNextevenpage (1827) Starts the new page layout on the next even-numbered page.

\$LwpStartTypeNextoddpager (1826) Starts the new page layout on the next odd-numbered page.

\$LwpStartTypeNextpage (1824) Starts the new page layout on the next page.

\$LwpStartTypeThispage (1825) Starts the new page layout on the current page.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: InsertPageNumber method

```
{button ,AL('H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_INSERTPAGENUMBER_METHOD_EXSCRIPT',1)} See example
```

Inserts a page number.

### Syntax

When called from the WPAApplication class:

```
[objectreference].InsertPageNumber([NumberingStyle,] [BeforeText,] [AfterText,] [StartingNumber,] [StartOnPage,] [Flags])
```

When called from the Text class:

```
[objectreference].InsertPageNumber(NumberingStyle, BeforeText, AfterText, StartingNumber, StartOnPage, Flags)
```

### Parameters

#### *NumberingStyle*

Specifies the numbering style for the page number you are inserting. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). When called from the WPAApplication object, this is an optional parameter with a default of \$LwpNumberingStyleBasic.

\$LwpNumberingStyleBasic (1588)

\$LwpNumberingStyleChar (1593)

\$LwpNumberingStyleFullpitchbasic (1594)

\$LwpNumberingStyleFullpitchlowercase (1597)

\$LwpNumberingStyleFullpitchuppercase (1596)

\$LwpNumberingStyleFullpitchverbasic (1595)

\$LwpNumberingStyleLowercaseletters (1590)

\$LwpNumberingStyleLowercaseroman (1592)

\$LwpNumberingStyleNone (1587)

\$LwpNumberingStyleUppercaseletters (1589)

\$LwpNumberingStyleUppercaseroman (1591)

#### *BeforeText*

The text you want Word Pro to place before the page number. Data type is String. When called from the WPAApplication object, this is an optional parameter.

#### *AfterText*

The text you want Word Pro to place after the page number. Data type is String. When called from the WPAApplication object, this is an optional parameter.

#### *StartingNumber*

An Integer which specifies the starting page number. For example, if you use 5 as the value for this parameter, Word Pro will use 5 as the first page number, regardless of the page on which you place the first page number. When called from the WPAApplication object, this is an optional parameter with a default of 0.

#### *StartOnPage*

An Integer which specifies the page on which the starting page number will appear. For example, if you use 1 as the value for this parameter, Word Pro will place the starting page number (specified in StartingNumber parameter) on the first page in your document. When called from the WPAApplication object, this is an optional parameter with a default of 0.

#### *Flags*

Data type is Variant which allows the value of this parameter to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these constants when you want Word Pro to combine the features listed below. Use the OR operator to combine constants. When called from the WPAApplication object, this is an optional parameter with a default of LwpPageNumberFlagsDefault.

LwpPageNumberFlagsDefault (&H0) Runs the page numbers from the starting page to the end of the document.

LwpPageNumberFlagsIncludebefore (&H4) Runs the page numbers for the entire document, regardless of where

the insertion point is.

LwpPageNumberFlagsIncludedivname (&H2) Includes the division name with the page number.

LwpPageNumberFlagsIncludesection (&H1) Includes the section name with the page number.

LwpPageNumberFlagsResetondivision (&H20) Resets the page numbers at the beginning of each new division.

LwpPageNumberFlagsResetonsection (&H10) Resets the page numbers at the beginning of each new section.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

Call this method from the WPAApplication object when you want to insert page numbers into the currently active document. If you accept all the default values for the parameters, Word Pro starts the page numbers with 1 on the first page of your document and no text before or after the page number. The page numbers will continue throughout the document.

Call this method from the Text object when you want to insert page numbers into a specific Text object in a specific place.

This method appears on TextMarker and ClickHere objects because these objects inherit the method from the Marker class. Do not call this method from a TextMarker or ClickHere object.

## Word Pro: InsertPath method

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTPATH\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts path(s) or file(s) in the Word Pro Preferences dialog box.

### Syntax

[objectreference].InsertPath(PathSelection, Path, [Path2],[Path3],[Path4],[Path5])

### Parameters

#### *PathSelection*

Indicates which path or file you are setting. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpSetDocumentsPath = 2081

\$LwpSetSmartmasterPath = 2082

\$LwpSetScriptPath = 2083

\$LwpSetSmarticonPath = 2084

\$LwpSetBackupPath = 2085

\$LwpSetUserdictPath = 2086

\$LwpSetUserdictFile = 2087

\$LwpSetGlossaryPath = 2088

\$LwpSetGlossaryFile = 2089

#### *Path*

A String expression.

#### *Path2*

A String expression. Optional parameter.

#### *Path3*

A String expression. Optional parameter.

#### *Path4*

A String expression. Optional parameter.

#### *Path5*

A String expression. Optional parameter.

### Return value

Data type is Boolean.

### Usage

## Word Pro: InsertRowOrColumn method

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INSERTROWORCOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a new row or column into a table object.

### Syntax

[objectreference].InsertRowOrColumn(TableInsType, InsertAfter, NumToInsert, [Position,] [CopyCellStyle])

### Parameters

#### *TableInsType*

The value of this Variant parameter must be one of the strings below or its numeric equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpTableInsTypeRow (1875)	Indicates that the method should insert rows.
\$LwpTableInsTypeColumn (1876)	Indicates that the method should insert columns.

#### *InsertAfter*

A Boolean expression that controls whether the new row or column will be inserted after the row or column specified in the Position parameter. If no value is specified in the Position parameter, the rows or columns will be inserted in relation to the current selection.

#### *NumToInsert*

An Integer value which specifies the number of rows or columns to insert.

#### *Position*

An optional Integer value which specifies the ID of the row or column next to which new items should be inserted. Default is the current row or column ID.

#### *CopyCellStyle*

A Boolean value which specifies whether or not to copy the cell style from the row or column specified in the Position parameter. If no value is specified for the Position parameter, the cell style of the row or column in the current selection will be used. Optional parameter. Default is True.

### Return value

Boolean.

### Usage

**Word Pro: InsertRuby method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTRUBY\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is implemented only in the Asian-language versions of Word Pro.

## Word Pro: InsertSection method

{button ,AL('H\_SECTIONTABS\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTSECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a new section marker at the insertion point in the currently active division of a document. When called from a SectionTabs object, this method accepts no parameters.

### Syntax

From the WPAApplication object:

[objectreference].InsertSection([StyleName,][UsePrevHeaderText,] [UsePrevFooterText,] [StartType,] [IsCreateIndex,] [ShowTab])

From a SectionTabs object:

[objectreference].InsertSection()

### Parameters

#### *StyleName*

A String expression which specifies the name of the page style you want to use for the new section. Optional parameter. If you do not provide a page style name, Word Pro will use the page style on the currently active page.

#### *UsePrevHeaderText*

Allows you to use the header text from the previous section or start a new header. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False.

#### *UsePrevFooterText*

Allows you to use the footer text from the previous section or start a new footer. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False.

#### *StartType*

An optional parameter which allows you to specify how the new section will begin. Default is \$LwpStartTypeThispage. Data type is Variant which allows you to use one of the string values below or its numeric equivalent (in parentheses).

\$LwpStartTypeNextevenpage (1827) Starts the new section on the next even-numbered page.

\$LwpStartTypeNextoddpager (1826) Starts the new section on the next odd-numbered page.

\$LwpStartTypeNextpage (1824) Starts the new section on the next page.

\$LwpStartTypeThispage (1825) Starts the new section on the currently active page.

#### *IsCreateIndex*

Allows you to indicate whether the new section is an index section or a normal section. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False.

#### *ShowTab*

Allows you to show or hide the new section's tab. A value of True will show the tab while a value of False will hide the tab. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: InsertSpecialTab method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTSPECIALTAB\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a tab in the horizontal ruler.

### Syntax

[objectreference].InsertSpecialTab([ p1,] [TabType,] [LeaderType,] [RelativeType,] [AlignChar])

### Parameters

*P1*

Data type is Variant. Optional parameter.

*TabType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTabTypeCenter (1864)

\$LwpTabTypeLeft (1863)

\$LwpTabTypeNumeric (1866)

\$LwpTabTypeRight (1865)

*LeaderType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTabLeaderDot (1857)

\$LwpTabLeaderHyphen (1856)

\$LwpTabLeaderLine (1858)

\$LwpTabLeaderNone (1855)

*RelativeType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTabRelativeCenter (1862)

\$LwpTabRelativeLeft (1860)

\$LwpTabRelativeRight (1861)

*AlignChar*

A Numeric expression. Optional parameter. The number must be an integer.

### Return value

### Usage



## Word Pro: InsertTab method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTTAB\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a tab in the horizontal ruler.

### Syntax

[objectreference].InsertTab([ p1,] [TabType,] [LeaderType,] [RelativeType,] [AlignChar])

### Parameters

*P1*

Data type is Variant. Optional parameter.

*TabType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTabTypeCenter (1864)

\$LwpTabTypeLeft (1863)

\$LwpTabTypeNumeric (1866)

\$LwpTabTypeRight (1865)

*LeaderType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTabLeaderDot (1857)

\$LwpTabLeaderHyphen (1856)

\$LwpTabLeaderLine (1858)

\$LwpTabLeaderNone (1855)

*RelativeType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTabRelativeCenter (1862)

\$LwpTabRelativeLeft (1860)

\$LwpTabRelativeRight (1861)

*AlignChar*

A Numeric expression. Optional parameter. The number must be an Integer.

### Return value

### Usage

## Word Pro: InsertText method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts text in a document.

### Syntax

[objectreference].InsertText(Text, [Split,] [TextType,])

### Parameters

#### *Text*

Data type is String.

#### *Split*

Allows you to create a new paragraph at the end of the inserted text. Data type is Boolean. Optional parameter. Default is False.

#### *TextType*

Data type is Variant. Optional parameter. Default value is "\$LwpTextTypeRoman." The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTextTypeHkatakana (1939)

Single-byte Japanese.

\$LwpTextTypeKanji (1938)

Double-byte kanji Japanese.

\$LwpTextTypeNative (2014)

we try to figure out what this is

\$LwpTextTypeRawUnicode (1940)

What can be converted to the native Windows.

\$LwpTextTypeRoman (1937)

English text.

\$LwpTextTypeUnicode (1936)

Unicode.

### Return value

### Usage

## Word Pro: InsertTOC method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTTOC\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a Table of Contents (TOC) in the currently active document. By default, this method inserts a TOC at the beginning of the document in a separate division based on the default SmartMaster (default.mwp). The default TOC derives its entries from the entire document.

### Syntax

[objectreference].InsertTOC([TOCGeneration,] [TOCPlacement,] [UseSeparateDivision,] [SmartMasterName])

### Parameters

#### *TOCGeneration*

Specifies the scope of the new TOC in terms of where the TOC will look for its TOC entries. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). Default is \$LwpGenerateAcrossEntireDoc, which generates a TOC for the entire document.

\$LwpGenerateAcrossCurrentDiv (414) Generates a TOC using entries found in the current division.

\$LwpGenerateAcrossCurrentSect (415) Generates a TOC using entries found in the current section.

\$LwpGenerateAcrossEntireDoc (412) Generates a TOC using all the entries in the currently active document.

\$LwpGenerateAcrossGroupedDivs (413) Generates a TOC using entries found in the currently active group of divisions.

\$LwpGenerateAcrossSelectedText (416) Generates a TOC using entries found in the current selection.

#### *TOCPlacement*

Data type is Variant. Optional parameter. Default is \$LwpTOCPlacementBeginofdoc. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTOCPlacementBeginofdivision (1842) Places the new TOC at the beginning of the currently active division.

\$LwpTOCPlacementBeginofdoc (1841) Places the new TOC at the beginning of the currently active document.

\$LwpTOCPlacementBeginofgroup (1843) Places the new TOC at the beginning of the currently active group of divisions.

\$LwpTOCPlacementBeginofsection (1845) Places the new TOC at the beginning of the currently active section.

\$LwpTOCPlacementInsertionpoint (1844) Places the new TOC at the insertion point in the currently active document.

#### *UseSeparateDivision*

A Numeric expression which allows you to place the new TOC in a separate division. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is True, which places the new TOC in its own division.

#### *SmartMasterName*

An optional String expression representing the name of the SmartMaster used for the TOC division.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: InternalCopy method

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS',0)}

[See list of classes](#)

{button ,AL(^H\_INTERNALCOPY\_METHOD\_EXSCRIPT',1)} [See example](#)

Copies the selected items into either the AppFoundry or the TempFoundry property on WPAApplication. Unlike the Copy command found on the Edit menu, the selected items are not placed in the external Windows Clipboard.

### Syntax

[objectreference].InternalCopy(Temporary)

### Parameters

*Temporary*

Optional parameter which uses a Boolean Integer value to indicate whether you want the selected items copied into the temporary Foundry object in the TempFoundry property (True or -1), or the standard application Foundry object, located in the AppFoundry property on WPAApplication (False or 0). Default of False (0) sends all the copied items to the AppFoundry property.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

The Foundry object located in the AppFoundry property is the usual place where things are stored during the copy process. Word Pro automatically clears the contents of AppFoundry and TempFoundry each time you use the InternalCopy method.

## Word Pro: InternalCut method

```
{button ,AL('H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_INTERNALCUT_METHOD_EXSCRIPT',1)} See example
```

Deletes the selected items from the document and places them in the AppFoundry or TempFoundry property on WPAApplication. Unlike the Cut command found on the Edit menu, the selected items are not placed in the external Windows Clipboard.

### Syntax

```
[objectreference].InternalCut(Temporary)
```

### Parameters

#### *Temporary*

Optional parameter which uses a Boolean integer value to indicate whether you want the selected items copied into the temporary Foundry object in the TempFoundry property (True or -1), or the standard application Foundry object, located in the AppFoundry property on WPAApplication (False or 0). Default of False (0) sends all the copied items to the AppFoundry property.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

The Foundry object located in the AppFoundry property is the usual place where things are stored during the copy process. Word Pro automatically clears the contents of AppFoundry and TempFoundry each time you use the InternalCut method.

## Word Pro: InternalPaste method

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS',0)}

[See list of classes](#)

{button ,AL(^H\_INTERNALPASTE\_METHOD\_EXSCRIPT',1)} [See example](#)

Pastes items from the specified Foundry object into the active document. Unlike the Paste command found on the Edit menu, the pasted items are drawn from the Foundry object, not the external Windows Clipboard.

### Syntax

[objectreference].WPApplication.InternalPaste(FoundryType)

[objectreference].ClickHere.InternalPaste([FoundryType])

[objectreference].TextMarker.InternalPaste([FoundryType])

[objectreference].Text.InternalPaste([FoundryType])

### Parameters

#### *FoundryType*

A String or Integer value used to specify the Foundry object used as the source for the paste operation. Data type is Variant which allows you to use one of the strings below or its numeric equivalent (in parentheses).

\$LwpFoundryTypeApplication (346) Specifies the AppFoundry property on WPApplication as the source for the paste.

\$LwpFoundryTypeDocument (345) Specifies the Foundry property on Division as the source for the paste.

\$LwpFoundryTypeTemporary (347) Specifies the TempFoundry property on WPApplication as the source for the paste.

When you call InternalPaste from WPApplication, there is no default value. You must provide one of the values listed above. However, when you call InternalPaste from a ClickHere, Text, or TextMarker object, this parameter uses the default value of \$LwpFoundryTypeApplication. You can override the default if you choose.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: InvalidateButton method

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INVALIDATEBUTTON\_METHOD\_EXSCRIPT',1)} [See example](#)

Cause the status bar button to be repainted (refreshed). This method forces the repaint by processing the written routine.

### Syntax

[objectreference].InvalidateButton()

### Parameters

### Return value

Integer. Always returns True.

### Usage

You can change or update the button. When the button is invalidated, an event is emitted. If it is a text button (LwpButtonTypeText), the StatusBarButtonOverrideText event occurs. If it is a graphic button (LwpButtonTypeGraphics), the StatusBarButtonOverrideGraphic event occurs. If it is a text and graphic button, the StatusBarButtonOverrideTextAndGraphic event occurs.

If you do not respond to the events for a custom button that has been invalidated, it will be blank.

**Note** If the text on the status bar button is never going to change, you can use the LwpButtonNoTextFromHost (&H800) parameter when the button is created.

**Word Pro: InvalidateWholeBar method**

{button ,AL('H\_STATUSBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INVALIDATEWHOLEBAR\_METHOD\_EXSCRIPT',1)} [See example](#)

Repaints (refreshes) the entire status bar.

**Syntax**

[objectreference].InvalidateWholeBar()

**Parameters****Return value**

The return values for this method will always be -1 and 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Use this method after you add or delete a button on the status bar to invalidate or refresh the bar, so that the new addition or deletion can display.



**Word Pro: Invalidate method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INVALIDATE\_METHOD\_EXSCRIPT',1)} [See example](#)

**Syntax**

[objectreference].Invalidate()

**Parameters****Return value****Usage**

**Word Pro: IsDataNameUsed method**

{button ,AL('H\_SCRIPTDATASET\_CLASS;H\_WPDATASET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDATANAMEUSED\_METHOD\_EXSCRIPT',1)} [See example](#)

Determines if the data name for an object is being used.

**Syntax**

[objectreference].IsDataNameUsed(DataName)

**Parameters**

*DataName*

A variable name in the data set. DataName is a string expression.

**Return value**

Integer. This method returns True (-1) if the data name for an object is being used. This method returns False (0) if the data name for an object is not being used.

**Usage**

This method allows you to see if the data set is using a particular data name.

## Word Pro: IsEmpty method

```
{button ,AL('H_BAGCOLLECTION_CLASS;H_BASECOLLECTION_CLASS;H_BASETABLE_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_CELLCOLLECTION_CLASS;H_CELLLAYOUTCOLLECTION_CLASS;H_CHARACTERSTYLECOLLECTION_CLASS;H_CLICKHERE_CLASS;H_CLICKHERECOLLECTION_CLASS;H_CONNECTEDLAYOUTCOLLECTION_CLASS;H_CONTENT_CLASS;H_CONTENTCOLLECTION_CLASS;H_DDELINKCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCINFOFIELDCOLLECTION_CLASS;H_DOCUMENTS_CLASS;H_DOCWINDOWCOLLECTION_CLASS;H_DROPCAPLAYOUTCOLLECTION_CLASS;H_EDITORCOLLECTION_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECTION_CLASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTION_CLASS;H_FOOTNOTETABLE_CLASSES;H_FORMULA_CLASS;H_FRAMELAYOUTCOLLECTION_CLASS;H_GLOSSARY_CLASS;H_GLOSSARYCOLLECTION_CLASS;H_GRAPHIC_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHICOLEOBJECT_CLASS;H_GRAPHICOLEOBJECTCOLLECTION_CLASS;H_GROUPLAYOUTCOLLECTION_CLASS;H_HEADERLAYOUTCOLLECTION_CLASS;H_ICONBARCOLLECTION_CLASS;H_LAYOUTCOLLECTION_CLASS;H_MARKERCOLLECTION_CLASS;H_MENUITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLASS;H_OLEOBJECT_CLASS;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLINESEQCOLLECTION_CLASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELAYOUTCOLLECTION_CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARALLELCOLSCOLLECTION_CLASS;H_PARALLELCOLUMNS_CLASS;H_POWERFIELDCOLLECTION_CLASS;H_ROWLAYOUTCOLLECTION_CLASS;H_RUBYLAYOUTCOLLECTION_CLASS;H_SECTIONCOLLECTION_CLASS;H_SILVERBULLETCOLLECTION_CLASS;H_SMARTCORRECTCOLLECTION_CLASS;H_SMARTFILLCOLLECTION_CLASS;H_STATUSBARBUTTONCOLLECTION_CLASS;H_STRINGCOLLECTION_CLASS;H_SUPERTABLE_CLASS;H_SUPERTABLECOLLECTION_CLASS;H_SUPERTABLELAYOUTCOLLECTION_CLASSES;H_TABLE_CLASS;H_TABLECOLLECTION_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGCOLLECTION_CLASS;H_TABLEHEADINGLAYOUTCOLLECTION_CLASS;H_TABLELAYOUTCOLLECTION_CLASS;H_TABLEMARKERCOLLECTION_CLASS;H_TABLEONLYCOLLECTION_CLASS;H_TEXT_CLASS;H_TEXTCOLLECTION_CLASS;H_TEXTMARKERCOLLECTION_CLASS;H_TEXTSTYLECOLLECTION_CLASS;H_VERSIONCOLLECTION_CLASS;H_WPDATASETCOLLECTION_CLASS',0)} See list of classes
```

```
{button ,AL('H_IEMPTY_METHOD_EXSCRIPT',1)} See example
```

Indicates whether or not a collection contains any item.

### Syntax

```
[objectreference].IsEmpty()
```

### Parameters

### Return value

Data type of Long. The legal values for this property are 1 and 0 (not -1 and 0). If you prefer, you can use LotusScript constants of True (1) and False (0) instead of the integer values.

### Usage

Use this method to determine whether or not a specific collection class contains any items. If the method returns True (1), then the collection class contains items. If the method returns False (0), then the collection class does not contain items.

## Word Pro: IsExportedAsNotesFX method

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISEXPORTEDASNOTESFX\_METHOD\_EXSCRIPT',1)} [See example](#)

Returns whether or not the specified DocInfo field is exported as a Notes FX file.

### Syntax

[objectreference].IsExportedAsNotesFX(Type)

### Parameters

*prexType*

You can return one of the Variant data types below to determine if it is a DocInfo field exported as a NotesFX file. The value of this parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpDocVarAllversionnames (210)	The names of all versions of a document.
\$LwpDocVarCreatedby (2046)	The Word Pro user name of the person who first saved the document.
\$LwpDocVarDatecreated (196)	The date and time the document was first created.
\$LwpDocVarDatelastrevision (197)	The last time the document was opened, edited, and saved.
\$LwpDocVarDescription (195)	Displays a description of the document.
\$LwpDocVarDivisionname (203)	The name of a division within the document.
\$LwpDocVarDoccategory (212)	The assigned category of the division or document.
\$LwpDocVarDocsize (202)	The size of the document in kilobytes.
\$LwpDocVarField (191)	Specifies the DocInfo field you want to export as Notes FX.
\$LwpDocVarFilename (192)	The file name you specified in the Save As dialog box when you saved the document.
\$LwpDocVarKeywords (215)	Displays the assigned keywords for the document.
\$LwpDocVarLasteditor (2047)	The initials of the editor who last saved the document.
\$LwpDocVarNone (190)	No DocInfo fields are set or reset as Notes FX.
\$LwpDocVarNumchars (201)	The number of characters in the document.
\$LwpDocVarNumpages (199)	The number of pages in the document.
\$LwpDocVarNumversions (209)	The number of versions in the document.
\$LwpDocVarNumwords (200)	The number of words in the document.
\$LwpDocVarOthereditors (2048)	The Word Pro user names of other people who saved the document.
\$LwpDocVarOtherversioneditors (207)	The Word Pro user names of other people who saved a version of the document.
\$LwpDocVarPath (193)	The drive and folder where the document is located.
\$LwpDocVarSectionname (204)	The name of a section in the document.
\$LwpDocVarStylesheet (194)	The style sheet used in the document.
\$LwpDocVarTotaledittime (198)	The total number of minutes the document was open.
\$LwpDocVarVersioncreatedate (206)	The date and time the version was first created.
\$LwpDocVarVersioncreatedby (205)	The Word Pro user name of the person who first saved the version of the document.
\$LwpDocVarVersionlasteditdate (213)	The date and time that the version was last saved.

\$LwpDocVarVersionlasteditedby (214)	The initials of the editor who last saved the version.
\$LwpDocVarVersionname (208)	The name of the current version.
\$LwpDocVarVersionnumrevisions (2049)	The number of times the version was opened, edited, and saved.
\$LwpDocVarVersionremarks (211)	Any remarks written by an editor to be reviewed by other assigned editors.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsFilterTypePresent method**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISFILTERTYPEPRESENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Determine whether or not the specified filter resides in Word Pro.

### **Syntax**

[objectreference].IsFilterTypePresent(Type, Import)

### **Parameters**

*Type*

A String expression

*Import*

A Numeric expression. You must use an Integer as the numeric expression.

### **Return value**

### **Usage**

## **Word Pro: IsMarkerEqualToSelection method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISMARKEREQUALTOSELECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Determines whether or not the current marker matches the current selection.

### **Syntax**

[objectreference].IsMarkerEqualToSelection(Marker)

### **Parameters**

*Marker*

A string expression

### **Return value**

### **Usage**

## Word Pro: IsPointWithin method

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_CLICKHERE\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASSES;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASS;H\_TABLEONLYCONT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPOINTWITHIN\_METHOD\_EXSCRIPT',1)} [See example](#)

Determines whether or not the specified point is located within the selection in the object from which this method is called.

### Syntax

[objectreference].IsPointWithin( X, Y)

### Parameters

X

Data type is Twips. Indicates the position of the point in units of Twips on the X axis.

Y

Data type is Twips. Indicates the position of the point in units of Twips on the Y axis.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage



## **Word Pro: IsTemporary method**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISTEMPORARY\_METHOD\_EXSCRIPT',1)} [See example](#)

Retrieves the value of the IsTemp property for an object.

### **Syntax**

[objectreference].IsTemporary()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: IsWMCommandValid method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISWMCOMMANDVALID\_METHOD\_EXSCRIPT',1)} [See example](#)

Determines whether the specified Word Pro menu item is available for use at the time this method is called.

### Syntax

[objectreference].IsWMCommandValid(CommandID)

### Parameters

*CommandID*

A Numeric expression (or constant) which specifies the ID for the menu item you are checking. Data type is Integer but you can use the appropriate constant for the menu ID. A complete list of menu ID constants is listed under [Word Pro Menu Command IDs](#)

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the menu item is valid or invalid respectively.

### Usage

### **Word Pro: Italic method**

{button ,AL('H\_FONT\_CLASS;H\_FONTMETRICS\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ITALIC\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the italic attribute for selected text, or all following text if no text is selected. Acts as a toggle, turning the attribute off if it is on and on if it is off. Equivalent to choosing Text - Attributes - Italic.

### **Syntax**

[objectreference].Italic()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: Item method

```
{button ,AL('H_BAGCOLLECTION_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_CELLCOLLECTION_CLASS;  
H_CELLLAYOUTCOLLECTION_CLASS;H_CHARACTERSTYLECOLLECTION_CLASS;H_CLICKHERECOLLEC  
TION_CLASS;H_CONNECTEDLAYOUTCOLLECTION_CLASS;H_CONTENTCOLLECTION_CLASS;H_DDELIN  
KCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCINFOFIELDCOLLECTION_CLASS;H_DOC  
UMENTS_CLASS;H_DOCWINDOWCOLLECTION_CLASS;H_DROPCAPLAYOUTCOLLECTION_CLASS;H_EDI  
TORCOLLECTION_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECTION_C  
LASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTION_CLASS;H_FRAMELAYOUT  
COLLECTION_CLASS;H_GLOSSARYCOLLECTION_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHIC  
OLEOBJECTCOLLECTION_CLASS;H_GROUPLAYOUTCOLLECTION_CLASS;H_HEADERLAYOUTCOLLECTI  
ON_CLASS;H_ICONBARCOLLECTION_CLASS;H_LAYOUTCOLLECTION_CLASS;H_MARKERCOLLECTION_  
CLASS;H_MENUITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLASS;H_OLEOBJECTCOLLEC  
TION_CLASS;H_OUTLINESEQCOLLECTION_CLASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELA  
YOUTCOLLECTION_CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARALLELCOLSCOLLECTION_  
CLASS;H_POWERFIELDCOLLECTION_CLASS;H_ROWLAYOUTCOLLECTION_CLASS;H_RUBYLAYOUTCOL  
LECTION_CLASS;H_SECTIONCOLLECTION_CLASS;H_SILVERBULLETCOLLECTION_CLASS;H_SMARTCO  
RRECTCOLLECTION_CLASS;H_SMARTFILLCOLLECTION_CLASS;H_STATUSBARBUTTONCOLLECTION_CL  
ASS;H_STRINGCOLLECTION_CLASS;H_SUPERTABLECOLLECTION_CLASS;H_SUPERTABLELAYOUTCOLL  
ECTION_CLASS;H_TABLECOLLECTION_CLASS;H_TABLEHEADINGCOLLECTION_CLASS;H_TABLEHEADIN  
GLAYOUTCOLLECTION_CLASS;H_TABLELAYOUTCOLLECTION_CLASS;H_TABLEMARKERCOLLECTION_C  
LASS;H_TABLEONLYCOLLECTION_CLASS;H_TEXTCOLLECTION_CLASS;H_TEXTMARKERCOLLECTION_C  
LASS;H_TEXTSTYLECOLLECTION_CLASS;H_VERSIONCOLLECTION_CLASS;H_WPDATASETCOLLECTION  
_CLASS',0)} See list of classes
```

```
{button ,AL('H_ITEM_METHOD_EXSCRIPT',1)} See example
```

Returns an item from a collection class. This method is defined as a part of every collection class. In all but a few collection classes, the return value depends on the type of item stored in a particular collection.

## Syntax

Item(idx)

Item(Name)

Item(ButtonName)

## Parameters

*idx*

Idx represents index. The index specifies which item in a collection you want to return.

A Long data type index specifies the numeric position of an index item in a collection. For example, items in an array are accessed by number (subscript). Therefore, you would use a Long data type index to return an index item in an array.

A String data type index item returns the string name of an index item in a collection. For example, items in a list are accessed by string name rather than by number (subscript). Therefore, you would use a String data type index to return an index item in a list.

The String idx data type is the parameter used in the Item method for most collection classes, with the exception of six classes. There are five classes that use the Long idx data type parameter. They are: DddLinkCollection, LongCollection, StringCollection, UnitCollection, VersionCollection. The final exception is the SmartFillCollection class which uses the Integer idx parameter.

*Name*

This parameter is used in the Item method on the EditorCollection class. Data type is String.

*ButtonName*

This parameter is used in the Item method on the StatusBarButtonCollection class. Data type is String.

## Return value

Each collection class returns a different data type.

BagCollection returns

BookmarkCollection returns

CellCollection returns

CellLayoutCollection returns

CharacterStyleCollection returns  
ClickHereCollection returns  
ConnectedLayoutCollection returns  
ContentCollection returns  
ScriptDataSetCollection returns  
DdeLinkCollection returns  
DivisionCollection returns  
Documents returns  
EditorCollection returns  
EndnoteLayoutCollection returns  
DocInfoFieldCollection returns  
PowerFieldCollection returns  
FooterLayoutCollection returns  
FootnoteCollection returns  
FootnoteLayoutCollection returns  
FrameLayoutCollection returns  
GlossaryCollection returns  
GraphicCollection returns  
GraphicOleObjectCollection returns  
GroupLayoutCollection returns  
HeaderLayoutCollection returns  
IconBarCollection returns  
LayoutCollection returns  
LongCollection returns  
MarkerCollection returns  
MenuItemCollection returns  
NoteLayoutCollection returns  
OleObjectCollection returns  
OutlineSeqCollection returns  
OutlineSeqItemCollection returns  
PageLayoutCollection returns  
ParagraphStyleCollection returns  
ParallelColsCollection returns  
RowLayoutCollection returns  
RubyLayoutCollection returns  
SectionCollection returns  
SilverBulletCollection returns  
StatusBarButtonCollection returns  
StringCollection returns  
SuperTableCollection returns  
SuperTableLayoutCollection returns  
TableCollection returns  
TableHeadingCollection returns  
TableHeadingLayoutCollection returns  
TableLayoutCollection returns  
TableMarkerCollection returns  
TableOnlyCollection returns

TextCollection returns

TextMarkerCollection returns

TextStyleCollection returns

UnitCollection returns

DocWindowCollection returns

## **Usage**

## **Word Pro: Link method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LINK\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Link(Path,Type)

### **Parameters**

#### *Path*

A String expression.

#### *Type*

A String expression.

### **Return value**

### **Usage**

## **Word Pro: Localize method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_LOCALIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Localize(SetToLocal)

### **Parameters**

*SetToLocal*

A Boolean expression, either True or False.

### **Return value**

### **Usage**



## **Word Pro: LowerCase method**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_LOWERCASE\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the lowercase attribute for selected text or all following text, if no text is selected. Acts as a toggle, turning the attribute off if it is on and on if it is off. Equivalent to choosing Text - Attributes - Other and then "Lower Case" in the Attributes box.

### **Syntax**

[objectreference].LowerCase()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: MacroAppend method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MACROAPPEND\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].MacroAppend()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: MacroCancel method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MACROCANCEL\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].MacroCancel()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: MacroCompile method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MACROCOMPILE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].MacroCompile()

### **Parameters**

### **Return value**

### **Usage**

**Word Pro: MacroEndRecord method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_MACROENDRECORD\_METHOD\_EXSCRIPT',1)} [See example](#)

Stops the Script Recorder in Word Pro.

**Syntax**

[objectreference].MacroEndRecord()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: MacroPlay method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_MACROPLAY\_METHOD\_EXSCRIPT',1)} [See example](#)

Plays a converted Ami Pro macro.

### **Syntax**

[objectreference].MacroPlay(MacroFileName)

### **Parameters**

*MacroFileName*

A String expression specifying the path and name of the Ami Pro macro file you want to play.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: MacroQuickPlay method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACROQUICKPLAY\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to play the most recently recorded Ami Pro quick macro.

### **Syntax**

[objectreference].MacroQuickPlay()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: MacroQuickRecord method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_MACROQUICKRECORD\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you record an Ami Pro macro without saving the keystrokes to a named macro file.

### **Syntax**

[objectreference].MacroQuickRecord()

### **Parameters**

### **Return value**

### **Usage**



## Word Pro: MacroRecord method

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MACRORECORD\_METHOD\_EXSCRIPT',1)} [See example](#)

[WPAApplication]

Allows you to create a script and save it to a file.

[ApplicationWindow]

Allows you to record a script and display it in the Script Editor.

### Syntax

WPAApplication.MacroRecord( MacroFileName,[SubroutineName])

ApplicationWindow.MacroRecord([MacroFileName,] [SubroutineName,] [InsertIntoScriptEditor])

### Parameters

#### *MacroFileName*

Data type is String. The filename must be a valid .LWP or .LSS file.

[ApplicationWindow]

Optional parameter.

#### *SubroutineName*

Data type is String. Optional parameter. If you specify this parameter, you must also specify the MacroFileName parameter.

#### *InsertIntoScriptEditor*

Data type is Boolean. Optional parameter. Default is False.

### Return value

Integer. The return values for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

This method corresponds to the Record Script dialog box. Equivalent to choosing Edit - Script & Macros - Record Script.

## **Word Pro: MacroResume method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MACRORESUME\_METHOD\_EXSCRIPT',1)} [See example](#)

Resumes the playing of an Ami Pro macro in Word Pro.

### **Syntax**

[objectreference].MacroResume()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: MacroRun method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_MACRORUN\_METHOD\_EXSCRIPT',1)} [See example](#)

Plays an Ami Pro macro in the Word Pro application.

### **Syntax**

[objectreference].MacroRun()

### **Parameters**

### **Return value**

### **Usage**

**Word Pro: MailDocument method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAILDOCUMENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves and mails the current document using the MAPI application specified in your WIN.INI file.

**Syntax**

[objectreference].MailDocument()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: MakeTableFromText method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAKETABLEFROMTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Converts the selected text into a table. Equivalent to choosing Create - Table when you have text selected.

### **Syntax**

[objectreference].MakeTableFromText()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

Word Pro will try to automatically parse the selected text into a table, using tab marks as the cell delimiters and paragraph marks as the row delimiters.

## **Word Pro: MakeUniqueLinkName method**

{button ,AL(^H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAKEUNIQUELINKNAME\_METHOD\_EXSCRIPT',1)} [See example](#)

This method returns the string name. If the name is not unique, a number sign will be added to the link's name.

### **Syntax**

[objectreference].MakeUniqueLinkName(LinkName)

### **Parameters**

*LinkName*

A String expression representing the name of the link. Required parameter.

### **Return value**

String.

### **Usage**

If the name of the Dde link is not unique, a number will be added to the end of the name. This is useful when you are creating many links in a division or section, and want to name them by the division or section name (for example, DdeSection1, DdeSection2, and so on).

## **Word Pro: ManualFrame method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MANUALFRAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Turns on the manual frame drawing tool so the user can draw a frame manually. Equivalent to Choosing Create - Frame and clicking Size & Place Frame Manually.

### **Syntax**

[objectreference].ManualFrame([FrameStyle])

### **Parameters**

*FrameStyle*

A String expression which allows you to specify the frame style you want to use for the frame the user will create. Optional parameter.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: ManualLinkFrames method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_MANUALLINKFRAMES\_METHOD\_EXSCRIPT',1)} [See example](#)

Turns on the manual frame linking tool. Equivalent to selecting a frame and choosing Frame - Link Frame Contents.

**Syntax**

[objectreference].ManualLinkFrames()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

A non-linked frame must be selected when this method is called. The user must click inside another non-selected, non-linked frame to link the contents of the two frames.



## Word Pro: ManualTable method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MANUALTABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Turns on the ManualTable drawing tool which allows the user to draw a table manually. Equivalent to choosing Create - Table and clicking Size & Place Table Manually.

### Syntax

[objectreference].ManualTable([TableStyle,] [Columns,] [Rows])

### Parameters

#### *TableStyle*

A String expression which allows you to specify the table style you want to use for the table the user will create. Optional parameter.

#### *Columns*

A Numeric expression which allows you to specify the number of columns that will be in the table the user draws. Optional parameter. Data type is Integer.

#### *Rows*

A Numeric expression which allows you to specify the number of rows that will be in the table the user draws. Optional parameter. Data type is Integer.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

**Word Pro: MarkRevisionInsert method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MARKREVISIONINSERT\_METHOD\_EXSCRIPT',1)} [See example](#)

Marks the selected text as text which was a revision insertion.

**Syntax**

[objectreference].MarkRevisionInsert()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## Word Pro: Mark method

```
{button ,AL('H_BASETABLE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOOUT_CLASS;H_DROPPLAYOOUT_CLASS;H_ENDNOTELAYOOUT_CLASS;H_FOOTERLAYOOUT_CLASS;H_FOOTNOTELAYOOUT_CLASS;H_FOOTNOTETABLE_CLASS;H_FORMULA_CLASS;H_FRAMEGROUPLAYOOUT_CLASS;H_FRAMELAYOOUT_CLASS;H_GLOSSARY_CLASS;H_GRAPHIC_CLASS;H_GRAPHICOLEOBJECT_CLASS;H_GROUPLAYOOUT_CLASS;H_HEADERLAYOOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOOUT_CLASS;H_OLEOBJECT_CLASS;H_PAGELAYOOUT_CLASS;H_PARALLELCOLUMNS_CLASS;H_ROWGROUPLAYOOUT_CLASS;H_ROWLAYOOUT_CLASS;H_RUBYLAYOOUT_CLASS;H_SUPERTABLEGROUPLAYOOUT_CLASS;H_SUPERTABLELAYOOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAYOOUT_CLASS;H_TABLELAYOOUT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_MARK_METHOD_EXSCRIPT',1)} See example
```

Inserts a Marker object of the type you specify, or modifies an existing Marker object. This method is defined by several classes and some classes require additional parameters. For a list of parameters, see Syntax. For details on calling the method from a particular class, see Usage.

### Syntax

For WPAApplication, Graphic, OleObject, and Layout (and all of its derived classes):

```
[objectreference].Mark(MarkerType)
```

For Formula:

```
[objectreference].Mark(MarkerType, [MarkerName])
```

For ClickHere, Text, and TextMarker:

```
[objectreference].Mark(MarkerType, [MarkerName,] [RangePart])
```

For FootnoteTable, ParallelColumns, TableHeading, and Table:

```
[objectreference].Mark(MarkerType, [StartRow,] [StartCol,] [NumOfRows,] [NumOfCols])
```

### Parameters

#### *MarkerType*

Allows you to specify which type of marker you want to insert or modify. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

<b>Value</b>	<b>Effect</b>
\$LwpMarkerTypeBookmark (590)	Inserts a bookmark marker.
\$LwpMarkerTypeClickhere (593)	Inserts a ClickHere marker.
\$LwpMarkerTypeDde (591)	Inserts a DDE marker.
\$LwpMarkerTypeDefault (589)	Inserts a positional marker not of any other type and is only used to mark a spot or position.
\$LwpMarkerTypeField (592)	Inserts a field marker.
\$LwpMarkerTypeRuby (594)	Inserts a Ruby marker. Used only in the Asian-language versions of Word Pro.

#### *MarkerName*

Use this parameter to modify an existing Marker object. Do not include a value for this parameter if you are inserting a new Marker object. The value of this parameter must be a String expression representing the name of the Marker object you want to modify. This parameter is only available when you call this method from a Formula, ClickHere, Text, or TextMarker object.

#### *RangePart*

Use this parameter to modify an existing Marker object. Do not include a value for this parameter if you are inserting a new Marker object. To use this parameter, you must also provide a value for the MarkerName parameter. Use one of the constants below to tell Word Pro to modify the starting point of the marker, the ending point, or both. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in

parentheses). Default is \$LwpRangePartBoth. This parameter is only available when you call this method from a Formula, ClickHere, Text, or TextMarker object.

<b>Value</b>	<b>Effect</b>
\$LwpRangePartBoth (1663)	Modifies the starting point and ending point of the named marker to match the position of the insertion point. If something is selected, the starting and ending points match the starting and ending points of the selection.
\$LwpRangePartEnd (1665)	Modifies the ending point of the named marker to match the position of the insertion point. If something is selected, the ending point matches the ending point of the selection.
\$LwpRangePartStart (1664)	Modifies the starting point of the marker to match the position of the insertion point. If something is selected, the starting point matches the starting point of the selection.

#### *StartRow*

Modifies the starting row for the named marker object within a table object. Used only with FootnoteTable, ParallelColumns, TableHeading, and Table objects. Data type is Integer. Optional parameter.

#### *StartCol*

Data type is Integer. Optional parameter.

#### *NumOfRows*

Data type is Integer. Optional parameter.

#### *NumOfCols*

Data type is Integer. Optional parameter.

#### **Return value**

A String value representing the name of the marker object you insert or modify.

#### **Usage**

### **Word Pro: Maximize method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXIMIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

Maximizes the Word Pro application window.

### **Syntax**

[objectreference].Maximize()

### **Parameters**

None

### **Return value**

Integer. The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: MergeAddDataRecord method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGEADDDATARECORD\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a new record to the active Merge data file.

### Syntax

[objectreference].MergeAddDataRecord(RecordEntry)

### Parameters

#### *RecordEntry*

A String expression representing data for the record you are adding. This string must include the data for each field, separated by the appropriate field delimiters, and end with the appropriate record delimiter. For example, here is a String for a data file that uses "~" as the field delimiter and "|" as the record delimiter:

```
Jane~Doe~100 Main St.~Atlanta~GA|
```

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

**Word Pro: MergeContinue method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGECONTINUE\_METHOD\_EXSCRIPT',1)} [See example](#)

Not implemented in Word Pro 97.

## Word Pro: MergeSetDataFile method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGESETDATAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Assigns the specified Merge data file to the currently active document.

### Syntax

[objectreference].MergeSetDataFile(DataFilePath, DocWindowHwnd, DataFileType, DataFileTypeID[, DescriptionFile] [, Delimiter])

### Parameters

#### *DataFilePath*

A String expression representing the path and name of the data file you want to use with the current document. Optional parameter.

#### *DocWindowHwnd*

A Numeric expression which specifies the window ID for the Merge document to which you want to attach the data file. Data type is Long. Use a value of 0 (zero) to attach the data file to the currently active window.

#### *DataFileType*

A String expression which specifies the file type of the data file. Use the null string ("") to have Word Pro automatically detect the file type. If you provide a value for DataFileType, the DataFileTypeID is not necessary.

#### *DataFileTypeID*

A String expression which specifies the ID which Windows registers to represent a specific filter. Use the null string ("") to have Word Pro automatically assign the filter to a Windows registry ID.

#### *DescriptionFile*

A String expression that tells Word Pro the name of the description file for your Merge data file. In most data files, each line is a single record and the first record contains the names of the Merge fields. However, some data files are exported from databases and do not have the field names or field delimiters. These data files must be accompanied by a separate description file which contains the names of the Merge fields and the field delimiter or field sizes. If you provide no value for this parameter and the data file is not a Word Pro or Ami Pro document, Word Pro prompts you to find out if the first record in the file contains the field names.

Do not use this parameter if you are using an Ami Pro or Word Pro data file. Data files created by Word Pro and Ami Pro do not need a description file. If your data file is in ASCII text or spreadsheet format, and the first line contains the field names, you do not need a description file but you must use a null string ("") for this parameter.

#### *Delimiter*

A String expression that identifies the field delimiter for your data file. The value of this parameter should be the single character that acts as the field delimiter. Use a null string ("") to indicate that your data file is tab-delimited. If you provide no value for this field, Word Pro assumes that your data file is formatted in fixed-length ASCII and looks for the description file specified in the DescriptionFile parameter.

**Note** The record delimiter is presumed to be a line break (paragraph marker).

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Word Pro first checks to be sure the file named in DataFilePath is a valid Merge data file. If the file is not a data file, this method returns a value of 0 (False), indicating failure.



## Word Pro: MergeStart method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGESTART\_METHOD\_EXSCRIPT',1)} [See example](#)

Starts the Merge operation between the Merge document and the Merge data file.

### Syntax

[objectreference].MergeStart([MergeAction,] [MergeFilePath])

### Parameters

#### *MergeAction*

If no Merge document is active, you can specify which Merge document you want to use in starting the Merge operation. Data type is Variant which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is a null string which causes Word Pro to start the Merge operation, using the currently active Merge document and its assigned data file. If no Merge document is active and no value is included for this parameter, this method returns an error.

\$LwpMergeActionNewfile (1492) Opens a new Merge document, based on the SmartMaster specified in the MergeFilePath parameter.

\$LwpMergeActionOpenfile (1493) Opens the Merge document specified in MergeFilePath and uses that document's data file for the Merge.

#### *MergeFilePath*

A String expression representing the path and name of a Merge document or SmartMaster for a new Merge document. Optional parameter.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Use this method with no parameters to start a Merge operation. Then use the Merge method to issue specific instructions about how to carry out the Merge operation, or to switch Merge documents.

## Word Pro: Merge method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to issue instructions to Word Pro during the Merge operation. You can change Merge documents, end the Merge process, send the Merge job to the printer, and so on.

### Syntax

[objectreference].Merge(MergeAction,[MergeFilePath])

### Parameters

#### *MergeAction*

Indicates what action to take with the currently active Merge document. You can use one of the following strings or its numeric equivalent (in parentheses) for the value of this parameter:

\$LwpMergeActionNewfile (1492) Opens a new Merge document based on the SmartMaster specified in the MergeFilePath parameter. The currently active Merge data file is assigned to the new merge document.

\$LwpMergeActionOpenfile (1493) Opens the Merge document specified in MergeFilePath and uses the currently active data file for the Merge.

\$LwpMergeActionNextRecord (1494) Moves to the next record in the Merge data file. Each time you use this value, you must use this method again with the value, \$LwpMergeActionMergeOne, to get the next record to appear on screen.

\$LwpMergeActionMergeOne (1496) Updates the screen to display the latest record merged.

\$LwpMergeActionClose (1497) Ends the Merge process. You must set the MSSStep value to 0, 1, or 2.

\$LwpMergeActionContinue (1498) Sends the entire Merge job to the printer.

#### *MergeFilePath*

A String expression representing the path and name of a Merge document or SmartMaster for a new Merge document. Optional parameter.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Use this method to prepare a Merge document for a Merge operation, or to issue instructions during a Merge operation. This method is capable of opening a new Merge document (for example, a letter or envelope) based on a SmartMaster, opening an existing Merge document, switching from one data file to another, moving to the next record in the data file, ending the Merge operation, or sending the Merge job to the printer.

## Word Pro: Messages method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MESSAGES\_METHOD\_EXSCRIPT',1)} [See example](#)

Handles message boxes while a script is running.

### Syntax

[objectreference].Messages(MsgBoxAction)

### Parameters

*MsgBoxAction*

Specifies whether Word Pro should display a message box and wait for the user's response, or take the default response identified in the message box. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpDisplayMsgboxAndWait (2073)

\$LwpTakeDefaultMsgboxAnswer (2074)

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

**Word Pro: Minimize method**

{button ,AL(`H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_MINIMIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

Minimizes the Word Pro application window by reducing it to an icon.

**Syntax**

[objectreference].Minimize()

**Parameters**

None

**Return value**

Integer. The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: MirrorPage method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_MIRRORPAGE_METHOD_EXSCRIPT',1)} See example
```

Makes a mirror image of the referenced layout object and sets up a left/right (complex) page.

### Syntax

[objectreference].MirrorPage()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

You must first create the left/right page layout. You then invoke this method to make one page of that left/right layout a mirror image of the other. This method only works on left/right page layouts.

## **Word Pro: MorphSelectionToTable method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MORPHSELECTIONTOTABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].MorphSelectionToTable()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: MostRecentVersion method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOSTRECENTVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].MostRecentVersion()

### **Parameters**

### **Return value**

### **Usage**

'Example: IconSize property

'This example script has not yet been created.



'Example: IdentifierColor property

'This example script has not yet been created.

'Example: ID property

'This example script has not yet been created.

'Example: IgnoreSoftHyphens property

'This example script has not yet been created.

'Example: IgnoreTab property

'This example script has not yet been created.

```
'Example: ImportGraphic method
' This example inserts the 'TOON1' Word Pro Drawing graphic into the current
' document. The graphic is placed in a frame based upon the 'Default
' Graphic/OLE' frame style.
' RUNTIME DEPENDENCIES: You must have a document open and have installed the
' the Word Pro clipart into the \\DRAWSYM subdirectory for this script to ork.

Dim FileName as String
FileName = .Application.Path & "\\DRAWSYM\\TOON1.SDW"
.ApplicationWindow.Filter.SetLastUsedFilter $LwpFilterTypeGraphic,"Word Pro raw"
.ApplicationWindow.UserInterfacePrefs.LastGraphicType = "Word Pro Draw"

.ImportGraphic FileName, ".SDW", False, False, "Default Graphic/OLE"
```

'Example: ImportInserted event

'This example script has not yet been created.

'Example: ImportInsert event

'This example script has not yet been created.

'Example: ImportPicture method

'This example script has not yet been created.



'Example: IncludeInitialsInNotes property

'This example script has not yet been created.

'Example: IncludeList property

'This example script has not yet been created.

'Example: IndentFromLeft property

'This example script has not yet been created.

'Example: IndentStyleName property  
'This example script has not yet been created.

'Example: Indent property

'This example script has not yet been created.

'Example: IndexAlphabeticSeparator property  
'This example script has not yet been created.

'Example: IndexDivision property

'This example script has not yet been created.

'Example: IndexIndentType property

'This example script has not yet been created.



'Example: IndexParent property

'This example script has not yet been created.

'Example: IndexPrimaryStyleName property  
'This example script has not yet been created.

'Example: IndexRange property

'This example script has not yet been created.

'Example: IndexSecondaryStyleName property  
'This example script has not yet been created.

'Example: IndexSection property

'This example script has not yet been created.

'Example: IndexSeparatorStyleName property  
'This example script has not yet been created.

'Example: IndexSource property

'This example script has not yet been created.

'Example: InDocument property

'This example script has not yet been created.



'Example: InfoBoxSelectionText property  
'This example script has not yet been created.

'Example: InitalizeRoute property

'This example script has not yet been created.

```
'Example: InitFindAndReplace method
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and then
' displays a message box. When you click OK, the script finds the first 'cat'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"
.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find
```

'Example: Initialize property

'This example script has not yet been created.

'Example: InitialsForFilters property  
'This example script has not yet been created.

'Example: InsertBreak method

' This example inserts three lines of text into the current document.

' Each line is part of the same paragraph.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "First"

.Text.InsertBreak \$LwpBreakTypeWord

.Text.InsertText "line of text on paragraph 1."

.Text.InsertBreak \$LwpBreakTypeLine

.Text.InsertText "Second line of text on paragraph 1."

.Text.InsertBreak \$LwpBreakTypePage

.Text.InsertText "Third line of text on paragraph 1."

'Example: InsertBullet method

' This example inserts 4 paragraphs with various bullet styles into the  
' current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

**.InsertBullet "Wingdings", "1"**

.Text.SplitParagraph

**.InsertBullet "Wingdings", "°"**

.Text.SplitParagraph

**.InsertBullet "Wingdings", "·"**

.Text.SplitParagraph

**.InsertBullet "Wingdings", "ó"**

'Example: InsertClickHereLink method

'This example script has not yet been created.



```
'Example: InsertClickHere method
' This example inserts a ClickHere block in the current document and assigns
' the return value (the ClickHere ID) to a variable (NewClickHereId).
' The script then stores the new ClickHere in a variable (NewClickHere) and
' inserts some text in it.
' The script then prints the names and values for all ClickHeres in the
' current division to the Script Editor Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim NewClickHereId as String
Dim NewClickHere as ClickHere

NewClickHereId = .InsertClickHere()

Set NewClickHere = .Division.Foundry.ClickHeres(NewClickHereId)
NewClickHere.InsertText "Some text for the click here."

Forall Clicks in .Division.Foundry.ClickHeres
    Print Clicks.Name &" -- " & Clicks.GetText($LWPGetObjectParagraph, False)
End Forall
```

'Example: InsertColumnBreak method

' This example inserts a column break in the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertColumnBreak

'Example: InsertDate method

' This example inserts the date and time into the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

**.InsertDate "Now() %FLSystemShortDate"**

.Text.InsertHardSpace

**.InsertDate "Now() %FLSystemTime"**

'Example: InsertDocInfo method

'

' This example inserts the filename, date created, file size and author name  
' predefined document fields, as well as an end-user field named  
' 'CustomField1'.

' RUNTIME DEPENDENCIES: You must have a document open and have created  
' an end-user field named 'CustomField1' for this script to work.

.InsertDocInfo \$LwpDocVarFilename

.InsertDocInfo \$LwpDocVarDatecreated

.InsertDocInfo \$LwpDocVarDocsize

.InsertDocInfo \$LwpDocVarCreatedby

.InsertDocInfo \$LwpDocVarField, "CustomField1"

```
'Example: InsertDocument method
' This example uses an input box to get a file name from you and then inserts
' the contents of that document into the currently active document.
' The default file name is "README95.LWP".
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim FileName as String
FileName = InputBox$ ("Enter a filename to insert, e.g., README95.LWP:", "Example
Script", "")
```

```
.InsertDocument FileName, "", "", False, True
```

```
'Example: InsertField method
' This example inserts several power fields into the current document.
' Each field is on a new line and is preceded by its description.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "Create date: "
.InsertField "CreateDate %DB"
.Text.SplitParagraph

.Text.InsertText "Edit date: "
.InsertField "EditDate %DB"
.Text.SplitParagraph

.Text.InsertText "Editing time: "
.InsertField "TotalEditingTime"
.Text.SplitParagraph

.Text.InsertText "Number of words: "
.InsertField "NumWords"
```

'Example: InsertFont property

'This example script has not yet been created.

'Example: InsertFootnote method

' This example inserts a footnote into the current document. The footnote

' is placed at the bottom of the page and is numbered 1.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Division.FootnoteOptions.FootnoteNumbering.StartingNumber = 1

.Division.FootnoteOptions.FootnoteNumbering.ResetWhen = LwpResetOptionEachPage

**.InsertFootnote \$LwpFnTypeAtBottomOfPage**



```
'Example: InsertFrame method
' This example inserts a frame based on the default frame style into the
' current document. If the frame is succesfully created, the NumCols
' properties is modified. After the message box is closed the the frame's
' NumCol property is reverted.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim FrameWidth as Integer
Dim FrameHeight as Integer
Dim FrameX as Integer
Dim FrameY as Integer
FrameWidth = 2265
FrameHeight = 2595
FrameX = 1410
FrameY = 2160

.InsertFrame FrameWidth, FrameHeight, FrameX, FrameY
If Not (.Frame Is Nothing) Then
    With .Frame.Layout
        .NumCols = 3
    End With
    Messagebox "Click OK to revert frame.",MB_OK,"Example Script"
    .FrameRevert
End If
```

'Example: InsertHardSpace method

' This example inserts 2 words seperated by a hard space into the current  
' document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "Some"

.Text.InsertHardSpace

.Text.InsertText "text."

'Example: InsertIndex method

' This example Inserts an index at the end of the current document in a

' separate division based on the default SmartMaster.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertIndex

'Example: InsertionMode property

'This example script has not yet been created.

'Example: InsertLink method

'This example script has not yet been created.

'Example: InsertMarker method

'This example script has not yet been created.

'Example: InsertNote method

- ' This example inserts a note, adds some text to it and then deselects the note.
- ' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

**.InsertNote**

```
.Text.InsertText "Some text for the note."  
.Type("[ESC]")
```

'Example: InsertNumber method

' This example insert the roman numeral 5 into the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertNumber \$LwpNumberingStyleUppercaseroman , 0, 0, 5



```
'Example: InsertOleDivision method
' This example insert an OLE division based on the bitmap graphic file named
' TEST.BMP
' RUNTIME DEPENDENCIES: You must have a document open and a file named
' TEST.BMP located in the C:\ directory for this script to work.
```

```
Dim FilePath As String
Dim ClassID As String
```

```
ClassID = "{0003000A-0000-0000-C000-000000000046}"
FilePath = "C:\TEST.BMP"
.InsertOleDivision $LwpDivLocInsertAfterCurrentdiv,
$LwpOleActionCreateembedded,ClassID,FilePath
```

'Example: InsertOne method

' This example inserts a right align tab with leader dots one inch from the  
' right margin.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.TabRack.InsertOne 1444, \$LwpTabTypeRight , \$LwpTabLeaderDot ,  
\$LwpTabRelativeRight , 32

'Example: InsertPageBreak method

' This example inserts a page break in the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertPageBreak

```
'Example: InsertPageLayout method
' This example inserts a new page layout based on the current page layout.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim StyleName as String
StyleName = .Page.Layout.Style.Name

.InsertPageLayout StyleName , 0, 0, 0
```

'Example: InsertPageNumber method

' This example inserts a page number on the current document in the

' following format: "Page 1 - Division Name".

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertPageNumber \$LwpNumberingStyleBasic, "Page ", " - ", 0, 1,  
\$LwpPageNumberFlagsIncludedivname + \$LwpPageNumberFlagsResetondivision

'Example: InsertPath method

' This example changes Word Pro's default document path.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Application.ApplicationWindow.UserInterfacePrefs.InsertPath \$LwpSetDocumentsPath,  
"C:\dev\lotus"

'Example: InsertRowOrColumn method

' This example creates a parallel column table with 3 columns. A new row is  
' inserted.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.BaseTable.InsertRowOrColumn \$LwpTableInsTypeRow, True, 1

'Example: InsertRuby method

' This example inserts a ruby into the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertRuby



'Example: InsertSection method

' This example inserts a new section in the active division of the current  
' document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.  
.InsertSection "Default Page", True, True, \$LwpStartTypeNextpage, False, True

'Example: InsertTab method

' This example inserts 5 consecutive tabs.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
rc = .Text.InsertTab(5)
```

'Example: InsertText method

' This example inserts some text into the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "Some text for the document."

'Example: InsertTOC method

' This example inserts a Table of Contents (TOC) at beginning of the active  
' document in a separate division based on the default SmartMaster.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.InsertTOC \$LwpGenerateAcrossEntireDoc, \$LwpTOCPlacementBeginofdoc, True

'Example: Interactive property

'This example script has not yet been created.

'Example: InternalCopy method

' This example copies the selected text, creates a new line and then pastes  
' the copied text on the new line.

' RUNTIME DEPENDENCIES: You must have a document open and text selected for  
' this script to work.

.Text.InternalCopy(True)

.Text.MoveToEnd \$LwpLocationTypeLine

.Text.SplitParagraph

.Text.InternalPaste \$LwpFoundryTypeTemporary

'Example: InternalCut method

' This example cuts the selected text, creates a new line and then pastes  
' the copied text on the new line.  
' RUNTIME DEPENDENCIES: You must have a document open and text selected for  
' this script to work.

```
.Text.InternalCut(True)  
.Text.MoveToEnd $LwpLocationTypeLine  
.Text.SplitParagraph  
.Text.InternalPaste $LwpFoundryTypeTemporary
```

'Example: InternalPaste method

' This example copies the selected text, creates a new line and then pastes  
' the copied text on the new line.

' RUNTIME DEPENDENCIES: You must have a document open and text selected for  
' this script to work.

.Text.InternalCopy(True)

.Text.MoveToEnd \$LwpLocationTypeLine

.Text.SplitParagraph

.Text.InternalPaste \$LwpFoundryTypeTemporary



'Example: InUseCount property

'This example script has not yet been created.

```
'Example: InvalidateButton method
' This example creates a new button to the status bar and then adds text to
' the button. The STATUSBARBUTTONOVERRIDE TEXT is then bound to the
' SetTheButtonText subroutine to set the button text during needs repainting.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ButtonName As String
Dim NewButton As StatusBarButton
With .ApplicationWindow.StatusBar
    ButtonName = .CreateNewbutton (0,0,100,&H1) 'create the new button
    Set NewButton = .StatusBarButtons(ButtonName)
    With .StatusBarButtons(ButtonName)
        .SetOverRideText("New Button...")
        Call .SetButtonText("Button",True)
        .InvalidateButton
        On Event STATUSBARBUTTONOVERRIDE TEXT From NewButton Call
            SetTheButtonText
    End With
    .InvalidateWholeBar ' Force the bar to repaint
End With
End Sub

Sub SetTheButtonText (Source As StatusBarButton, ButtonName As String)
    'Add the the button text each time the status bar needs repainting.
    Source.SetOverRideText("New Button...")
    End 2
End Sub
```

```
'Example: InvalidateWholeBar method
' This example creates a new button to the status bar and then adds text to
' the button. The STATUSBARBUTTONOVERRIDE TEXT is then bound to the
' SetTheButtonText subroutine to set the button text during needs repainting.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim ButtonName As String
Dim NewButton As StatusBarButton
With .ApplicationWindow.StatusBar
    ButtonName = .CreateNewbutton (0,0,100,&H1) 'create the new button
    Set NewButton = .StatusBarButtons(ButtonName)
    With .StatusBarButtons(ButtonName)
        .SetOverRideText("New Button...")
        Call .SetButtonText("Button",True)
        .InvalidateButton
        On Event STATUSBARBUTTONOVERRIDE TEXT From NewButton Call
            SetTheButtonText
    End With
    .InvalidateWholeBar ' Force the bar to repaint
End With
End Sub

Sub SetTheButtonText (Source As StatusBarButton, ButtonName As String)
    'Add the the button text each time the status bar needs repainting.
    Source.SetOverRideText("New Button...")
    End 2
End Sub
```

'Example: Invalidate method

' This example creates a bookmark based on the current marker name and then  
' deletes the marker and repaints the active document to clear its bookmark  
' identifier.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
Dim MarkerName As String
```

```
MarkerName = .Mark($LwpMarkerTypeBookmark)
```

```
.Division.BookmarkManager.AddBookmark "ExampleBookmark", MarkerName
```

```
MessageBox "Click OK to delete the bookmark.",MB_OK,"Example Script"
```

```
rc =.Division.Foundry.Markers.Item(MarkerName).DeleteMarker()
```

```
rc = .ActiveDocWindow.Invalidate()
```

'Example: IsActionOnButtonDown property

'This example script has not yet been created.

'Example: IsActive property

'This example script has not yet been created.

'Example: IsAdopted property

'This example script has not yet been created.

'Example: IsAmiProTableImport property  
'This example script has not yet been created.



'Example: IsAsciiCRLF property

'This example script has not yet been created.

'Example: IsAsciiKeepStyle property  
'This example script has not yet been created.

'Example: IsAutoGrow property

'This example script has not yet been created.

'Example: IsBadReference property

'This example script has not yet been created.

'Example: IsBorder property

'This example script has not yet been created.

'Example: IsBothSidesEqual property  
'This example script has not yet been created.

'Example: IsBottomAligned property  
'This example script has not yet been created.

'Example: IsBreakable property

'This example script has not yet been created.



'Example: IsBubbleHelp property

'This example script has not yet been created.

'Example: IsCellMenuEnabled property

'This example script has not yet been created.

'Example: IsCentered property

'This example script has not yet been created.

'Example: IsChangedSinceTimeSave property

'This example script has not yet been created.

'Example: IsChanged property

'This example script has not yet been created.

'Example: IsChildSpannable property  
'This example script has not yet been created.

'Example: IsCollapsed property

'This example script has not yet been created.

'Example: IsCollapsible property

'This example script has not yet been created.



'Example: IsColumnBreakable property

'This example script has not yet been created.

'Example: IsColumnBreakAfter property

'This example script has not yet been created.

'Example: IsColumnBreakBefore property

'This example script has not yet been created.

'Example: IsComplex property

'This example script has not yet been created.

'Example: IsConnected property

'This example script has not yet been created.

'Example: IsContentReplaceable property

'This example script has not yet been created.

**Word Pro: IconSize property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ICONSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that indicates the size in which the icons are displayed. Refers to all icons on every bar. You can set the size, either Regular or Large, in the "Icon size" field in the SmartIcons Setup dialog box.

**Data Type**

Variant (Enumerated)

IconSize

**Syntax**

iconsizevalue = [objectreference].IconSize

[objectreference].IconSize = iconsizevalue

**Legal values**

\$LwplconSizeSupervga (395) Sets or changes the size of displayed icons to Large.

\$LwplconSizeVga (394) Sets or changes the size of displayed icons to Regular.

**Usage**

You can write a script to query or change the size of displayed icons.

## **Word Pro: ID property**

{button ,AL('H\_ICONBAR\_CLASS;H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

[IconBar]

The ID of the icon bar.

[MenuItem]

The menu ID for the menu item. ID is a read-only property which Word Pro sets each time you create your menu item.

## **Data Type**

Long

## **Syntax**

idvalue = [objectreference].ID

## **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## **Usage**



## **Word Pro: IgnoreSoftHyphens property**

{button ,AL('H\_HYPHENATIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IGNORESOFTHYPHENS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ignoresofthyphensvalue = [objectreference].IgnoreSoftHyphens

[objectreference].IgnoreSoftHyphens = ignoresofthyphensvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IgnoreTab property**

{button ,AL('H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IGNORETAB\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

ignoretabvalue = [objectreference].IgnoreTab

[objectreference].IgnoreTab = ignoretabvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IncludeInitialsInNotes property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INCLUDEINITIALSINNOTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

includeinitialsinnotesvalue = [objectreference].IncludeInitialsInNotes

[objectreference].IncludeInitialsInNotes = includeinitialsinnotesvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IncludeList property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INCLUDELIST\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to instruct Find & Replace to search for text in different areas in the document.

### **Data Type**

Integer (Enumerated Bitmask)

Variant

### **Syntax**

includelistvalue = [objectreference].IncludeList

[objectreference].IncludeList = includelistvalue

### **Legal values**

LwpIncludeListAllText (&H1) Sets the option to search all text in the document.

LwpIncludeListFootnotes (&H20) Sets the option to search all text in footnotes.

LwpIncludeListFrames (&H10) Sets the option to search all text in frames.

LwpIncludeListHeadersFooters (&H4) Sets the option to search all text in headers and footers.

LwpIncludeListMainDocText (&H2) Sets the option to search text only in the main part of the document.

LwpIncludeListTables (&H8) Sets the option to search all text in tables.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text, clicking Options, and selecting an option in the "Include" box in the "Find & replace scope" section.

### **Word Pro: IndentFromLeft property**

{button ,AL(^H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_INDENTFROMLEFT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

indentfromleftvalue = [objectreference].IndentFromLeft

[objectreference].IndentFromLeft = indentfromleftvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: IndentStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDENTSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

indentstylevalue = [objectreference].IndentStyleName

[objectreference].IndentStyleName = indentstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: IndexAlphabeticSeparator property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXALPHABETICSEPARATOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

indexalphabeticseparatorvalue = [objectreference].IndexAlphabeticSeparator

[objectreference].IndexAlphabeticSeparator = indexalphabeticseparatorvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IndexDivision property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXDIVISION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

indexdivisionvalue = [objectreference].IndexDivision

[objectreference].IndexDivision = indexdivisionvalue

**Legal values****Usage**



## **Word Pro: IndexIndentType property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXINDENTTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

indexindentypevalue = [objectreference].IndexIndentType

[objectreference].IndexIndentType = indexindentypevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IndexParent property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXPARENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

indexparentvalue = [objectreference].IndexParent

[objectreference].IndexParent = indexparentvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: IndexPrimaryStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXPRIMARYSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

indexprimarystylevalue = [objectreference].IndexPrimaryStyleName

[objectreference].IndexPrimaryStyleName = indexprimarystylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: IndexRange property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXRANGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

indexrangevalue = [objectreference].IndexRange

[objectreference].IndexRange = indexrangevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: IndexSecondaryStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXSECONDARYSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

indexsecondarystylevalue = [objectreference].IndexSecondaryStyleName

[objectreference].IndexSecondaryStyleName = indexsecondarystylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: IndexSection property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXSECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

String

**Syntax**

indexsectionvalue = [objectreference].IndexSection

[objectreference].IndexSection = indexsectionvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: IndexSeparatorStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXSEPARATORSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

indexseparatorstylevalue = [objectreference].IndexSeparatorStyleName

[objectreference].IndexSeparatorStyleName = indexseparatorstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: IndexSource property**

{button ,AL('H\_INDEXSECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDEXSOURCE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

GenerateFrom

### **Syntax**

indexsourcevalue = [objectreference].IndexSource

[objectreference].IndexSource = indexsourcevalue

### **Legal values**

\$LwpGenerateFromCurrentdivision (362)

\$LwpGenerateFromCurrentleveldivision (361)

\$LwpGenerateFromCurrentsection (363)

\$LwpGenerateFromEntiredocument (360)

\$LwpGenerateFromMarker (364)

### **Usage**



**Word Pro: InDocument property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INDOCUMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

indocumentvalue = [objectreference].InDocument

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: InfoBoxSelectionText property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INFOBOXSELECTIONTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

infoboxselectiontextvalue = [objectreference].InfoBoxSelectionText

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: InitializeRoute property**

{button ,AL('H\_MAILROUTING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INITIALZEROUTE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) An internal flag that indicates the default state (broadcast vs. route) when you use the Initialize property to set defaults.

**Data Type**

[Integer](#)

**Syntax**

initializeroutevalue = [objectreference].InitializeRoute

[objectreference].InitializeRoute = initializeroutevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Used in conjunction with the Initialize property. First, set Initialize to True. Then set the desired value for this property (broadcast or route). If the Initialize property is 0, this property is ignored.

**Word Pro: Initialize property**

{button ,AL('H\_MAILROUTING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INITIALIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) An internal flag that indicates if you can set up defaults in the Mail Routing dialog box.

**Data Type**

[Integer](#)

**Syntax**

initializevalue = [objectreference].Initialize

[objectreference].Initialize = initializevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

A value of 1 indicates user-defined settings.

## **Word Pro: InitialsForFilters property**

{button ,AL('H\_NOTELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INITIALSFILTERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)The initials of the user that created a specific comment note.

### **Data Type**

String

### **Syntax**

initialsforfiltersvalue = [objectreference].InitialsForFilters

[objectreference].InitialsForFilters = initialsforfiltersvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

The default value for this property is contained in the "Initials" setting, which is located in the Personal panel of the Word Pro Preferences dialog box. Modifying the NameForFilters property of a specific comment note does not change the name setting under Word Pro Preferences.

### **Word Pro: InsertionMode property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INSERTIONMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

insertionmodevalue = [objectreference].InsertionMode

[objectreference].InsertionMode = insertionmodevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: Interactive property**

{button ,AL('H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INTERACTIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not Word Pro is in interactive mode. If it is in interactive mode (True), the user can interact with the application using the keyboard, mouse, or other input device.

### **Data Type**

[Integer](#)

### **Syntax**

interactivevalue = [objectreference].Interactive

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

If you suspect that Word Pro may be engaged in an OLE automation session when your script runs, you can use this property as part of an IF...THEN statement to make sure that Word Pro is available before executing your script.

**Word Pro: Interval property**

{button ,AL('H\_LWPTIMER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INTERVAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

intervalvalue = [objectreference].Interval

[objectreference].Interval = intervalvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**



## **Word Pro: InUseCount property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_INUSECOUNT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A count of the number of places in a division where a character style object is used.

### **Data Type**

Long

### **Syntax**

inusecountvalue = [objectreference].InUseCount

### **Legal values**

### **Usage**

**Word Pro: IsActionOnButtonDown property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISACTIONONBUTTONDOWN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) A flag that indicates if the action represented by an icon occurs when the icon is pressed down. Typically, the icon action occurs on the WM\_LBUTTONDOWN. Setting this flag causes the action to occur on the WM\_LBUTTONDOWNDOWN message.

**Data Type**

[Integer](#)

**Syntax**

[objectreference].IsActionOnButtonDown = isactiononbuttondownvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You must first select an icon using either the SelectStandardIcon or SelectCustomIcon method for this property to function. Default is False. If the flag is set to True, the action occurs when the icon is depressed. If set to False, the action occurs when the icon is let up.

**Word Pro: IsActive property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISACTIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

isactivevalue = [objectreference].IsActive

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**UsageUsage**

## Word Pro: IsAdopted property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISADOPTED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Not implemented.

## Data Type

[Integer](#)

## Syntax

isadoptedvalue = [objectreference].IsAdopted

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

### **Word Pro: IsAmiProTableImport property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISAMIPROTABLEIMPORT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether the Merge data file is derived from an Ami Pro table.

### **Data Type**

[Integer](#)

### **Syntax**

isamiprotatableimportvalue = [objectreference].IsAmiProTableImport

[objectreference].IsAmiProTableImport = isamiprotatableimportvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: IsAsciiCRLF property**

{button ,AL('H\_FILTER\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISASCIICRLF\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isasciicrlfvalue = [objectreference].IsAsciiCRLF

[objectreference].IsAsciiCRLF = isasciicrlfvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsAsciiKeepStyle property**

{button ,AL('H\_FILTER\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISASCIKEEPSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isasciikeepstylevalue = [objectreference].IsAsciiKeepStyle

[objectreference].IsAsciiKeepStyle = isasciikeepstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: IsAutoGrow property

{button ,AL(`H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ISAUTOGROW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not the height of a row in a table object increases when text needs to wrap to another line, or when the point or font size of the text changes.

### Data Type

Integer

### Syntax

isautogrowvalue = [objectreference].IsAutoGrow

[objectreference].IsAutoGrow = isautogrowvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Functionally equivalent to simultaneously modifying the "Automatic row height" setting for all rows within a table.



**Word Pro: IsBadReference property**

{button ,AL('H\_CELLLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISBADREFERENCE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not a formula in a table cell refers to a cell that does not exist in the table.

**Data Type**

Integer

**Syntax**

isbadreferencevalue = [objectreference].IsBadReference

[objectreference].IsBadReference = isbadreferencevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

If this property returns True, the formula in the table cell refers to a cell that does not exist in the table. If this property returns False, the formula in the table cell does not refer to a cell that does not exist in the table.

**Word Pro: IsBorder property**

{button ,AL('H\_CHARACTERBORDER\_CLASS;H\_PARAGRAPHBORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

isbordervalue = [objectreference].IsBorder

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsBothSidesEqual property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISBOTHSIDeseQUAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isbothsidesequalvalue = [objectreference].IsBothSidesEqual

[objectreference].IsBothSidesEqual = isbothsidesequalvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: IsBottomAligned property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISBOTTOMALIGNED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not the contents of a layout are aligned to the bottom of the content area in a layout.

### Data Type

Integer

### Syntax

isbottomalignedvalue = [objectreference].IsBottomAligned

[objectreference].IsBottomAligned = isbottomalignedvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: IsBreakable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISBREAKABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not a break can be inserted into the content of a layout object.

## Data Type

[Integer](#)

## Syntax

isbreakablevalue = [objectreference].IsBreakable

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: IsBubbleHelp property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISBUBBLEHELP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) A flag that indicates whether bubble help for icons will be shown when the cursor hovers over an icon. If set to True, bubble help displays. This flag can be changed through the user interface in the "Show icon descriptions (bubble help)" field in the SmartIcons Setup dialog box.

**Data Type**

[Integer](#)

**Syntax**

[objectreference].IsBubbleHelp= isbubblehelpvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You can write a script that lets bubble help display or not display over all icons. The user interface is found in the "Show icon descriptions (bubble help)" field in the SmartIcons Setup dialog box.

**Word Pro: IsCellMenuEnabled property**

{button ,AL('H\_CONTEXTMENUOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCELLMENUENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Controls whether or not a cell menu displays in a cell.

**Data Type**

[Integer](#)

**Syntax**

iscellmenuenabledvalue = [objectreference].IsCellMenuEnabled

[objectreference].IsCellMenuEnabled = iscellmenuenabledvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Setting this property to True allows the cell menu to display when the focus is in a text object. Setting this property to False prevents the display of the cell menu when the focus is in a text object.

**Word Pro: IsCentered property**

{button ,AL('H\_DOCWINDOW\_CLASS;H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCENTERED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines whether or not each page of a printed document is centered.

**Data Type**

[Integer](#)

**Syntax**

iscenteredvalue = [objectreference].IsCentered

[objectreference].IsCentered = iscenteredvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## **Word Pro: IsChangedSinceTimeSave property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCHANGEDSINCETIMESAVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ischangedsincetimesavevalue = [objectreference].IsChangedSinceTimeSave

[objectreference].IsChangedSinceTimeSave = ischangedsincetimesavevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsChanged property**

```
{button ,AL('H_CLICKHERE_CLASS;H_DIVISION_CLASS;H_MARKER_CLASS;H_POWERFIELD_CLASS;H_RUB  
YMARKER_CLASS;H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL('H_ISCHANGED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ischangedvalue = [objectreference].IsChanged

[objectreference].IsChanged = ischangedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsChildSpannable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISCHILDSPANNABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows a layout object's children to span across pages.

## Data Type

[Integer](#)

## Syntax

ischildspannablevalue = [objectreference].IsChildSpannable

[objectreference].IsChildSpannable = ischildspannablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: IsCollapsed property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISCOLLAPSED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) This property is used with the NoteLayout property to determine if a Note is collapsed.

## Data Type

Integer

## Syntax

iscollapsedvalue = [objectreference].IsCollapsed

[objectreference].IsCollapsed = iscollapsedvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

When a NoteLayout object is collapsed, it appears in Word Pro as a small colored box.

## Word Pro: IsCollapsible property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISCOLLAPSIBLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines whether or not a layout object shrinks to an icon.

## Data Type

Integer

## Syntax

iscollapsiblevalue = [objectreference].IsCollapsible

[objectreference].IsCollapsible = iscollapsiblevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

May cause unpredictable results if used with layout objects which are not comment note objects.

## Word Pro: IsColumnBreakable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERABLEGROUPLAYOUT_CLASS;H_SUPERABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISCOLUMNBREAKABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not a column break can be inserted into the content of a layout object.

## Data Type

[Integer](#)

## Syntax

iscolumnbreakablevalue = [objectreference].IsColumnBreakable

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: IsColumnBreakAfter property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCOLUMNBREAKAFTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

iscolumnbreakaftervalue = [objectreference].IsColumnBreakAfter

[objectreference].IsColumnBreakAfter = iscolumnbreakaftervalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsColumnBreakBefore property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCOLUMNBREAKBEFORE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

iscolumnbreakbeforevalue = [objectreference].IsColumnBreakBefore

[objectreference].IsColumnBreakBefore = iscolumnbreakbeforevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## Word Pro: IsComplex property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISCOMPLEX_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a page layout pertains to all pages, or just odd/even pages.

## Data Type

[Integer](#)

## Syntax

iscomplexvalue = [objectreference].IsComplex

[objectreference].IsComplex = iscomplexvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Equivalent to the "Settings for" option located on the Size & Margins panel of the Page layout InfoBox for page layout objects.

## Word Pro: IsConnected property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISCONNECTED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) This property indicates whether or not a frame is a member of a group.

## Data Type

[Integer](#)

## Syntax

isconnectedvalue = [objectreference].IsConnected

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

A frame layout object is a member of a group if it is grouped together with sibling frame layout objects. Equivalent to setting the "Group" option on the Frame menu.

## Word Pro: IsContentReplaceable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOOUT_CLASS;H_DROPCAPLAYOOUT_CLASS;H_ENDNOTELAYOOUT_CLASS;H_FOOTERLAYOOUT_CLASS;H_FOOTNOTELAYOOUT_CLASS;H_FRAMEGROUPLAYOOUT_CLASS;H_FRAMELAYOOUT_CLASS;H_GROUPLAYOOUT_CLASS;H_HEADERLAYOOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOOUT_CLASS;H_PAGELAYOOUT_CLASS;H_ROWGROUPLAYOOUT_CLASS;H_ROWLAYOOUT_CLASS;H_RUBYLAYOOUT_CLASS;H_SUPERTABLEGROUPLAYOOUT_CLASS;H_SUPERTABLELAYOOUT_CLASS;H_TABLEHEADINGLAYOOUT_CLASS;H_TABLELAYOOUT_CLASS;H_TOCSUPERTABLELAYOOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISCONTENTREPLACEABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) For internal use only.

## Data Type

Integer

## Syntax

iscontentreplaceablevalue = [objectreference].IsContentReplaceable

[objectreference].IsContentReplaceable = iscontentreplaceablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## **Word Pro: IsContents property**

{button ,AL('H\_FOUNDRY\_CLASS;H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCONTENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) For internal use only. Do not use this language element in your scripts.

### **Data Type**

[Integer](#)

### **Syntax**

### **Legal values**

### **Usage**

**Word Pro: IsContinuedFrom property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCONTINUEDFROM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

iscontinuedfromvalue = [objectreference].IsContinuedFrom

[objectreference].IsContinuedFrom = iscontinuedfromvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

### **Word Pro: IsContinuedOn property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCONTINUEDON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

iscontinuedonvalue = [objectreference].IsContinuedOn

[objectreference].IsContinuedOn = iscontinuedonvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsCumulative property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISCUMULATIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

iscumulativevalue = [objectreference].IsCumulative

[objectreference].IsCumulative = iscumulativevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsDebug property**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDEBUG\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isdebugvalue = [objectreference].IsDebug

[objectreference].IsDebug = isdebugvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: IsDisableWarningMessages property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDISABLEWARNINGMESSAGES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if the Word Pro version warning messages are enabled.

### **Data Type**

[Integer \(bool\)](#)

### **Syntax**

isdisablewarningmessagesvalue = [objectreference].IsDisableWarningMessages

[objectreference].IsDisableWarningMessages = isdisablewarningmessagesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to the "Disable" field on the General panel of the Word Pro Preferences dialog box. If the legal value for this property is False, Word Pro turns off all version warning messages. If the legal value for this property is True, Word Pro warns the user that there is a newer version of the document.

## **Word Pro: IsDisplayMisspelled property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDISPLAYMISPELLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isdisplaymisspelledvalue = [objectreference].IsDisplayMisspelled

[objectreference].IsDisplayMisspelled = isdisplaymisspelledvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsDivisionExternal property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDIVISIONEXTERNAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isdivisionexternalvalue = [objectreference].IsDivisionExternal

[objectreference].IsDivisionExternal = isdivisionexternalvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsDocLoading property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDOCLOADING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isdocloadingvalue = [objectreference].IsDocLoading

[objectreference].IsDocLoading = isdocloadingvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: IsDocumentInRoute property**

{button ,AL('H\_MAILROUTING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDOCUMENTINROUTE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An internal flag that indicates a value as to whether a document is in a mail route.

### **Data Type**

[Integer](#)

### **Syntax**

isdocumentinroutevalue = [objectreference].IsDocumentInRoute

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsDoubleWordError property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDOUBLEWORDERROR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isdoubleworderrorvalue = [objectreference].IsDoubleWordError

[objectreference].IsDoubleWordError = isdoubleworderrorvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

## **Word Pro: IsDraw property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISDRAW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

isdrawvalue = [objectreference].IsDraw

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsEmptyDoc property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_IEMPTYDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

isemptydocvalue = [objectreference].IsEmptyDoc

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## Word Pro: IsEmpty property

```
{button ,AL('H_BAGCOLLECTION_CLASS;H_BASECOLLECTION_CLASS;H_BASETABLE_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_CELLCOLLECTION_CLASS;H_CELLLAYOUTCOLLECTION_CLASS;H_CHARACTERSTYLECOLLECTION_CLASS;H_CLICKHERE_CLASS;H_CLICKHERECOLLECTION_CLASS;H_CONNECTEDLAYOUTCOLLECTION_CLASS;H_CONTENT_CLASS;H_CONTENTCOLLECTION_CLASS;H_DDELINKCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCINFOFIELDCOLLECTION_CLASS;H_DOCUMENTS_CLASS;H_DOCWINDOWCOLLECTION_CLASS;H_DROPCAPLAYOUTCOLLECTION_CLASS;H_EDITORCOLLECTION_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECTION_CLASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTION_CLASS;H_FOOTNOTETABLE_CLASSES;H_FORMULA_CLASS;H_FRAMELAYOUTCOLLECTION_CLASS;H_GLOSSARY_CLASS;H_GLOSSARYCOLLECTION_CLASS;H_GRAPHIC_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHICOLEOBJECT_CLASS;H_GRAPHICOLEOBJECTCOLLECTION_CLASS;H_GROUPLAYOUTCOLLECTION_CLASS;H_HEADERLAYOUTCOLLECTION_CLASS;H_ICONBARCOLLECTION_CLASS;H_LAYOUTCOLLECTION_CLASS;H_MARKERCOLLECTION_CLASS;H_MENUITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLASS;H_OLEOBJECT_CLASS;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLINESEQCOLLECTION_CLASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELAYOUTCOLLECTION_CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARALLELCOLSCOLLECTION_CLASS;H_PARALLELCOLUMNS_CLASS;H_POWERFIELDCOLLECTION_CLASS;H_ROWLAYOUTCOLLECTION_CLASS;H_RUBYLAYOUTCOLLECTION_CLASS;H_SECTIONCOLLECTION_CLASS;H_SILVERBULLETCOLLECTION_CLASS;H_SMARTCORRECTCOLLECTION_CLASS;H_SMARTFILLCOLLECTION_CLASS;H_STATUSBARBUTTONCOLLECTION_CLASS;H_STRINGCOLLECTION_CLASS;H_SUPERTABLE_CLASS;H_SUPERTABLECOLLECTION_CLASS;H_SUPERTABLELAYOUTCOLLECTION_CLASSES;H_TABLE_CLASS;H_TABLECOLLECTION_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGCOLLECTION_CLASS;H_TABLEHEADINGLAYOUTCOLLECTION_CLASS;H_TABLELAYOUTCOLLECTION_CLASS;H_TABLEMARKERCOLLECTION_CLASS;H_TABLEONLYCOLLECTION_CLASS;H_TEXT_CLASS;H_TEXTCOLLECTION_CLASS;H_TEXTMARKERCOLLECTION_CLASS;H_TEXTSTYLECOLLECTION_CLASS;H_VERSIONCOLLECTION_CLASS;H_WPDATASETCOLLECTION_CLASS',0)} See list of classes
```

```
{button ,AL('H_IEMPTY_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not the text stream is empty.

[ClickHere block]

Indicates whether a ClickHere block is in its collapsed state.

[Content][Table]

Indicates whether or not a specific content object is empty.

## Data Type

Integer

## Syntax

isemptyvalue = [objectreference].IsEmpty

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This property returns True only for Text contents that contains no characters. For example, when a default frame is created, it contains an empty Text content. When the user types some characters into the frame, the Text content is no longer empty. If the user deletes all those characters, it is empty again.

### **Word Pro: IsEnabled property**

{button ,AL('H\_ICONBAR\_CLASS;H\_KINSOKU\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that tells you whether or not an icon bar object is enabled and will display, thus allowing you to click on an icon in the set. If an icon bar is disabled, it is also usually hidden.

### **Data Type**

[Integer](#)

### **Syntax**

isenabledvalue = [objectreference].IsEnabled

[objectreference].IsEnabled = isenabledvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This property allows you to display an icon bar object and click on the icons.

**Word Pro: IsEquation property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISEQUATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

isequationvalue = [objectreference].IsEquation

**Legal values**

True False

**Usage**

## Word Pro: IsErrorChecking property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISERRORCHECKING_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set error checking to either allow or prevent an invalid operation.

### Data Type

Integer

### Syntax

iserrorcheckingvalue = [objectreference].IsErrorChecking

[objectreference].IsErrorChecking = iserrorcheckingvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

This property allows you to determine whether or not the inherent error checking of a layout object is enabled. Error checking performed on a frame layout object, for example, would prevent an illegal setting for the size or the placement of the frame.

This property can be used by document filters in order to fully control layout objects and their placement within a Word Pro document.

Setting this property to True (-1) prevents an invalid operation. This is the default value. Setting this property to False (0) allows an invalid operation.

## Word Pro: IsExpandDown property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_IEXPANDDOWN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to expand a layout object to the bottom to accommodate its contents.

## Data Type

[Integer](#)

## Syntax

isexpanddownvalue = [objectreference].IsExpandDown

[objectreference].IsExpandDown = isexpanddownvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: IsExpandLeft property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_IEXPANDLEFT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to expand a layout object to the left to accommodate its contents.

## Data Type

[Integer](#)

## Syntax

isexpandleftvalue = [objectreference].IsExpandLeft

[objectreference].IsExpandLeft = isexpandleftvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: IsExpandRight property

{button ,AL(^H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DIVISIONINFO\_CLASS;H\_DROPAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLEAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_IEXPANDRIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to expand a layout object to the right to accommodate its contents.

### Data Type

[Integer](#)

### Syntax

isexpandrightvalue = [objectreference].IsExpandRight

[objectreference].IsExpandRight = isexpandrightvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## Word Pro: IsExpandUp property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_IEXPANDUP_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to expand a layout object upward to accommodate its contents.

## Data Type

[Integer](#)

## Syntax

isexpandupvalue = [objectreference].IsExpandUp

[objectreference].IsExpandUp = isexpandupvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage



## **Word Pro: IsExportedToNotesFX property**

{button ,AL('H\_BOOKMARK\_CLASS;H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IEXPORTEDTONOTESFX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether data will be transmitted from an OLE-linked document in Notes to a link in a Word Pro document.

### **Data Type**

[Integer](#)

### **Syntax**

isexportedtonotesfxvalue = [objectreference].IsExportedToNotesFX

[objectreference].IsExportedToNotesFX = isexportedtonotesfxvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This property is not used for a DdeLink object.

[Bookmark]

When you are working in an OLE-linked document and the document is opened, Notes can transmit data to enter into a Word Pro bookmark. When you close the Word Pro document, the data retransmits back to Notes.

### **Word Pro: IsExportedToOldNotesFX property**

{button ,AL('H\_BOOKMARK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IEXPORTEDTOOLDNOTESFX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property does not apply to or appear in Word Pro documents. Indicates whether data will be transmitted from Notes to a bookmark only from an Ami Pro document.

### **Data Type**

[Integer](#)

### **Syntax**

isexportedtooldnotesfxvalue = [objectreference].IsExportedToOldNotesFX

[objectreference].IsExportedToOldNotesFX = isexportedtooldnotesfxvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Only marked as Notes FX when you are working in an Ami Pro document. This property does not apply to or appear in Word Pro documents.

### **Word Pro: IsExternalFile property**

{button ,AL('H\_DIVISIONINFO\_CLASS;H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISEXTERNALFILE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

isexternalfilevalue = [objectreference].IsExternalFile

[objectreference].IsExternalFile = isexternalfilevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: IsFixedLength property**

{button ,AL(^H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS;0)} [See list of classes](#)

{button ,AL(^H\_ISFIXEDLENGTH\_PROPERTY\_EXSCRIPT;1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isfixedlengthvalue = [objectreference].IsFixedLength

[objectreference].IsFixedLength = isfixedlengthvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsFooter property

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCO  
NTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISFOOTER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns whether or not a container has a footer object.

## Data Type

Integer

## Syntax

isfootervalue = [objectreference].IsFooter

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: IsFrameMenuEnabled property**

{button ,AL('H\_CONTEXTMENUOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISFRAMEMENUENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Controls whether or not a frame menu displays in a frame.

**Data Type**

[Integer](#)

**Syntax**

isframemenuenabledvalue = [objectreference].IsFrameMenuEnabled

[objectreference].IsFrameMenuEnabled = isframemenuenabledvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Setting this property to True allows the frame menu to display when the focus is in a text object. Setting this property to False prevents the display of the frame menu when the focus is in a text object.

## **Word Pro: IsGrabBar property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISGRABBAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) An indicator that tells you if there is a solid color grab bar adjacent to the icons in a set. The grab bar allows you to drag the set to a different position on the workspace. Word Pro displays a hand over the solid color bar that closes when you click.

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].IsGrabBar = isgrabbarvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

You can write a script to turn this indicator off but, if you do so, the solid color grab bar disappears from the workspace.

## **Word Pro: IsGrammarError property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISGRAMMARERROR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isgrammarerrorvalue = [objectreference].IsGrammarError

[objectreference].IsGrammarError = isgrammarerrorvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



**Word Pro: IsGraphicMenuEnabled property**

{button ,AL('H\_CONTEXTMENUOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISGRAPHICMENUENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Controls whether or not a graphic menu displays in a graphic.

**Data Type**

[Integer](#)

**Syntax**

isgraphicmenuenabledvalue = [objectreference].IsGraphicMenuEnabled

[objectreference].IsGraphicMenuEnabled = isgraphicmenuenabledvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Setting this property to True allows the graphic menu to display when the focus is in a text object. Setting this property to False prevents the display of the graphic menu when the focus is in a text object.

## **Word Pro: IsHangover property**

{button ,AL('H\_KINSOKU\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISHANGOVER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ishangovervalue = [objectreference].IsHangover [objectreference].IsHangover = ishangovervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsHeader property

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCO  
NTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISHEADER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Returns whether or not a container has a header object.

## Data Type

Integer

## Syntax

isheadervalue = [objectreference].IsHeader

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

### **Word Pro: IsHiddenMark property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISHIDDENMARK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ishiddenmarkvalue = [objectreference].IsHiddenMark

[objectreference].IsHiddenMark = ishiddenmarkvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsHideHeaderFooter property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISHIDEHEADERFOOTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

ishideheaderfootervalue = [objectreference].IsHideHeaderFooter

[objectreference].IsHideHeaderFooter = ishideheaderfootervalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsHighlightNote property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISHIGHLIGHTNOTE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if you are in comment note mode.

### **Data Type**

Integer (Bool)

### **Syntax**

ishighlightnotevalue = [objectreference].IsHighlightNote

[objectreference].IsHighlightNote = ishighlightnotevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

If the value for this property is True (-1), then Word Pro is in comment note mode. If the value for this property is False (0), Word Pro is not in comment note mode.

## Word Pro: IsHonorProtected property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISHONORPROTECTED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to determine whether or not the protection should be honored for a layout object. This property is not implemented in Word Pro '97.

### Data Type

Integer

### Syntax

ishonorprotectedvalue = [objectreference].IsHonorProtected

[objectreference].IsHonorProtected = ishonorprotectedvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### Usage

Setting a layout object's IsHonorProtected property to False allows you to gain access to the layout object, even if it is protected. A user will also be able to access a protected layout object if its IsHonorProtected property is set to False.

**Word Pro: IsHorizontalScrollBarCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISHORIZONTALSCROLLBARCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

ishorizontalscrollbarcleanscrnvalue = [objectreference].IsHorizontalScrollBarCleanScrn

[objectreference].IsHorizontalScrollBarCleanScrn = ishorizontalscrollbarcleanscrnvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



**Word Pro: IsHoverHelp property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISHOVERHELP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not information about the icon displays in the title bar window when the cursor is positioned above the icon. This property does not apply to Word Pro.

**Data Type**

[Integer](#)

**Syntax**

isoverhelpvalue = [objectreference].IsHoverHelp

[objectreference].IsHoverHelp = isoverhelpvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Indicates whether or not information about the icon displays in the title bar window when the cursor is positioned above the icon. This property does not apply to Word Pro.

**Word Pro: IsIconDepressible property**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISICONDEPRESSIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) A flag that indicates whether or not an icon will depress if you click down on it.

**Data Type**

[Integer](#)

**Syntax**

[objectreference].IsIconDepressible = isicondepressiblevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

**Usage**

You must first select an icon using either the SelectStandardIcon or SelectCustomIcon method for this property to function.

**Word Pro: IsInBulletEditMode property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISINBULLETEDITMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not the current paragraph is in bullet edit mode.

**Data Type**

[Integer](#)

**Syntax**

isinbulleteditmodevalue = [objectreference].IsInBulletEditMode

[objectreference].IsInBulletEditMode = isinbulleteditmodevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

### **Word Pro: IsIndex property**

{button ,AL(^H\_INDEXSECTION\_CLASS;H\_SECTION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISINDEX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

isindexvalue = [objectreference].IsIndex

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsInDraft property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISINDRAFT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isindraftvalue = [objectreference].IsInDraft

[objectreference].IsInDraft = isindraftvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsInOutline property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISINOUTLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isinoutlinevalue = [objectreference].IsInOutline

[objectreference].IsInOutline = isinoutlinevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsInPageSort property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISINPAGESORT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isinpagesortvalue = [objectreference].IsInPageSort

[objectreference].IsInPageSort = isinpagesortvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

'Example: IsContents property

'This language element is not available for use.



'Example: IsContinuedFrom property

'This example script has not yet been created.

'Example: IsContinuedOn property

'This example script has not yet been created.

'Example: IsCumulative property

'This example script has not yet been created.

'Example: IsDataNameUsed method

'This example script has not yet been created.

'Example: IsDebug property

'This example script has not yet been created.

'Example: IsDisableWarningMessages property  
'This example script has not yet been created.

'Example: IsDisplayMisspelled property  
'This example script has not yet been created.

'Example: IsDivisionExternal property

'This example script has not yet been created.



'Example: IsDocLoading property

'This example script has not yet been created.

'Example: IsDocumentInRoute property

'This example script has not yet been created.

'Example: IsDoubleWordError property  
'This example script has not yet been created.

'Example: IsDraw property

'This example script has not yet been created.

'Example: IsEmptyDoc property

'This example script has not yet been created.

'Example: IsEmpty method

'This example script has not yet been created.

'Example: IsEmpty property

'This example script has not yet been created.

'Example: IsEnabled property

'This example script has not yet been created.



'Example: IsEquation property

'This example script has not yet been created.

'Example: IsErrorChecking property  
'This example script has not yet been created.

'Example: IsExpandDown property

'This example script has not yet been created.

'Example: IsExpandLeft property

'This example script has not yet been created.

'Example: IsExpandRight property

'This example script has not yet been created.

'Example: IsExpandUp property

'This example script has not yet been created.

'Example: IsExportedAsNotesFX method

' This example determines whether or not the DocInfo fields will be exported  
' to Lotus Notes fields for OLE Embedded Word Pro documents.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
Print .ActiveDocument.DocInfo.IsExportedAsNotesFX($LwpDocVarCreatedby )  
Print .ActiveDocument.DocInfo.IsExportedAsNotesFX($LwpDocVarDescription )  
Print .ActiveDocument.DocInfo.IsExportedAsNotesFX($LwpDocVarDocsize )
```

' This example inserts the 'created by' docinfo field. The field is made  
' Lotus Notes FX aware.

```
.ActiveDocument.DocInfo.IsExportedAsNotesFX $LwpDocVarCreatedby  
.InsertDocInfo $LwpDocVarCreatedby
```

'Example: IsExportedToNotesFX property  
'This example script has not yet been created.



'Example: IsExportedToOldNotesFX property  
'This example script has not yet been created.

'Example: IsExternalFile property

'This example script has not yet been created.

```
'Example: IsFilterTypePresent method
' This example prints all the text and table filter types to the Lotus Script
' Output panel.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Forall TextAndTable In .ApplicationWindow.Filter.TextAndTableImports
    FilterType = TextAndTable
    Print FilterType & " " &
        .ApplicationWindow.Filter.IsFilterTypePresent(FilterType,1)
End Forall
```

'Example: IsFixedLength property

'This example script has not yet been created.

'Example: IsFooter property

'This example script has not yet been created.

'Example: IsFrameMenuEnabled property  
'This example script has not yet been created.

'Example: IsGrabBar property

'This example script has not yet been created.

'Example: IsGrammarError property

'This example script has not yet been created.



'Example: IsGraphicMenuEnabled property  
'This example script has not yet been created.

'Example: IsHangover property

'This example script has not yet been created.

'Example: IsHeader property

'This example script has not yet been created.

'Example: IsHiddenMark property

'This example script has not yet been created.

'Example: IsHideHeaderFooter property

'This example script has not yet been created.

'Example: IsHighlightNote property  
'This example script has not yet been created.

'Example: IsHonorProtected property  
'This example script has not yet been created.

'Example: IsHorizontalScrollBarCleanScrn property  
'This example script has not yet been created.



'Example: IsHoverHelp property

'This example script has not yet been created.

'Example: IsIconDepressible property  
'This example script has not yet been created.

'Example: IsInBulletEditMode property

'This example script has not yet been created.

'Example: IsIndex property

'This example script has not yet been created.

'Example: IsInDraft property

'This example script has not yet been created.

'Example: IsInOutline property

'This example script has not yet been created.

'Example: IsInPageSort property

'This example script has not yet been created.

'Example: IsIntersectSiblings property

'This example script has not yet been created.



'Example: IsLastStop property

'This example script has not yet been created.

'Example: IsLesser property

'This example script has not yet been created.

'Example: IsLinked property

'This example script has not yet been created.

'Example: IsLocal property

'This example script has not yet been created.

'Example: IsLockedForRevisions property

'This example script has not yet been created.

'Example: IsLotusChart property

'This example script has not yet been created.

'Example: IsMarginSameAsParent property

'This example script has not yet been created.

'Example: IsMarginsInColor property

'This example script has not yet been created.



'Example: IsMarkerEqualToSelection method

'This example script has not yet been created.

'Example: IsMarkerValid property

'This example script has not yet been created.

'Example: IsMirrorImage property

'This example script has not yet been created.

'Example: IsMisspelled property

'This example script has not yet been created.

'Example: IsNotCopyable property

'This example script has not yet been created.

'Example: IsNotCopyImage property

'This example script has not yet been created.

'Example: IsNotGroupable property

'This example script has not yet been created.

'Example: IsNoUICommAllowed property

'This example script has not yet been created.



'Example: IsOnClipboard property

'This example script has not yet been created.

'Example: IsOpen property

'This example script has not yet been created.

'Example: IsOverridden property

'This example script has not yet been created.

'Example: IsOverride property

'This example script has not yet been created.

'Example: IsPageBreakAfter property  
'This example script has not yet been created.

'Example: IsPageBreakBefore property  
'This example script has not yet been created.

'Example: IsPageBreakMarks property  
'This example script has not yet been created.

'Example: IsPageBreakWithin property  
'This example script has not yet been created.



'Example: IsPageBreak property

'This example script has not yet been created.

'Example: IsParaDemandLoad property  
'This example script has not yet been created.

'Example: IsParagraphNumberingDown property  
'This example script has not yet been created.

'Example: IsParallelColumnsMenuEnabled property  
'This example script has not yet been created.

'Example: IsPartOfGroup property

'This example script has not yet been created.

'Example: IsPersistent property

'This example script has not yet been created.

'Example: IsPointWithin method

'This example script has not yet been created.

'Example: IsPrePrintedForm property  
'This example script has not yet been created.



'Example: IsPrintable property

'This example script has not yet been created.

'Example: IsPrintClickHereBlocks property  
'This example script has not yet been created.

'Example: IsPrintInBackground property  
'This example script has not yet been created.

'Example: IsPrivate property

'This example script has not yet been created.

'Example: IsProtected property

'This example script has not yet been created.

'Example: IsRegistered property

'This example script has not yet been created.

'Example: IsRepeat property

'This example script has not yet been created.

'Example: IsReplaceable property

'This example script has not yet been created.



'Example: IsReplacement property

'This example script has not yet been created.

'Example: IsResetParagraphNumber property

'This example script has not yet been created.

'Example: IsRevisionMark property

'This example script has not yet been created.

'Example: IsRevision property

'This example script has not yet been created.

'Example: IsSame property

'This example script has not yet been created.

'Example: IsSave property

'This example script has not yet been created.

'Example: IsScalableBorder property  
'This example script has not yet been created.

'Example: IsScalable property

'This example script has not yet been created.



'Example: IsScripting property

'This example script has not yet been created.

'Example: IsScrollable property

'This example script has not yet been created.

## Word Pro: IsIntersectSiblings property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISINTERSECTSIBLINGS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) For internal use only.

## Data Type

[Integer](#)

## Syntax

isintersectsiblingsvalue = [objectreference].IsIntersectSiblings

[objectreference].IsIntersectSiblings = isintersectsiblingsvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## **Word Pro: IsLastStop property**

{button ,AL('H\_MAILROUTING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISLASTSTOP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An internal flag that indicates whether or not the document has reached the last stop on the mail route.

### **Data Type**

[Integer](#)

### **Syntax**

islaststopvalue = [objectreference].IsLastStop

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

The flag indicates a True value if the document has reached the last stop on the mail route and a False value for all other stops.

**Word Pro: IsLesser property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISLESSER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

islesservalue = [objectreference].IsLesser

[objectreference].IsLesser = islesservalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsLinked property**

{button ,AL('H\_BOOKMARK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISLINKED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is obsolete and is not used.

**Data Type**

[Integer](#)

**Syntax**

islinkedvalue = [objectreference].IsLinked

[objectreference].IsLinked = islinkedvalue

**Legal values**

This property is not used.

**Usage**

This property is not used.

### Word Pro: IsLocal property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_COLU  
MNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTELA  
YOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLA  
SS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;  
H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHSTYLE_CLASS;H_ROWGROUPLAYOUT  
_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SU  
PERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTA  
BLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISLOCAL_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether a layout object was derived from the SmartMaster or was created locally.

### Data Type

[Integer](#)

### Syntax

```
islocalvalue = [objectreference].IsLocal
```

```
[objectreference].IsLocal = islocalvalue
```

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

This property is used in the same way when using this property from a CharacterStyle or ParagraphStyle object.

## **Word Pro: IsLockedForRevisions property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISLOCKEDFORREVISIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

islockedforrevisionsvalue = [objectreference].IsLockedForRevisions

[objectreference].IsLockedForRevisions = islockedforrevisionsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**



**Word Pro: IsLotusChart property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISLOTUSCHART\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

islotuschartvalue = [objectreference].IsLotusChart

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: IsMarginSameAsParent property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISMARGINSAMEASPARENT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines whether a layout object's margins are to be kept the same as its parent layout.

### Data Type

[Integer](#)

### Syntax

ismarginsameasparentvalue = [objectreference].IsMarginSameAsParent

[objectreference].IsMarginSameAsParent = ismarginsameasparentvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

This property allows you to control whether a child layout object maintains the same margin settings as its parent. For example, a header layout object usually adheres to its parent's margin values. A header layout object's parent is usually a page layout. If you set the header layout object's IsMarginSameAsParent property to False, the header layout object margins no longer correspond with the page layout object's margins.

## **Word Pro: IsMarginsInColor property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISMARGINSINCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ismarginsincolorvalue = [objectreference].IsMarginsInColor

[objectreference].IsMarginsInColor = ismarginsincolorvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: IsMarkerValid property**

{button ,AL(`H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ISMARKERVALID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

ismarkerValidvalue = [objectreference].IsMarkerValid

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsMirrorImage property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISMIRRORIMAGE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Not implemented.

## Data Type

Integer

## Syntax

ismirrorimagevalue = [objectreference].IsMirrorImage

[objectreference].IsMirrorImage = ismirrorimagevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: IsMisspelled property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISMISPELLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

ismisspelledvalue = [objectreference].IsMisspelled

[objectreference].IsMisspelled = ismisspelledvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: IsNotCopyable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISNOTCOPYABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates that a layout object cannot be copied.

## Data Type

Integer

## Syntax

isnotcopyablevalue = [objectreference].IsNotCopyable

[objectreference].IsNotCopyable = isnotcopyablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Setting a layout object's IsNotCopyable property to True disables the copy icon and the copy menu item while in that layout object. Setting this property to True will not disable the Windows keyboard shortcut commands for copying an item.

**Word Pro: IsNotCopyImage property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISNOTCOPYIMAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

isnotcopyImagevalue = [objectreference].IsNotCopyImage

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## Word Pro: IsNotGroupable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISNOTGROUPABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a layout can be grouped with any sibling layout objects.

## Data Type

[Integer](#)

## Syntax

isnotgroupablevalue = [objectreference].IsNotGroupable

[objectreference].IsNotGroupable = isnotgroupablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Setting this value to True will prevent a layout object from being grouped with sibling layout objects. For example, if you set this property to True for a certain frame layout object, you will not be able to group that frame with other frame layout objects.

## Word Pro: IsNoUICommAllowed property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISNOUICOMMALLOWED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not the user is locked out of the user interface for a layout object.

## Data Type

[Integer](#)

## Syntax

isnouicommallowedvalue = [objectreference].IsNoUICommAllowed

[objectreference].IsNoUICommAllowed = isnouicommallowedvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

If a layout object's IsNoUICommAllowed property is set to True, users will not be able to open the InfoBox directly to the layout object's properties. However, users will be able to access properties for a layout object if the InfoBox is already open when the user enters the layout object.

For example, if a frame layout object's IsNoUICommAllowed property is set to True, the Frame Properties menu option will be unavailable on both the right-click and pull-down menus. However, if the InfoBox is open when the user enters the frame layout, the user will be able to change the focus of the InfoBox to the frame.

This property is not meant to completely restrict access to the properties for a layout object.

**Word Pro: IsOnClipBoard property**

{button ,AL('H\_BOOKMARK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISONCLIPBOARD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is obsolete and is not used.

**Data Type**

[Integer](#)

**Syntax**

isonclipboardvalue = [objectreference].IsOnClipBoard

[objectreference].IsOnClipBoard = isonclipboardvalue

**Legal values**

This property is not used.

**Usage**

This property is not used.

## **Word Pro: IsOpen property**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISOPEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

isopenvalue = [objectreference].IsOpen

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsOverridden property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_CLICKHERE\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTEELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISOVERRIDDEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not the character styles or paragraph styles are overridden by local attributes anywhere in the current paragraph.

## Data Type

[Integer](#)

## Syntax

isoverriddenvalue = [objectreference].IsOverridden

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

[Layout]

If this property indicates that the layout object has been overridden, then the layout object either did not come from a SmartMaster or the layout object was modified after the SmartMaster was last loaded.

Check the StyleExceptions property to determine which attributes are overridden.

## Word Pro: IsOverride property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISOVERRIDE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a layout overrides another layout.

## Data Type

Integer

## Syntax

isoverridevalue = [objectreference].IsOverride

[objectreference].IsOverride = isoverridevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False.

## Usage

This property receives its instructions from the ConditionType property. IsOverride is currently only used when converting Ami Pro documents to Word Pro format.

## **Word Pro: IsPageBreakAfter property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPAGEBREAKAFTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ispagebreakaftervalue = [objectreference].IsPageBreakAfter

[objectreference].IsPageBreakAfter = ispagebreakaftervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsPageBreakBefore property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPAGEBREAKBEFORE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

ispagebreakbeforevalue = [objectreference].IsPageBreakBefore

[objectreference].IsPageBreakBefore = ispagebreakbeforevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## **Word Pro: IsPageBreakMarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPAGEBREAKMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

ispagebreakmarksvalue = [objectreference].IsPageBreakMarks

[objectreference].IsPageBreakMarks = ispagebreakmarksvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsPageBreakWithin property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPAGEBREAKWITHIN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

ispagebreakwithinvalue = [objectreference].IsPageBreakWithin

[objectreference].IsPageBreakWithin = ispagebreakwithinvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: IsPageBreak property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS';0)} [See list of classes](#)

{button ,AL('H\_ISPAGEBREAK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not a page break will be forced to occur beneath a row layout.

## Data Type

[Integer](#)

## Syntax

ispagebreakvalue = [objectreference].IsPageBreak

[objectreference].IsPageBreak = ispagebreakvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Equivalent to the "Page break after row" option, which is located in the Misc panel of the InfoBox for cell layout objects.

## **Word Pro: IsParaDemandLoad property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPARADEMANDLOAD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].IsParaDemandLoad = IsParaDemandLoadvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsParagraphNumberingDown property

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISPARAGRAPHNUMBERINGDOWN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not paragraph numbering will restart at the top of each table column.

### Data Type

Integer

### Syntax

isparagraphnumberingdownvalue = [objectreference].IsParagraphNumberingDown

[objectreference].IsParagraphNumberingDown = isparagraphnumberingdownvalue

### Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Equivalent to the "Restart paragraph numbers on each column" setting, which is located in the Misc panel of the InfoBox for table objects.

## **Word Pro: IsParagraphParent property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPARAGRAPH\_PARENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

isparagraphparentvariable = [objectreference].IsParagraphParent

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsParallelColumnsMenuEnabled property**

{button ,AL('H\_CONTEXTMENUOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPARALLELCOLUMNSMENUENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Controls whether or not a parallel columns menu displays in a parallel column.

**Data Type**

[Integer](#)

**Syntax**

isparallelcolumnsmenuenabledvalue = [objectreference].IsParallelColumnsMenuEnabled

[objectreference].IsParallelColumnsMenuEnabled = isparallelcolumnsmenuenabledvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Setting this property to True allows the parallel columns menu to display when the focus is in a text object. Setting this property to False prevents the display of the parallel columns menu when the focus is in a text object.

## Word Pro: IsPartOfGroup property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISPARTOFGROUP_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not a layout object is part of a group of layout objects.

## Data Type

[Integer](#)

## Syntax

ispartofgroupvalue = [objectreference].IsPartOfGroup

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage



**Word Pro: IsPersistent property**

{button ,AL('H\_SCRIPTDATASET\_CLASS;H\_WPDATASET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPERSISTENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether the data set is saved with the document between sessions.

**Data Type**

[Integer](#)

**Syntax**

ispersistentvalue = [objectreference].IsPersistent

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

This property returns True if the data set is saved with the document between sessions. This property returns False if the data set is not saved with the document between sessions.

**Usage**

All WPDataSets contained in a document are saved with the document. WPDataSets in the WPAApplication object are not saved with the document.

## **Word Pro: IsPrePrintedForm property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPREPRINTEDFORM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prevents protected text from being printed if "Allow editing of protected text" is deselected on the Other Panel of the TeamSecurity dialog box. This property also prevents lines and background colors/patterns in column blocks, frames, headers/footers, pages, paragraphs, and table cells from being printed.

### **Data Type**

Integer

### **Syntax**

ispreprintedformvalue = [objectreference].IsPrePrintedForm

[objectreference].IsPrePrintedForm = ispreprintedformvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing File - Print, clicking Options and selecting "On preprinted form."

## Word Pro: IsPrintable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISPRINTABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not the layout will be printed.

## Data Type

Integer

## Syntax

isprintablevalue = [objectreference].IsPrintable

[objectreference].IsPrintable = isprintablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This property does not affect the display or the flow of text. When a document is printed, layout objects that have the IsPrintable property set to True leave a blank space equal to the size of the layout object.

**Word Pro: IsPrintClickHereBlocks property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPRINTCLICKHEREBLOCKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines whether or not unfilled ClickHere blocks that contain prompt text are printed.

**Data Type**

[Integer](#)

**Syntax**

isprintclickhereblocksvalue = [objectreference].IsPrintClickHereBlocks

[objectreference].IsPrintClickHereBlocks = isprintclickhereblocksvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to choosing File - Print, clicking Options, and selecting "With unfilled click here blocks."

## **Word Pro: IsPrintInBackground property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISPRINTINBACKGROUND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if the "Print in background" option is turned on or off. Only valid for Word Pro 97 or Word Pro 96 running under Windows 3.1.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

isprintinbackgroundvalue = [objectreference].IsPrintInBackground

[objectreference].IsPrintInBackground = isprintinbackgroundvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Although a user can set this property, it has no effect when running WordPro 96 under Windows 95 or OS/2.

### **Word Pro: IsPrivate property**

{button ,AL(^H\_CHARACTERSTYLE\_CLASS;H\_CLICKHERE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_SILVERBULLET\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISPRIVATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether this characterstyle can be seen by a user as part of the selectable styles.

### **Data Type**

Integer

### **Syntax**

isprivatevalue = [objectreference].IsPrivate

[objectreference].IsPrivate = isprivatevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsProtected property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISPROTECTED_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a layout object is marked as protected.

## Data Type

[Integer](#)

## Syntax

isprotectedvalue = [objectreference].IsProtected

[objectreference].IsProtected = isprotectedvalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

Equivalent to protecting a layout object by choosing the option in the "Other options" box, located on the Misc panel of the InfoBox for certain layout objects.

### **Word Pro: IsRegistered property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISREGISTERED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

isregisteredvalue = [objectreference].IsRegistered

[objectreference].IsRegistered = isregisteredvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



**Word Pro: IsRepeat property**

{button ,AL('H\_FOOTNOTEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISREPEAT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isrepeatvalue = [objectreference].IsRepeat

[objectreference].IsRepeat = isrepeatvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsReplaceable property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_CONTENT\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_FORMULA\_CLASS;H\_GLOSSARY\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS;H\_PARALLEL\_COLUMNS\_CLASS;H\_SUPERTABLE\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISREPLACEABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not you can replace a content object with another content object.

## **Data Type**

Integer

## **Syntax**

isreplaceablevalue = [objectreference].IsReplaceable

## **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## **Usage**

This property is useful for importing content from other documents. Text is replaceable only if the content object is empty. If content is present, the content object is not replaceable. If only a ClickHere block, the content object can be replaced.

**Word Pro: IsReplacement property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISREPLACEMENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if the "replace current document" option is turned on or off.

**Data Type**

[Integer \(Bool\)](#)

**Syntax**

isreplacementvalue = [objectreference].IsReplacement

[objectreference].IsReplacement = isreplacementvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Although a user can set this property, it has no effect when running Word Pro 96 under Windows 95 or OS/2. Word Pro 96 for Windows 95 and OS/2 automatically add any newly opened documents to the desktop.

### **Word Pro: IsResetParagraphNumber property**

{button ,AL(`H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_ISRESETPARAGRAPHNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether paragraph numbering is cumulative across cells within a table.

### **Data Type**

Integer

### **Syntax**

isresetparagraphnumbervalue = [objectreference].IsResetParagraphNumber

[objectreference].IsResetParagraphNumber = isresetparagraphnumbervalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Setting this property to True will force paragraph numbering to restart between each table cell. Setting this property to false will allow the IsParagraphNumberingDown property to control in which circumstances paragraph numbering will reset.

### **Word Pro: IsRevisionMark property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISREVISIONMARK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

isrevisionmarkvalue = [objectreference].IsRevisionMark

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsRevision property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISREVISION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not a layout object has been revised.

## Data Type

[Integer](#)

## Syntax

isrevisionvalue = [objectreference].IsRevision

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: IsSame property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) Icon bar objects that have this flag checked are placed in the same location on the workspace where they were on the previous close. Moving just one of the bars with this flag set will cause all bars with this flag set to move.

**Data Type**

[Integer](#)

**Syntax**

[objectreference].IsSame = issamevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to setting this indicator in the "Bars that will appear in the same location" field in the SmartIcons Setup dialog box. All context icon bar objects have this flag set to True by default (except for the Universal set). For example, if you are working in text and move the Text icon bar object to the bottom of the workspace, and then start to work in a frame, the Frame icon bar object will also display on the bottom of the workspace.

## **Word Pro: IsSave property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSAVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) An indicator that tells you whether or not an icon bar object will be saved. If you move the icon bar to another location on the workspace or you add an icon to it, the icon bar object must be saved to a file in order to preserve your changes.

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].IsSave = issavevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

If this flag is on, when Word Pro closes, icon bar information is saved. If the flag is turned off, no changes made to the icon bar are saved. Therefore, if you moved a bar, or added or removed icons from a bar while Word Pro was active, none of this information would be saved.



**Word Pro: IsScalableBorder property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSCALABLEBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) An indicator that tells you whether or not you can change (scale) the size of an icon bar object.

**Data Type**

[Integer](#)

**Syntax**

[objectreference].IsScalableBorder = isscalablebordervalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

**Usage**

This flag is set to True by default. If it is not turned on, you cannot drag the icon bar set to a new size.

**Word Pro: IsScalable property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSCALABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

isscalablevalue = [objectreference].IsScalable

**Legal values**

True False

**Usage**

## Word Pro: IsScripting property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISSCRIPTING_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) A flag that indicates whether an Ami Pro layout object had a macro that should run. For internal use only.

## Data Type

[Integer](#)

## Syntax

isscriptingvalue = [objectreference].IsScripting

[objectreference].IsScripting = isscriptingvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: IsScrollable property

{button ,AL(^H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DIVISIONINFO\_CLASS;H\_DROPAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLEAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISSCROLLABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines whether or not a scroll bar is inserted in a note layout object.

## Data Type

[Integer](#)

## Syntax

isscrollablevalue = [objectreference].IsScrollable

[objectreference].IsScrollable = isscrollablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

## Usage

This property is only used with note layout objects.

## Word Pro: IsShowing property

{button ,AL(^H\_ICONBAR\_CLASS;H\_ICONBARMANAGER\_CLASS;H\_RULER\_CLASS;H\_SECTIONTABS\_CLASS',  
0)} [See list of classes](#)

{button ,AL(^H\_ISSHOWING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether a specific user interface object is currently visible.

[IconBar]

Indicates whether the icon bar object is currently visible.

[IconBarManager]

A flag that Indicates whether or not any icon bars will be visible.

[Ruler]

Indicates whether or not the ruler object is currently visible.

[SectionTabs]

Indicates whether or not a section or division tab in a document is currently visible.

## Data Type

[Integer](#)

## Syntax

isshowingvalue = [objectreference].IsShowing

[objectreference].IsShowing = isshowingvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

[IconBarManager]

Equivalent to choosing View - Show/Hide - SmartIcons on the menu.

[SectionTabs]

Equivalent to clicking the tab icon at the right of the divider tab area to display or hide divider tabs.

## Word Pro: IsSingleClickEntry property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISSINGLECLICKENTRY_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not a layout can be entered with a single click.

## Data Type

[Integer](#)

## Syntax

issingleclickentryvalue = [objectreference].IsSingleClickEntry

[objectreference].IsSingleClickEntry = issingleclickentryvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: IsSizable property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISSIZABLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set whether or not sizing handles will function on the layout object.

## Data Type

Integer

## Syntax

issizablevalue = [objectreference].IsSizable

[objectreference].IsSizable = issizablevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

If a layout object's IsSizable property is set to False, users will not be able to size the layout object with the InfoBox or by using the mouse.

### **Word Pro: IsSizingViaMouse property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISSIZINGVIAMOUSE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not you can use the mouse to size table columns.

### **Data Type**

Integer

### **Syntax**

issizingviamousevalue = [objectreference].IsSizingViaMouse

[objectreference].IsSizingViaMouse = issizingviamousevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: IsSmartCorrectEnabled property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSMARTCORRECTENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

issmartcorrectenabledvalue = [objectreference].IsSmartCorrectEnabled

[objectreference].IsSmartCorrectEnabled = issmartcorrectenabledvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsSmartShadeEnabled property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSMARTSHADEENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

issmartshadeenabledvalue = [objectreference].IsSmartShadeEnabled

[objectreference].IsSmartShadeEnabled = issmartshadeenabledvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: IsSnapTo property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISSNAPTO_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a child layout snaps to the grid of the current layout.

## Data Type

[Integer](#)

## Syntax

issnaptovalue = [objectreference].IsSnapTo

[objectreference].IsSnapTo = issnaptovalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Equivalent to the "Snap frames to grid" setting, located on the Misc panel of the InfoBox for certain layout objects.

**Word Pro: IsSortFromEnd property**

{button ,AL('H\_SORTKEY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSORTFROMEND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies whether or not words or numbers are placed first or last in a row.

**Data Type**

[Integer](#)

**Syntax**

issortfromendvalue = [objectreference].IsSortFromEnd

[objectreference].IsSortFromEnd = issortfromendvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

This property returns True if the word being placed either first or last in the row is selected from the end of the field.

This property returns False if the word being placed either first or last in the row is selected from the beginning of the field.

**Usage**

Equivalent to choosing Sort - Text and selecting an option in the "Word" box.

## **Word Pro: IsSpellBarUp property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSPELLBARUP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

isspellbarupvalue = [objectreference].IsSpellBarUp

[objectreference].IsSpellBarUp = isspellbarupvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

## **Word Pro: IsSpellMode property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSPELLMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isspellmodevalue = [objectreference].IsSpellMode

[objectreference].IsSpellMode = isspellmodevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: IsSqueeze property**

{button ,AL('H\_KINSOKU\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSQUEEZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

issqueezevalue = [objectreference].IsSqueeze [objectreference].IsSqueeze = issqueezevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsStyleSheet property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISSTYLESHEET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

isstylesheetvalue = [objectreference].IsStyleSheet

[objectreference].IsStyleSheet = isstylesheetvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## Word Pro: IsTableHeading property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_ISTABLEHEADING_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to make a row layout or row group layout object a table heading.

## Data Type

[Integer](#)

## Syntax

istableheadingvalue = [objectreference].IsTableHeading

[objectreference].IsTableHeading = istableheadingvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

Equivalent to choosing "Mark as Repeated Heading" in the Table menu.

## **Word Pro: IsTemp property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISTEMP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A value of True indicates that this style will be removed when the InUseCount property value drops to 0.

### **Data Type**

Integer

### **Syntax**

istempvalue = [objectreference].IsTemp

[objectreference].IsTemp = istempvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsTextLocked property**

{button ,AL('H\_DIVISIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IStEXTLOCKED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

istextlockedvalue = [objectreference].IsTextLocked

[objectreference].IsTextLocked = istextlockedvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsTextMenuEnabled property**

{button ,AL('H\_CONTEXTMENUOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_IStEXTMENUENABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Controls whether or not a text menu displays when you click in a text object.

**Data Type**

[Integer](#)

**Syntax**

istextmenuenabledvalue = [objectreference].IsTextMenuEnabled

[objectreference].IsTextMenuEnabled = istextmenuenabledvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Setting this property to True allows the Text menu to display when the focus is in a text object. Setting this property to False prevents the display of the Text menu when the focus is in a text object.

## Word Pro: IsTOC property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_ISTOC_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates whether or not a table is a special TOC table.

## Data Type

[Integer](#)

## Syntax

istocvalue = [objectreference].IsTOC

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: IsTrueType property**

{button ,AL('H\_FONT\_CLASS;H\_FONTMETRICS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISTRUETYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

istruetypevalue = [objectreference].IsTrueType

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsUndoOn property**

{button ,AL('H\_DIVISION\_CLASS;H\_FOUNDRY\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISUNDOON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Turns the Undo recorder on (True) or off (False) for the objects in the Foundry object.

### **Data Type**

[Integer](#)

### **Syntax**

isundoonvalue = [objectreference].IsUndoOn

[objectreference].IsUndoOn = isundoonvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Word Pro is capable of recording all of your actions and allowing you to undo those actions in succession. Word Pro does this by recording any changes made to an object and keeping track of each object's status before and after each change.

You can use this property to turn the Undo recorder on or off for all the objects in a particular Foundry object. This is particularly useful when you want to run a lengthy script that makes many changes to a document. By turning the Undo recorder off for each active division's Foundry, you will save memory, prevent users from undoing your script's changes, preserve any changes made by the user, and possibly improve the overall performance of your script.

## **Word Pro: IsUpdateAutomatic property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISUPDATEAUTOMATIC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that indicates if updates will automatically be sent to Word Pro.

### **Data Type**

[Integer](#)

### **Syntax**

isupdateautomaticvalue = [objectreference].IsUpdateAutomatic

[objectreference].IsUpdateAutomatic = isupdateautomaticvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

You can turn the "Automatic update" option on and off in the Manage Links dialog box by choosing Edit - Manage Links. You can also choose to update manually.



### **Word Pro: IsUpdateIndex property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISUPDATEINDEX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines whether or not Word Pro updates the index while printing a document.

### **Data Type**

[Integer](#)

### **Syntax**

isupdateindexvalue = [objectreference].IsUpdateIndex

[objectreference].IsUpdateIndex = isupdateindexvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing File - Print, clicking Options, and selecting "Index" in the Update section in the Print Options dialog box.

### **Word Pro: IsUpdateTOC property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISUPDATETOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines whether or not Word Pro updates the table of contents while printing a document.

### **Data Type**

[Integer](#)

### **Syntax**

isupdatetocvalue = [objectreference].IsUpdateTOC

[objectreference].IsUpdateTOC = isupdatetocvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing File - Print, clicking Options, and selecting "Table of contents" in the Update section in the Print Options dialog box.

## **Word Pro: IsValid property**

{button ,AL('H\_BASEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVALID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates if the object from which you call this property is available to LotusScript at the time the script is run. This is particularly useful when you need to determine if a MenuItem object is available from the current context.

### **Data Type**

[Integer](#)

### **Syntax**

isvalidvalue = [objectreference].IsValid

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Word Pro often hides or deactivates certain objects under specific circumstances. For example, the Frame menu and its menu items are marked as invalid (IsValid = 0), unless a frame is active in the focus. Use this property when you want to check to see if an object is available to LotusScript from the current context.

Some objects, such as WPAApplication, are always valid.

**Word Pro: IsVerticalScrollBarCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVERTICALSCROLLBARCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isverticalscrollbarcleanscrnvalue = [objectreference].IsVerticalScrollBarCleanScrn

[objectreference].IsVerticalScrollBarCleanScrn = isverticalscrollbarcleanscrnvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewAnchor property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWANCHOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewanchorvalue = [objectreference].IsViewAnchor

[objectreference].IsViewAnchor = isviewanchorvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewBookmarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWBOOKMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewbookmarksvalue = [objectreference].IsViewBookmarks

[objectreference].IsViewBookmarks = isviewbookmarksvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsViewClickHereBlocks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWCLICKHEREBLOCKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewclickhereblocksvalue = [objectreference].IsViewClickHereBlocks

[objectreference].IsViewClickHereBlocks = isviewclickhereblocksvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewColGuides property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWCOLGUIDES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewcolguidesvalue = [objectreference].IsViewColGuides

[objectreference].IsViewColGuides = isviewcolguidesvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



**Word Pro: IsViewColumnBreakMarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWCOLUMNBREAKMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewcolumnbreakmarksvalue = [objectreference].IsViewColumnBreakMarks

[objectreference].IsViewColumnBreakMarks = isviewcolumnbreakmarksvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsViewDDELinks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWDDELINKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewddelinksvalue = [objectreference].IsViewDDELinks

[objectreference].IsViewDDELinks = isviewddelinksvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewHorzRuler property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWHORZRULER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewhorzrulervalue = [objectreference].IsViewHorzRuler

[objectreference].IsViewHorzRuler = isviewhorzrulervalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewHorzScrollBar property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWHORZSCROLLBAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewhorzscrollbarvalue = [objectreference].IsViewHorzScrollBar

[objectreference].IsViewHorzScrollBar = isviewhorzscrollbarvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewMenuCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWMENUCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewmenucleanscrnvalue = [objectreference].IsViewMenuCleanScrn

[objectreference].IsViewMenuCleanScrn = isviewmenucleanscrnvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsViewNotes property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWNOTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewnotesvalue = [objectreference].IsViewNotes

[objectreference].IsViewNotes = isviewnotesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewOutlineFlowToScreen property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWOUTLINEFLOWTOSCREEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewoutlineflowtoscreenvalue = [objectreference].IsViewOutlineFlowToScreen

[objectreference].IsViewOutlineFlowToScreen = isviewoutlineflowtoscreenvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewOutlineIndent property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWOUTLINEINDENT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewoutlineindentvalue = [objectreference].IsViewOutlineIndent

[objectreference].IsViewOutlineIndent = isviewoutlineindentvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



## **Word Pro: IsViewPageLayoutMarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWPAGELAYOUTMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewpagelayoutmarksvalue = [objectreference].IsViewPageLayoutMarks

[objectreference].IsViewPageLayoutMarks = isviewpagelayoutmarksvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewParallelColumnBorder property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWPARALLELCOLUMNBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewparallelcolumnbordervalue = [objectreference].IsViewParallelColumnBorder

[objectreference].IsViewParallelColumnBorder = isviewparallelcolumnbordervalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewPictures property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWPICTURES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewpicturesvalue = [objectreference].IsViewPictures

[objectreference].IsViewPictures = isviewpicturesvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewReturnIconCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWRETURNICONCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewreturniconcleanscrnvalue = [objectreference].IsViewReturnIconCleanScrn

[objectreference].IsViewReturnIconCleanScrn = isviewreturniconcleanscrnvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewReturns property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWRETURNS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewreturnsvalue = [objectreference].IsViewReturns

[objectreference].IsViewReturns = isviewreturnsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: IsViewRulerMarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWRULERMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewrulermarksvalue = [objectreference].IsViewRulerMarks

[objectreference].IsViewRulerMarks = isviewrulermarksvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsViewSectionBreakMarks property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWSECTIONBREAKMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewsectionbreakmarksvalue = [objectreference].IsViewSectionBreakMarks

[objectreference].IsViewSectionBreakMarks = isviewsectionbreakmarksvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewSectionTabs property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWSECTIONTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewsectiontabsvalue = [objectreference].IsViewSectionTabs

[objectreference].IsViewSectionTabs = isviewsectiontabsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**



**Word Pro: IsViewSmartIconsCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWSMARTICONSLEANSRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewsmarticonsleanscrnvalue = [objectreference].IsViewSmartIconsCleanScrn

[objectreference].IsViewSmartIconsCleanScrn = isviewsmarticonsleanscrnvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: IsViewStatusBarCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWSTATUSBARCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewstatusbarcleanscrnvalue = [objectreference].IsViewStatusBarCleanScrn

[objectreference].IsViewStatusBarCleanScrn = isviewstatusbarcleanscrnvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsViewTableGridLines property**

{button ,AL(^H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ISVIEWTABLEGRIDLINES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewtablegridlinesvalue = [objectreference].IsViewTableGridLines

[objectreference].IsViewTableGridLines = isviewtablegridlinesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: IsViewTableHeadings property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWTABLEHEADINGS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewtableheadingsvalue = [objectreference].IsViewTableHeadings

[objectreference].IsViewTableHeadings = isviewtableheadingsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewTabs property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewtabsvalue = [objectreference].IsViewTabs

[objectreference].IsViewTabs = isviewtabsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

### **Word Pro: IsViewThumbBarCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWTHUMBBARCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

isviewthumbbarcleanscrnvalue = [objectreference].IsViewThumbBarCleanScrn

[objectreference].IsViewThumbBarCleanScrn = isviewthumbbarcleanscrnvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: IsViewTitleBarCleanScrn property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ISVIEWTITLEBARCLEANSCRN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

isviewtitlebarcleanscrnvalue = [objectreference].IsViewTitleBarCleanScrn

[objectreference].IsViewTitleBarCleanScrn = isviewtitlebarcleanscrnvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

'Example: IsShowing property

```
' This example uses IsShowing to test the status of the iconbar
  IsShowing = .ApplicationWindow.IconBarManager.IsShowing
If IsShowing = True Then
' if the icons are currently showing then toggle to hide
  .ToggleIconBar
End If
```



'Example: IsSingleClickEntry property  
'This example script has not yet been created.

'Example: IsSizable property

'This example script has not yet been created.

'Example: IsSizingViaMouse property  
'This example script has not yet been created.

'Example: IsSmartCorrectEnabled property

'This example script has not yet been created.

'Example: IsSmartShadeEnabled property  
'This example script has not yet been created.

'Example: IsSnapTo property

'This example script has not yet been created.

'Example: IsSortFromEnd property

'This example script has not yet been created.

'Example: IsSpellBarUp property

'This example script has not yet been created.



'Example: IsSpellMode property

'This example script has not yet been created.

'Example: IsSqueeze property

'This example script has not yet been created.

'Example: IsStyleSheet property  
'This example script has not yet been created.

'Example: IsTableHeading property

'This example script has not yet been created.

'Example: IsTemporary method

'This example script has not yet been created.

'Example: IsTemp property

'This example script has not yet been created.

'Example: IsTextLocked property

'This example script has not yet been created.

'Example: IsTextMenuEnabled property  
'This example script has not yet been created.



'Example: IStOC property

'This example script has not yet been created.

'Example: IsTrueType property

'This example script has not yet been created.

'Example: IsUndoOn property

''This example turns the Undo recorder off in the currently active division and then turns it on again.

'Paste this example in the Sub Main section of your script.

.Foundry.IsUndoOn = False

.Foundry.IsUndoOn = True

'Example: IsUpdateAutomatic property  
'This example script has not yet been created.

'Example: IsUpdateIndex property

'This example script has not yet been created.

'Example: IsUpdateTOC property

'This example script has not yet been created.

'Example: IsValid property

'This example script has not yet been created.

'Example: IsVerticalScrollBarCleanScrn property  
'This example script has not yet been created.



'Example: IsViewAnchor property

'This example script has not yet been created.

'Example: IsViewBookmarks property

'This example script has not yet been created.

'Example: IsViewClickHereBlocks property  
'This example script has not yet been created.

'Example: IsViewColGuides property  
'This example script has not yet been created.

'Example: IsViewColumnBreakMarks property  
'This example script has not yet been created.

'Example: IsViewDDELinks property

'This example script has not yet been created.

'Example: IsViewHorzRuler property  
'This example script has not yet been created.

'Example: IsViewHorzScrollBar property

'This example script has not yet been created.



'Example: IsViewMenuCleanScrn property

'This example script has not yet been created.

'Example: IsViewNotes property

'This example script has not yet been created.

'Example: IsViewOutlineFlowToScreen property  
'This example script has not yet been created.

'Example: IsViewOutlineIndent property  
'This example script has not yet been created.

'Example: IsViewPageLayoutMarks property  
'This example script has not yet been created.

'Example: IsViewParallelColumnBorder property  
'This example script has not yet been created.

'Example: IsViewPictures property

'This example script has not yet been created.

'Example: IsViewReturnIconCleanScrn property  
'This example script has not yet been created.



'Example: IsViewReturns property

'This example script has not yet been created.

'Example: IsViewRulerMarks property  
'This example script has not yet been created.

'Example: IsViewSectionBreakMarks property  
'This example script has not yet been created.

'Example: IsViewSectionTabs property

'This example script has not yet been created.

'Example: IsViewSmartIconsCleanScrn property  
'This example script has not yet been created.

'Example: IsViewStatusBarCleanScrn property  
'This example script has not yet been created.

'Example: IsViewTableGridLines property  
'This example script has not yet been created.

'Example: IsViewTableHeadings property  
'This example script has not yet been created.



'Example: IsViewTabs property

'This example script has not yet been created.

'Example: IsViewThumbBarCleanScrn property  
'This example script has not yet been created.

'Example: IsViewTitleBarCleanScrn property  
'This example script has not yet been created.

'Example: IsViewVertRuler property  
'This example script has not yet been created.

'Example: IsViewVertScrollBar property  
'This example script has not yet been created.

```
'Example: IsWMCommandValid method
' This example uses the IsWmCommandvalid to ensure that a table is selected
' before querying for table information.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
If (.IsWmCommandvalid(633)) Then
    Dim MyTable As Table
    Set MyTable = .Table
    Print "Table Name: " & MyTable.Name
    Print "Table Rows:" & Str$(MyTable.NumRows)
    Print "Table Cols:" & Str$(MyTable.NumCols)
End If
```

'Example: IsWordProChart property

'This example script has not yet been created.

'Example: IsWorkingDir property

'This example script has not yet been created.



'Example: Italic method

' This example first inserts sample text in the current document and selects  
' the paragraph. The script then uses the Italic method to toggle the Italic  
' attribute.

' RUNTIME DEPENDENCIES: You must have a document open with selected text  
' for this script to work.

.Text.InsertText "This is some sample text."  
.SelectParagraph

**.Italic**

'Example: *Italic property*

'This example script has not yet been created.

'Example: Items property

'This example script has not yet been created.

```
'Example: Item method
' This example creates a bookmark from the current marker and then prints
' the bookmark's marker name by indexing into the Bookmarks collection using
' the Item method
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim MarkerName As String
MarkerName = .Mark($LwpMarkerTypeBookmark)
.Division.BookmarkManager.AddBookmark "ExampleBookmark", MarkerName
Print .Division.BookmarkManager.Bookmarks.Item("ExampleBookmark").MarkerName
```

'Example: JoinCorners property

'This example script has not yet been created.

'Example: JoinHeight property

'This example script has not yet been created.

'Example: JoinType property

'This example script has not yet been created.

'Example: JoinWidth property

'This example script has not yet been created.



'Example: Join property

'This example script has not yet been created.

'Example: Justifiable property

'This example script has not yet been created.

'Example: KeepWithNext property

'This example script has not yet been created.

'Example: KeepWithPrev property  
'This example script has not yet been created.

'Example: KeyboardLanguage property  
'This example script has not yet been created.

'Example: KeyStroke event

'This example script has not yet been created.

'Example: KeywordColor property

'This example script has not yet been created.

'Example: Keywords property

'This example script has not yet been created.



'Example: Kinsoku property

'This example script has not yet been created.

'Example: LandscapeMode property

'This example script has not yet been created.

'Example: Language property

'This example script has not yet been created.

'Example: LastChild property

'This example script has not yet been created.

'Example: LastDocPath property

'This example script has not yet been created.

'Example: LastEditorName property

'This example script has not yet been created.

'Example: LastGraphicPath property  
'This example script has not yet been created.

'Example: LastGraphicType property  
'This example script has not yet been created.



'Example: LastMacroPath property

'This example script has not yet been created.

'Example: LastName property

'This example script has not yet been created.

'Example: LastPage property

'This example script has not yet been created.

'Example: LastUsedDateFormula property

'This example script has not yet been created.

'Example: Last property

'This example script has not yet been created.

'Example: LayoutName property

'This example script has not yet been created.

'Example: LayoutOverride property

'This example script has not yet been created.

'Example: Layouts property

'This example script has not yet been created.



'Example: Layout property

'This example script has not yet been created.

'Example: LeaderDotDashChar property  
'This example script has not yet been created.

'Example: LeaderDotDotChar property

'This example script has not yet been created.

'Example: LeaderDotType property

'This example script has not yet been created.

'Example: LeaderDotUnderscoreChar property  
'This example script has not yet been created.

'Example: LeadingText property

'This example script has not yet been created.

'Example: LeftBorder property

'This example script has not yet been created.

'Example: LeftExternalMargin property

'This example script has not yet been created.



'Example: LeftIntArea property

'This example script has not yet been created.

'Example: LeftPage property

'This example script has not yet been created.

'Example: LeftTopCellId property

'This example script has not yet been created.

'Example: Left property

'This example script has not yet been created.

'Example: Length property

'This example script has not yet been created.

'Example: Level property

'This example script has not yet been created.

'Example: LineLocation property

'This example script has not yet been created.

'Example: LineMix property

'This example script has not yet been created.



'Example: LineNumberOptions property  
'This example script has not yet been created.

'Example: LinePlacement property

'This example script has not yet been created.

'Example: LinesSpacedEveryNthUnit property  
'This example script has not yet been created.

'Example: LineValid property

'This example script has not yet been created.

'Example: LinkAvailable property

'This example script has not yet been created.

'Example: LinkDisplayNameFileLength property  
'This example script has not yet been created.

'Example: LinkDisplayName property

'This example script has not yet been created.

'Example: LinkedFileName property

'This example script has not yet been created.



'Example: Linked property

'This example script has not yet been created.

'Example: LinkGraphic property

'This example script has not yet been created.

```
'Example: Link method
' This example copies the current division to the file LINKFILE.LWP.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim DivId as String
Dim LinkFile as String

LinkFile = ..ActiveDocument.Path & "\LINKFILE.LWP"
DivId = .Division.Name

.ActiveDocument.Divisions(DivId).Link LinkFile, "Lotus Word Pro"
```

'Example: ListCount property

'This example script has not yet been created.

'Example: LoadFilesMaximized property

'This example script has not yet been created.

'Example: Localize method

'This example script has not yet been created.

'Example: Location property

'This example script has not yet been created.

'Example: Locked property

'This example script has not yet been created.



'Example: LockForNotesUserName property

'This example script has not yet been created.

'Example: Locks property

'This example script has not yet been created.

'Example: LowerCaseAscent property

'This example script has not yet been created.

'Example: LowerCase method

' This example toggles the lowercase attribute of the selected text, displays  
' a message box, then toggles the lowercase again.

' RUNTIME DEPENDENCIES: You must have a document open and some text selected  
' for this script to work.

**.LowerCase**

MessageBox "Click OK undo lowercase change.",MB\_OK,"Example Script"

**.LowerCase**

'Example: LowerCase property

'This example script has not yet been created.

'Example: LwpMenuBar property

'This example script has not yet been created.

'Example: MacroAppend method

'This example script has not yet been created.

'Example: MacroCancel method

'This example script has not yet been created.



'Example: MacroCompile method

'This example script has not yet been created.

'Example: MacroEndRecord method

'This example script has not yet been created.

'Example: MacroName property

'This example script has not yet been created.

'Example: MacroPaths property

'This example script has not yet been created.

'Example: MacroPath property

'This example script has not yet been created.

'Example: MacroPlay method

'This example script has not yet been created.

'Example: MacroQuickPlay method

'This example script has not yet been created.

'Example: MacroQuickRecord method

'This example script has not yet been created.



'Example: MacroRecord method

'This example script has not yet been created.

'Example: MacroResume method

'This example script has not yet been created.

'Example: MacroRun method

'This example script has not yet been created.

'Example: MacroStatus property

'This example script has not yet been created.

'Example: Macro property

'This example script has not yet been created.

'Example: MailDocument method

'This example script has not yet been created.

'Example: MailRouteingPtr property

'This example script has not yet been created.

'Example: MailRouting property

'This example script has not yet been created.



'Example: MaintainAspectRatio property

'This example script has not yet been created.

'Example: MaintainEditor property

'This example script has not yet been created.

```
'Example: MakeTableFromText method
' This example inserts 5 rows and 5 columns of text into the current document
' where each column is seperated by a tab and each row by a paragraph break.
' The text is then selected and converted into a table.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim RowCount as Integer
Dim ColumnCount as Integer
Dim OutputText as string

'Insert text in 5 tabbed rows
For RowCount = 1 To 5
  For ColumnCount = 1 To 5
    OutputText = "r" & RowCount & "c" & ColumnCount
    .Text.InsertText OutputText
    .Text.InsertTab
  Next
  .Text.SplitParagraph
Next

'Select the text
.Text.Shade $LwpLocationTypeParagraph,$LwpNavigateDirectionLeft,5

'Convert to table
.MakeTableFromText
```

'Example: MakeUniqueLinkName method

' This example prints a unique DDE link name to the Lotus Script Output panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .Division.DdeLinkManager.MakeUniqueLinkName("DDELink")

'Example: ManualFrame method

' This example opens the Create Frame dialog box so you can create a  
' frame manually.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ManualFrame

```
'Example: ManualTable method
' This example opens the Create Table dialog box so you can create a
' table manually.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ManualTable
```

'Example: MarginBottom property

'This example script has not yet been created.

'Example: MarginColor property

'This example script has not yet been created.



'Example: MarginLeft property

'This example script has not yet been created.

'Example: MarginRight property

'This example script has not yet been created.

'Example: MarginTop property

'This example script has not yet been created.

'Example: MarkCharacter property

'This example script has not yet been created.

'Example: MarkerClass property

'This example script has not yet been created.

'Example: MarkerName property

'This example script has not yet been created.

'Example: Markers property

'This example script has not yet been created.

'Example: MarkPosition property

'This example script has not yet been created.



'Example: MarkRevisionInsert method  
'This example script has not yet been created.

'Example: MarkType property

'This example script has not yet been created.

'Example: Mark method

' This example creates a bookmark, adds some text to it and then expands it to cover the inserted text.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MarkerName as String

MarkerName = .Text.Mark(\$LwpMarkerTypeBookmark)

.Division.BookmarkManager.AddBookmark "NewBookMark" , MarkerName

.Text.InsertText "This is a new bookmark"

.Text.Mark \$LwpMarkerTypeBookmark,MarkerName,\$LwpRangePartEnd

'Example: MasterName property

'This example script has not yet been created.

'Example: Master property

'This example script has not yet been created.

'Example: MatchType property

'This example script has not yet been created.

'Example: MaxBottomBorder property

'This example script has not yet been created.

'Example: MaxBottomGutter property

'This example script has not yet been created.



'Example: MaxContentHeight property

'This example script has not yet been created.

'Example: MaxContentWidth property

'This example script has not yet been created.

'Example: MaxHorzPaneDistance property

'This example script has not yet been created.

'Example: MaxHyphLines property

'This example script has not yet been created.

'Example: MaxIdenticalConsecSentOpens property  
'This example script has not yet been created.

'Example: MaxIdenticalSentOpensWithin10 property  
'This example script has not yet been created.

'Example: MaximizeOnStartUp property  
'This example script has not yet been created.

'Example: Maximize method

' This example maximizes the main application window.

.ApplicationWindow.Maximize



'Example: MaximumWordsinaSentence property  
'This example script has not yet been created.

'Example: MaxLeftBorder property

'This example script has not yet been created.

'Example: MaxLeftGutter property

'This example script has not yet been created.

'Example: MaxNumColsAllowed property

'This example script has not yet been created.

'Example: MaxNumRowsAllowed property

'This example script has not yet been created.

'Example: MaxRightBorder property

'This example script has not yet been created.

'Example: MaxRightGutter property

'This example script has not yet been created.

'Example: MaxSplitCols property

'This example script has not yet been created.



'Example: MaxSplitRows property

'This example script has not yet been created.

'Example: MaxTopBorder property

'This example script has not yet been created.

'Example: MaxTopGutter property

'This example script has not yet been created.

'Example: MaxVertPaneDistance property

'This example script has not yet been created.

'Example: MergeContinue method

'This example script has not yet been created.

'Example: MergeInfoPtr property

'This example script has not yet been created.

'Example: MergeOptions property

'This example script has not yet been created.

```
'Example: MergeSetDataFile method
' This example assigns the merge data file 'DATAFILE.LWP' to the currently
' active document.
' RUNTIME DEPENDENCIES: You must have a document open and a file named
' 'DATAFILE.LWP' located in the Word Pro default documents directory for this
' script to work.

Dim DataFile As String
Dim Status As Integer

DataFile = .ApplicationWindow.UserInterfacePrefs.DocPath
DataFile = DataFile & "\DATAFILE.LWP"

.ApplicationWindow.ActiveDocument.MergeOptions.MergeStepNumber = $LwpMergeStepNumber1

Status = .MergeSetDataFile (DataFile, 0)
If Status = False Then
    Exit Sub
End If
```



```
'Example: MergeStart method
' This example merges data for the current merge document.
' RUNTIME DEPENDENCIES: You must have a document open which has been assigned
' to a merge data file has inserted merge fields for this script to work.
```

```
Dim stat As Integer
```

```
' Set up to merge and view
.ApplicationWindow.ActiveDocument.MergeOptions.Options = &H2
.ApplicationWindow.ActiveDocument.MergeOptions.MergeStepNumber =
    $LwpMergeStepNumber3
```

**.MergeStart**

```
Do
    stat = .Merge($LwpMergeActionNextRecord)
    .Merge $LwpMergeActionMergeOne
Loop Until stat = False

.Merge $LwpMergeActionClose
.CloseMergeDataFile
```

'Example: MergeStepNumber property

'This example script has not yet been created.

## Word Pro: MaxIdenticalConsecSentOpens property

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXIDENTICALCONSECSSENTOPENS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates how many identical openers for consecutive sentences will be flagged by Grammar Check in a document.

### Data Type

[Integer](#)

### Syntax

maxidenticalconsecsentopensvalue = [objectreference].MaxIdenticalConsecSentOpens

[objectreference].MaxIdenticalConsecSentOpens = maxidenticalconsecsentopensvalue

### Legal values

The legal values for this property range from 0 (never flag consecutive sentence openers) to 9 (flag 9 or more consecutive sentence openers). Default is 3.

### Usage

Use this property to set a maximum number of identical sentence openers for consecutive sentences in Grammar Check. Choose from 0 to 9. Equivalent to choosing Edit - Check Grammar, clicking Options, and entering a number in the Maximum number of identical sentence openers section, "For consecutive sentences" field, on the Grammatical Style panel.

## **Word Pro: MaxIdenticalSentOpensWithin10 property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXIDENTICALSENTOPENSWITHIN10\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates how many identical sentence openers, within 10 sentences, will be flagged by Grammar Check in a document.

### **Data Type**

Integer

### **Syntax**

maxidenticalsentopenswithin10value = [objectreference].MaxIdenticalSentOpensWithin10

[objectreference].MaxIdenticalSentOpensWithin10 = maxidenticalsentopenswithin10value

### **Legal values**

The legal values for this property range from 0 (never flag consecutive sentence openers) to 9 (flag 9 or more consecutive sentence openers within 10 sentences). Default is 3.

### **Usage**

Use this property to set a maximum number of identical sentence openers within 10 sentences in Grammar Check. Choose from 0 to 9. Equivalent to choosing Edit - Check Grammar, clicking Options, and entering a number in the Maximum number of identical sentence openers section, "Within 10 sentences" field, on the Grammatical Style panel.

## **Word Pro: MaximizeOnStartUp property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXIMIZEONSTARTUP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Value to indicate if the "Load Word Pro maximized" option is enabled in Word Pro Preferences.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

maximizeonstartupvalue = [objectreference].MaximizeOnStartUp

[objectreference].MaximizeOnStartUp = maximizeonstartupvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to the "Load Word Pro maximized" option on the General panel of the Word Pro Preferences dialog box. If the value is True (-1), Word Pro loads maximized. If the value is False (0), Word Pro loads in the smaller window.

## **Word Pro: MaximumWordsinaSentence property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXIMUMWORDSINASENTENCE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates a maximum number of words that will be allowed in a sentence before Grammar Check alerts you .

### **Data Type**

Integer

### **Syntax**

maximumwordsinasentencevalue = [objectreference].MaximumWordsinaSentence

[objectreference].MaximumWordsinaSentence = maximumwordsinasentencevalue

### **Legal values**

The legal values for this property range from 25 to 75. Default is 35.

### **Usage**

Use this property to set a maximum number of words in a sentence. Choose from 25 to 75. Equivalent to choosing Edit - Check Grammar, clicking Options, and entering a number in the "Maximum number of words per sentence" field on the Grammatical Style panel.

**Word Pro: MaxLeftBorder property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXLEFTBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the left border of a row in a table object.

**Data Type**

Long

**Syntax**

maxleftbordervalue = [objectreference].MaxLeftBorder

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

### **Word Pro: MaxLeftGutter property**

{button ,AL(`H\_Basetable\_Class;H\_FOOTNOTETABLE\_Class;H\_GLOSSARY\_Class;H\_PARALLELCOLUMNS\_Class;H\_TABLE\_Class;H\_TABLEHEADING\_Class',0)} [See list of classes](#)

{button ,AL(`H\_MAXLEFTGUTTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the left gutter of a row in a table object.

### **Data Type**

Long

### **Syntax**

maxleftguttervalue = [objectreference].MaxLeftGutter

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**



### **Word Pro: MaxNumColsAllowed property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXNUMCOLSALLOWED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the maximum number of columns that can be inserted without widening the table container. Word Pro uses the default column width to determine the number of columns inserted.

### **Data Type**

Integer

### **Syntax**

maxnumcolsallowedvalue = [objectreference].MaxNumColsAllowed

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: MaxNumRowsAllowed property**

{button ,AL('H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXNUMROWSALLOWED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the maximum number of rows that can be inserted without enlarging the table container.

### **Data Type**

[Integer](#)

### **Syntax**

maxnumrowsallowedvalue = [objectreference].MaxNumRowsAllowed

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: MaxRightBorder property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXRIGHTBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the right border of a row in a table object.

**Data Type**

Long

**Syntax**

maxrightbordervalue = [objectreference].MaxRightBorder

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

**Word Pro: MaxRightGutter property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXRIGHTGUTTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the right gutter of a row in a table object.

**Data Type**

Long

**Syntax**

maxrightguttervalue = [objectreference].MaxRightGutter

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

## **Word Pro: MaxSplitCols property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXSPLITCOLS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the maximum number of columns into which a table cell can be split.

### **Data Type**

Integer

### **Syntax**

maxsplitcolsvalue = [objectreference].MaxSplitCols

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

When table cells are split, Word Pro inserts the number of columns you specify into the current cell, and places the contents of the cell into the upper left cell of the newly created cells.

## **Word Pro: MaxSplitRows property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXSPLITROWS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the maximum number of rows into which a table cell can be split.

### **Data Type**

Integer

### **Syntax**

maxsplitrowsvalue = [objectreference].MaxSplitRows

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

When table cells are split, Word Pro inserts the number of rows you specify into the current cell, and places the contents of the cell into the upper left cell of the newly created cells.

### **Word Pro: MaxTopBorder property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXTOPBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the top border of a row in a table object.

### **Data Type**

Long

### **Syntax**

maxtopbordervalue = [objectreference].MaxTopBorder

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

### **Word Pro: MaxTopGutter property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_MAXTOPGUTTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the specific maximum width that can be set for the top gutter of a row in a table object.

### **Data Type**

Long

### **Syntax**

maxtopguttervalue = [objectreference].MaxTopGutter

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**



## **Word Pro: MaxVertPaneDistance property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MAXVERTPANEDISTANCE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Long](#)

### **Syntax**

maxvertpanedistancevalue = [objectreference].MaxVertPaneDistance

[objectreference].MaxVertPaneDistance = maxvertpanedistancevalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

**Word Pro: MergeInfoPtr property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGEINFOPTR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Do not write to this property.

**Data Type**

[Long](#)

**Syntax**

mergeinfoPtrvalue = [objectreference].MergeInfoPtr

[objectreference].MergeInfoPtr = mergeinfoPtrvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Word Pro uses this property to access other Merge classes that are not available to users.

## **Word Pro: MergeStepNumber property**

{button ,AL('H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MERGESTEPNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) An indicator which you can use to set the steps involved in the Merge process.

### **Data Type**

Variant (Enumerated)

### **Syntax**

mergestepnumbervalue = [objectreference].MergeStepNumber

[objectreference].MergeStepNumber = mergestepnumbervalue

### **Legal values**

\$LwpMergeStepNumber0 (1505) The first step of a Merge (selecting a data file).

\$LwpMergeStepNumber1 (1506) Inserts a merge field.

\$LwpMergeStepNumber2 (1507) Sets a delimiter.

\$LwpMergeStepNumber3 (1508) Prints the Merge data file.

### **Usage**

Set these values before using the WMCommand in order to bring up the dialog box of your choice.

## Word Pro: MinBottomMargin property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_MINBOTTOMMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The minimum value to which a layout object's bottom margin is allowed to be set.

### Data Type

Long

### Syntax

minbottommarginvalue = [objectreference].MinBottomMargin

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## Word Pro: MinHeight property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_MINHEIGHT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The minimum value to which a layout object's height can be set.

## Data Type

Long

## Syntax

minheightvalue = [objectreference].MinHeight

[objectreference].MinHeight = minheightvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

Controls the height of a layout object, so that if the content becomes smaller than a specific height, then the layout object does not become smaller than that specific height. The effect of setting this property can be seen in frames which have automatic sizing enabled.

## Word Pro: MinLeftMargin property

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL(^H_MINLEFTMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The minimum value to which the left margin can be set.

### Data Type

[Long](#)

### Syntax

minleftmarginvalue = [objectreference].MinLeftMargin

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## Word Pro: MinRightMargin property

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL(^H_MINRIGHTMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The minimum value to which the right margin can be set.

### Data Type

Long

### Syntax

minrightmarginvalue = [objectreference].MinRightMargin

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## Word Pro: MinTopMargin property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_MINTOPMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The minimum value to which the top margin can be set.

## Data Type

[Long](#)

## Syntax

mintopmarginvalue = [objectreference].MinTopMargin

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage



## **Word Pro: ModifiedDateString property**

{button ,AL('H\_DOCINFO\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MODIFIEDDATESTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the date the file was last changed as a string.

### **Data Type**

[String](#)

### **Syntax**

modifieddatestringvalue = [objectreference].ModifiedDateString

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: ModifiedTimeString property**

{button ,AL('H\_DOCINFO\_CLASS;H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MODIFIEDTIMESTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the time the file was last changed as a string.

### **Data Type**

String

### **Syntax**

modifiedtimestringvalue = [objectreference].ModifiedTimeString

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: ModifiedTimeValue property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MODIFIEDTIMEVALUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Returns the time the file was last changed as a long data type.

**Data Type**

Long

**Syntax**

modifiedtimevaluevalue = [objectreference].ModifiedTimeValue

[objectreference].ModifiedTimeValue = modifiedtimevaluevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The value that this property returns represents the number of seconds that have elapsed since midnight on January 1, 1970.

## **Word Pro: MouseButtonForManipulation property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOUSEBUTTONFORMANIPULATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MouseButton

### **Syntax**

mousebuttonformanipulationvalue = [objectreference].MouseButtonForManipulation

[objectreference].MouseButtonForManipulation = mousebuttonformanipulationvalue

### **Legal values**

\$LwpMouseButtonLeft (1509)

\$LwpMouseButtonMiddle (1511)

\$LwpMouseButtonRight (1510)

### **Usage**

## **Word Pro: MouseButtonForSelection property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOUSEBUTTONFORSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MouseButton

### **Syntax**

mousebuttonforselectionvalue = [objectreference].MouseButtonForSelection

[objectreference].MouseButtonForSelection = mousebuttonforselectionvalue

### **Legal values**

\$LwpMouseButtonLeft (1509)

\$LwpMouseButtonMiddle (1511)

\$LwpMouseButtonRight (1510)

### **Usage**

## **Word Pro: MouseProperty property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOUSEPROPERTY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MouseButton

### **Syntax**

mousepropertyvalue = [objectreference].MouseProperty

[objectreference].MouseProperty = mousepropertyvalue

### **Legal values**

\$LwpMouseButtonLeft (1509)

\$LwpMouseButtonMiddle (1511)

\$LwpMouseButtonRight (1510)

### **Usage**

## **Word Pro: MultiCellPasteOn property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MULTICELLPASTEON\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

multicellpasteonvalue = [objectreference].MultiCellPasteOn

[objectreference].MultiCellPasteOn = multicellpasteonvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: MultiCompareParaTagSet property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MULTICOMPAREPARATAGSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

multicompareparatagsetvalue = [objectreference].MultiCompareParaTagSet

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: MultiCompareParaTag property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MULTICOMPAREPARATAG\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

multicompareparatagvalue = [objectreference].MultiCompareParaTag

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: NameBasedOnStyle property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_NAMEBASEDONSTYLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The name of the style from which the layout object was created.

## Data Type

String

## Syntax

namebasedonstylevalue = [objectreference].NameBasedOnStyle

[objectreference].NameBasedOnStyle = namebasedonstylevalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

When setting the NameBasedOnStyle property, be sure to use the proper case for the layout style being assigned. For example, "Default Frame" is not equivalent to "default frame."

## **Word Pro: NameForFilters property**

{button ,AL('H\_NOTELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NAMEFORFILTERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the user that created a specific comment note.

### **Data Type**

String

### **Syntax**

nameforfiltersvalue = [objectreference].NameForFilters

[objectreference].NameForFilters = nameforfiltersvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

The default value for this property is contained in the "User Name" setting, which is located in the Personal panel of the Word Pro Preferences dialog box. Modifying the NameForFilters property of a specific comment note does not change the name setting under Word Pro Preferences.

## Word Pro: Name property

{button ,AL('H\_BASEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The Name property contains the name of an object. This can be useful when trying to access an object from the object's collection. Some objects allow you to change their names. Other objects are named internally by Word Pro and do not allow you to change their names.

This property is inherited by all Word Pro objects, but not every object makes use of this property.

### Data Type

String

### Syntax

namevalue = [objectreference].Name

[objectreference].Name = namevalue

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage

Some of the objects which make use of this property include WPAApplication, TextDocument, Division, DivisionInfo, Frame, Table, ClickHere, and Bookmark. The information stored in this property and the proper use of that information is detailed below.

*CurrentApplication*.Name

The name of the application which, in this case, is always "Lotus Word Pro".

*<TextDocumentobject>*.Name

The name of the document represented by the TextDocument on which the Name property is located.

*<Divisionobject>*.Name

The internal division name (a hexadecimal value) which uniquely identifies a division.

*<DivisionInfoobject>*.Name

The external division name (a string value) which represents the name seen in the division tab.

*<Layoutobject>*.Name

Returns the name of a frame, a table, or other object that uses a layout, such as FrameLayout or TableLayout.

*<ClickHereobject>*.Name

The name of the ClickHere object from which you called the Name property. You must use the ClickHereCollection object to access ClickHeres.

*<Bookmarkobject>*.Name

The name of the Bookmark object from which you called the Name property. You must use the BookmarkCollection object in the BookmarksByMarkerName property of a division to access the bookmarks by name.

### **Word Pro: NewDocMacroName property**

{button ,AL('H\_AUTORUNMACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWDOCMACRONAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

newdocmacrovalue = [objectreference].NewDocMacroName

[objectreference].NewDocMacroName = newdocmacrovalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: NewFile property**

{button ,AL('H\_REVIEWVERSIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWFILE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].NewFile = newfilevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: NewVersion property**

{button ,AL('H\_REVIEWVERSIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWVERSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].NewVersion = newversionvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: NextClickHere property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTCLICKHERE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the ClickHere bloick which is next in the division (uses Tab order).

**Data Type**

[String](#)

**Syntax**

nextclickHerevalue = [objectreference].NextClickHere

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**



### **Word Pro: NextName property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

nextnamevalue = [objectreference].NextName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: NextNeighbor property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTNEIGHBOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

nextneighborvalue = [objectreference].NextNeighbor

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: NextSection property**

{button ,AL('H\_INDEXSECTION\_CLASS;H\_SECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTSECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

String

**Syntax**

nextsectionvalue = [objectreference].NextSection

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: NextStyleName property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

nextstylevalue = [objectreference].NextStyleName

[objectreference].NextStyleName = nextstylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: NextText property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

nexttextvalue = [objectreference].NextText

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: NoHyphenate property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NOHYPHENATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

nohyphenatevalue = [objectreference].NoHyphenate

[objectreference].NoHyphenate = nohyphenatevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: NormalParagraph property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NORMALPARAGRAPH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not any local paragraph styles have been set in the current paragraph.

### **Data Type**

[Integer](#)

### **Syntax**

normalparagraphvalue = [objectreference].NormalParagraph

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: Normal property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NORMAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

normalvalue = [objectreference].Normal

[objectreference].Normal = normalvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



**Word Pro: NotesFlow property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NOTESFLOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read) Indicates whether or not Notes flow is currently active.

**Data Type**

[Integer](#)

**Syntax**

notesflowvalue = [objectreference].NotesFlow

[objectreference].NotesFlow = notesflowvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You can use this property to determine if Notes flow is currently active. You cannot turn the property on or off because it is read-only.

```
'Example: Merge method
' This example merges data for the current merge document.
' RUNTIME DEPENDENCIES: You must have a document open which has been assigned
' to a merge data file has inserted merge fields for this script to work.

Dim stat As Integer

' Set up to merge and view
.ApplicationWindow.ActiveDocument.MergeOptions.Options = &H2
.ApplicationWindow.ActiveDocument.MergeOptions.MergeStepNumber =
    $LwpMergeStepNumber3
.MergeStart

Do
    stat = .Merge($LwpMergeActionNextRecord)
    .Merge $LwpMergeActionMergeOne
Loop Until stat = False
```

'Example: Messages method

' This example shows how to provide a default response to a dialog box. The

' MessageBox dialog is never displayed here.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Messages \$LwpTakeDefaultMsgboxAnswer

MessageBox "Click OK ",MB\_OK,"Example Script"

'Example: MinBottomMargin property

'This example script has not yet been created.

'Example: MinHeight property

'This example script has not yet been created.

'Example: Minimize method

' This example minimizes the main application window.

.ApplicationWindow.Minimize

'Example: MinLeftMargin property

'This example script has not yet been created.

'Example: MinRightMargin property

'This example script has not yet been created.



'Example: MinTopMargin property

'This example script has not yet been created.

```
'Example: MirrorPage method
' This example sets up a left/right (complex) page.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
.Page.Layout.RightPage.MarginLeft = 720
.Page.Layout.IsComplex = True
.Page.Layout.LeftPage.MirrorPage
```

'Example: ModifiedDateString property

'This example script has not yet been created.

'Example: ModifiedTimeString property

'This example script has not yet been created.

'Example: ModifiedTimeValue property

'This example script has not yet been created.

```
'Example: MorphSelectionToTable method
' This example inserts 5 rows and 5 columns of text into the current document
' where each column is separated by a tab and each row by a paragraph break.
' The text is then selected and morphed into a table.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim RowCount as Integer
Dim ColumnCount as Integer
Dim OutputText as string
For RowCount = 1 To 5
    For ColumnCount = 1 To 5
        OutputText = "r" & RowCount & "c" & ColumnCount
        .Text.InsertText OutputText
        .Text.InsertTab
    Next
    .Text.SplitParagraph
Next
.Text.Shade $LwpLocationTypeParagraph,$LwpNavigateDirectionLeft,5
.Text.MorphSelectionToTable
```

'Example: MostRecentVersion method

'This example script has not yet been created.

'Example: MouseButtonForManipulation property  
'This example script has not yet been created.



'Example: MouseButtonForSelection property  
'This example script has not yet been created.

'Example: MouseDown event

'This example script has not yet been created.

'Example: MouseProperty property

'This example script has not yet been created.

'Example: MouseUp event

'This example script has not yet been created.

'Example: MoveDivision method

' This example moves the division named 'Division1' to last division in the  
' current document.

' RUNTIME DEPENDENCIES: You must have a document open containing 2 or more  
' divisions with one of the divisions named 'Division1' for this script to  
' work.

Dim Div As Division

Set Div = Bind("!Division1")

DivName = Div.Name

.MoveDivision DivName

```
'Example: MoveDown method
' This example inserts 5 lines into the current document and then moves the
' cursor up 4 lines and then down 2.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Dim LineNumber as Integer
For LineNumber = 1 To 5
    .Type("Line " & LineNumber )
    .Text.SplitParagraph
Next
MessageBox "Click OK to move up 4 lines.",MB_OK,"Example Script"
.Text.MoveUp(4)
MessageBox "Click OK to move down 2 lines.",MB_OK,"Example Script"
.Text.MoveDown(2)
```

'Example: Moved event

'This example script has not yet been created.

```

'Example: MoveItem method
' This example adds a new menu item name 'New Menu' to the File menu just
' below the Save item. All items from the Edit menu are then moved to
' 'New Menu'.
' RUNTIME DEPENDENCIES: You must have not deleted the Edit or File menus
' for this script to work.

Dim MenuName as String
Dim SourceMenu As MenuItem
Dim DestinationMenu As MenuItem
Dim MenuSpacer as String
MenuSpacer = Chr$(8)
MenuName = "&New Edit"

' Set DestinationMenu to the File menu
Set DestinationMenu=.Applicationwindow.LwpMenuBar.Items.Item("&File")

' Set SourceMenu to the Edit Menu
Set SourceMenu = .ApplicationWindow.LwpMenuBar.Items.Item("&Edit")

' Create a new Edit Menu
DestinationMenu.DeleteItem "My Edit"
DestinationMenu.NewItem MenuName,,0,"&Save" & MenuSpacer & "Ctrl+S"

' Copy all the items from the Edit Menu to My new Menu
Forall Items In SourceMenu.Items
    DestinationMenu.Items(MenuName).MoveItem Items, True, ,
End Forall

```



```
'Example: MoveParagraph method
' This example inserts two paragraphs into the current document and then
' moves the second paragraph above the first.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "This is the first paragraph inserted.[Enter]"
.Type "This is the second paragraph inserted. It "
.Type "will be moved before the first paragraph."
.Text.MoveParagraph $LwpMoveDirectionUp
```

'Example: MoveToBack method

'This example script has not yet been created.

```
'Example: MoveToEnd method
' This example inserts 5 sentences of text into the current document.
' The cursor is then positioned to the beginning of the paragraph, and then
' then advanced to the next word, sentence, line and paragraph.
' RUNTIME DEPENDENCIES: You must have a document open and the cursor located
' in a multilined paragraph for this script to work.
```

```
Dim SentenceNumber as Integer
Dim WordNumber as Integer
For SentenceNumber = 1 To 5
    For WordNumber = 1 To 5
        .Text.InsertText "Word" & Format$(WordNumber) & " "
    Next
    .Type (". ")
Next
.Text.MoveToStart $LwpLocationTypeParagraph
.Text.MoveToEnd $LwpLocationTypeWord
.Text.MoveToEnd $LwpLocationTypeSentence
.Text.MoveToEnd $LwpLocationTypeLine
.Text.MoveToEnd $LwpLocationTypeParagraph
```

'Example: MoveToFront method

'This example script has not yet been created.

```
'Example: MoveToStart method
' This example inserts 5 sentences of text into the current document.
' The cursor is then positioned to the beginning of the paragraph.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceNumber as Integer
Dim WordNumber as Integer
For SentenceNumber = 1 To 5
    For WordNumber = 1 To 5
        .Text.InsertText "Word" & Format$(WordNumber) & " "
    Next
    .Type (". ")
Next
.Text.MoveToStart $LwpLocationTypeParagraph
```

```
'Example: MoveUp method
' This example inserts 5 lines into the current document and then moves the
' cursor up 4 lines.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Dim LineNumber as Integer
For LineNumber = 1 To 5
    .Type("Line " & LineNumber )
    .Text.SplitParagraph
Next
MessageBox "Click OK to move up 4 lines.",MB_OK,"Example Script"
.Text.MoveUp(4)
```

```
'Example: Move method
' This example moves the Word Pro's application window.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim XPosition as Long
Dim YPosition as Long
XPosition = 1440
YPosition = 2880
.ApplicationWindow.Move XPosition,YPosition
```

'Example: MultiCellPasteOn property  
'This example script has not yet been created.



'Example: MultiCompareParaTagSet property  
'This example script has not yet been created.

'Example: MultiCompareParaTag property  
'This example script has not yet been created.

'Example: NameBasedOnStyle property

'This example script has not yet been created.

'Example: NameForFilters property

'This example script has not yet been created.

'Example: Name property

'This example script has not yet been created.

'Example: Negative property

'This example script has not yet been created.

'Example: NewDivision method

' This example creates a new division in the active document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewDivision

'Example: NewDocMacroName property  
'This example script has not yet been created.



'Example: NewDocument method

' This example creates a new document based on the 'DEFAULT.MWP' SmartMaster.

.NewDocument , , "DEFAULT.MWP", ,

'Example: NewFile property

'This example script has not yet been created.

'Example: NewFrame method

' This example creates a new frame 2 inches by 2 inches in the current  
' document. The frame's upper left corner is positioned 1 inch from the left  
' edge of the page and 2 inches down from the top of the page.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.NewFrame 2880, 2880, 1440, 2880

```
'Example: NewItem method
' This example creates a new menu item named 'Example Menu' to the File
' menu. The mnuMenuSub subroutine is assigned to run each time the new
' menu item is selected
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim NewMenu As MenuItem
Dim MenuName as String
Dim MenuSpacer as String

MenuName = "&Example Menu"
MenuSpacer = Chr$(8)

' Set menu object
Set NewMenu =.ApplicationWindow.LwpMenuBar.Items.Item("&File")

' Create a new menu off of the File Menu and before the Save option
' Delete it first to prevent duplicates
NewMenu.DeleteItem MenuName
NewMenu.NewItem MenuName,"!mnuMenuSub",0,"&Save" & MenuSpacer & "Ctrl+S"
End Sub
```

'Example: NewVersion property

'This example script has not yet been created.

'Example: NewWindow method

' This example creates a new document window based upon the current document.  
' A message box then prompts you to close the new window.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

**.NewWindow**

```
Messagebox "Click OK to close the current window.",MB_OK,"Example Script"  
.CloseDocWindow
```

'Example: NextClickHere property

'This example script has not yet been created.

'Example: NextItem method

'This example script has not yet been created.



'Example: NextName property

'This example script has not yet been created.

'Example: NextNeighbor property

'This example script has not yet been created.

'Example: NextSection property

'This example script has not yet been created.

'Example: NextStyleName property

'This example script has not yet been created.

'Example: NextText property

'This example script has not yet been created.

'Example: NextVersion method

'This example script has not yet been created.

'Example: Next method

'This example script has not yet been created.

'Example: NoHyphenate property

'This example script has not yet been created.



'Example: NormalParagraph property

'This example script has not yet been created.

'Example: NormalText method

' This example first inserts sample text in the current document, selects  
' the paragraph, and makes that paragraph bold. The script then uses the  
' NormalText method to return the selected paragraph to the normal attributes  
' for that paragraph style.  
' RUNTIME DEPENDENCIES: You must have a document open with selected text  
' for this script to work.

```
.Text.InsertText "This is some sample text."  
.SelectParagraph  
.Bold
```

```
.NormalText
```

'Example: Normal property

'This example script has not yet been created.

'Example: NoteColor property

'This example script has not yet been created.

'Example: NoteLayouts property

'This example script has not yet been created.

'Example: NotesFlow property

'This example script has not yet been created.

'Example: NumberEveryNthLine property

'This example script has not yet been created.

'Example: NumberFound property

'This example script has not yet been created.



'Example: NumberingPosition property  
'This example script has not yet been created.

'Example: NumberingStyleName property

'This example script has not yet been created.

'Example: Numbering property

'This example script has not yet been created.

'Example: NumberOfDataFields property

'This example script has not yet been created.

'Example: NumberOfReplacements property

'This example script has not yet been created.

'Example: NumberOfRevisions property

'This example script has not yet been created.

'Example: NumberOfVersions method

' This example prints the number of versions for the current document to the

' Lotus Script Output panel.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .ActiveDocument.VersionManager.NumberOfVersions(\$LwpVersionObjectType)

'Example: NumberSequenceName property

'This example script has not yet been created.



'Example: NumberWhichLines property  
'This example script has not yet been created.

'Example: Number property

'This example script has not yet been created.

'Example: NumCharsInDoc property

'This example script has not yet been created.

'Example: NumCharsInParagraph property

'This example script has not yet been created.

'Example: NumColsSpannedOneCell property  
'This example script has not yet been created.

'Example: NumCols property

'This example script has not yet been created.

'Example: NumContainers property

'This example script has not yet been created.

'Example: NumDecimalPlaces property

'This example script has not yet been created.



'Example: NumericFormat property

'This example script has not yet been created.

'Example: NumFields property

'This example script has not yet been created.

'Example: NumLinesOfSpaceAbove property

'This example script has not yet been created.

'Example: NumLinesOfSpaceBelow property

'This example script has not yet been created.

'Example: NumLinesOfSpace property

'This example script has not yet been created.

'Example: NumOfRecentFiles property

'This example script has not yet been created.

'Example: NumPagesInDoc property

'This example script has not yet been created.

'Example: NumParagraphs property

'This example script has not yet been created.



'Example: NumRowsSpannedOneCell property

'This example script has not yet been created.

'Example: NumRowsThatFit property

'This example script has not yet been created.

'Example: NumRowsToFit property

'This example script has not yet been created.

'Example: NumRows property

'This example script has not yet been created.

'Example: NumTabs property

'This example script has not yet been created.

'Example: NumUndoLevels property

'This example script has not yet been created.

'Example: NumWindowsViewingDoc property

'This example script has not yet been created.

'Example: NumWordsInDoc property

'This example script has not yet been created.



'Example: ObjectType property

'This example script has not yet been created.

'Example: Object property

'This example script has not yet been created.

'Example: Oblique property

'This example script has not yet been created.

'Example: Offset property

'This example script has not yet been created.

## Word Pro: MoveDivision method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVEDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the specified division to the place specified by the Parent, DivisionLocation, or NeighborName parameters. If you do not specify where the division should be moved, it will be moved inside the current division.

### Syntax

[objectreference].MoveDivision(DivisionName,[Parent,] [DivisionLocation,] [NeighborName])

### Parameters

#### *DivisionName*

A String expression which specifies the internal name of the division you want to move.

#### *Parent*

A String expression representing the internal name of the division which you want to become the parent of the division you are moving. Optional parameter.

#### *DivisionLocation*

A String or Integer value which indicates where you want the division to be moved in relation to the currently active division. Data type is Variant which allows the value of this parameter to be one of the three division locations listed below or its numeric equivalent (in parentheses). Default is \$LwpDivLocInsertAtInsertionPt.

\$LwpDivLocInsertBeforeCurrentdiv (184) Moves the division to a position before the currently active division.

\$LwpDivLocInsertAfterCurrentdiv (185) Moves the division to a position after the currently active division.

\$LwpDivLocInsertAtInsertionPt (186) Moves the division to the insertion point. All items that fall before the insertion point remain part of the active division. All items after the insertion point become part of the moved division.

**Note** If the insertion point is in a table cell or a frame, Word Pro splits the contents of the cell or frame, leaving the items before the insertion point intact and moving the items after the insertion point into the moved division. Items outside the cell or frame are not affected and remain in the original division.

#### *NeighborName*

A String expression representing the name of the division which you want to become the neighbor of the moved division. Optional parameter.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

For more information on division names in LotusScript, see [Overview: Division names in LotusScript](#)

**Word Pro: MoveDown method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVEDOWN\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point down the specified number of times. Each down movement is equivalent to pressing the Down arrow.

**Syntax**

[objectreference].MoveDown(Count)

**Parameters**

*Count*

An Integer expression which specifies the number of times the insertion point is moved down.

**Return value****Usage**

## **Word Pro: MoveItem method**

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVEITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves a menu item from one place to another location on the same or different parent menu item.

### **Syntax**

[objectreference].MoveItem(FromItem, [After,][TargetText,][Caption])

### **Parameters**

#### *FromItem*

Specifies the menu item you want to move.

#### *After*

The default of True places the moved item after the last item in the parent menu item object. Setting the value of After to False places the moved item before the first item in the parent menu item object. Optional Boolean expression. A Boolean expression is either True or False.

#### *TargetText*

An optional String expression that allows you to specify any menu item object and place the moved item before or after it.

#### *Caption*

The name of the copied menu item that displays on the menu. You can use this optional String parameter to change the caption of a copied menu item.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Call this method to move menu items from one menu item parent object to another, or to reorder menu items within a single menu item parent object.

## **Word Pro: MoveParagraph method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVEPARAGRAPH\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the paragraph in up or down.

### **Syntax**

[objectreference].MoveParagraph(MoveDirection)

### **Parameters**

*MoveDirection*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpMoveDirectionDown (1513)

\$LwpMoveDirectionUp (1512)

### **Return value**

### **Usage**



## Word Pro: MoveToBack method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_MOVETOBACK_METHOD_EXSCRIPT',1)} See example
```

Moves a layout to the back of all its siblings.

### Syntax

```
[objectreference].MoveToBack()
```

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this method to reposition a layout object in relation to its sibling layout objects. For example, if you have a number of overlapping frames on a page, you can use this method on one of the FrameLayout objects to position it behind its sibling frames. Equivalent to changing a frame's priority setting in Word Pro. Note that in this situation, the placement of the frames must be set to "On current page" in order for the priority option to be enabled.

## **Word Pro: MoveToEnd method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVETOEND\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the selected Text or TextMarker object to the end of a specified type of location.

### **Syntax**

[objectreference].MoveToEnd(LocationType)

### **Parameters**

#### *LocationType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpLocationTypeDocument (573)

\$LwpLocationTypeLine (571)

\$LwpLocationTypeParagraph (572)

\$LwpLocationTypeSelection (568)

\$LwpLocationTypeSentence (570)

\$LwpLocationTypeStream (574)

\$LwpLocationTypeWord (569)

### **Return value**

Boolean.

### **Usage**

## Word Pro: MoveToFront method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_MOVETOFRONT_METHOD_EXSCRIPT',1)} See example
```

Moves a layout to the front of all its siblings.

### Syntax

[objectreference].MoveToFront()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Use this method to reposition a layout object in relation to its sibling layout objects. For example, if you have a number of overlapping frames on a page, you can use this method on one of the FrameLayout objects to position it in front of its sibling frames. Equivalent to changing a frame's priority setting in Word Pro. Note that in this situation, the placement of the frames must be set to "On current page" in order for the priority option to be enabled.

## Word Pro: MoveToStart method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVETOSTART\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is not available in the 16-bit version of Word Pro.

In the 32-bit version of Word Pro, MoveToStart moves the selected Text or TextMarker object to the beginning of a specified type of location.

### Syntax

[objectreference].MoveToStart(LocationType)

### Parameters

*LocationType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpLocationTypeDocument (573)

\$LwpLocationTypeLine (571)

\$LwpLocationTypeParagraph (572)

\$LwpLocationTypeSelection (568)

\$LwpLocationTypeSentence (570)

\$LwpLocationTypeStream (574)

\$LwpLocationTypeWord (569)

### Return value

Boolean.

### Usage

## **Word Pro: MoveUp method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVEUP\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].MoveUp(Count)

### **Parameters**

*Count*

A Numeric expression. You must use an integer as the Numeric expression.

### **Return value**

### **Usage**

## Word Pro: Move method

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DIVISION\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_MOVE\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is defined in the following classes:

[Division]

Allows you to move a division object.

[StatusBar]

Allows you to move the status bar.

[TextDocument]

Allows you to move a TextDocument object.

[Window]

Allows you to move a window object to the coordinates specified by the XPosition and YPosition parameters.

[Application Window]

Allows you to move the application window to the coordinates specified by the XPosition and YPosition parameters.

## Syntax

[objectreference].ActiveDocWindow.Move(Name,[ParentName,] [BeforeNeighbor,] [NeighborName]

[objectreference].Division.Move(XPosition,YPosition)

[objectreference].ActiveDocument.Move(XPosition,YPosition)

[objectreference].IconBarManager.Move(XPosition,YPosition)

[objectreference].StatusBar.Move(XPosition,YPosition)

[objectreference].ApplicationWindow.Move(XPosition,YPosition)

## Parameters

*XPosition*

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*YPosition*

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

*MoveName*

Data type is String.

*ParentName*

Data type is String. Optional parameter.

*BeforeNeighbor*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False (0).

*NeighborName*

Data type is String. Optional parameter.

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This method is not valid for IconBarManager.

## Word Pro: NewDivision method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new division in the active document.

### Syntax

[objectreference].NewDivision([ParentName,] [BeforeNeighbor,] [NeighborName,] [Initialize,][Split])

### Parameters

#### *ParentName*

A String expression which allows you to specify the internal name of the parent for the new division. Optional parameter.

#### *BeforeNeighbor*

An Integer expression which allows you to specify whether you want the new division to be placed before the neighbor or after the neighbor. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False.

#### *NeighborName*

A String expression which allows you to specify the internal name of the neighbor for the new division. Optional parameter.

#### *Initialize*

An Integer expression which allows you to copy all the styles from the neighboring division into the new division. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True, which copies all the styles from the neighboring division.

#### *Split*

An Integer expression which allows you to split the current division at the insertion point and place the new division at the split. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False, which does not split the current division.

### Return value

A String expression which represents the internal name of the external division.

For more information on division names in LotusScript, see [Overview: Division names in LotusScript](#)

### Usage

## Word Pro: NewDocument method

{button ,AL(^H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEWDOCUMENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new Word Pro document, including the DocWindow and TextDocument objects.

### Syntax

[objectreference].NewDocument([DocFileName,][DocFilePath,] [SmartMasterName,][SmartMasterDir,] [Password,] [Kind])

### Parameters

#### *DocFileName*

An optional String expression representing the name of the new file. You can leave this parameter blank if you want to open an untitled document.

#### *DocFilePath*

An optional String expression representing the path where you want to store the new file.

#### *SmartMasterName*

An optional String expression representing the name of the SmartMaster from which you want to create the new document. To create a plain document, leave this parameter empty.

#### *SmartMasterDir*

An optional String expression representing the path where the SmartMaster for the new document is stored. If the SmartMaster you want to use is stored in the default SmartMaster directory, you do not have to include this parameter.

#### *Password*

An optional String expression representing the password you want to assign the new document. Providing a value for this parameter activates the password feature for the new document. If you lose or forget the password, the document cannot be opened.

#### *Kind*

An optional String expression representing the file format for the new document.

### Return value

Long.

### Usage

Use this method to create a new document.



## Word Pro: NewFrame method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWFRAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a new frame into the active Word Pro document.

### Syntax

[objectreference].InsertFrame(Width, Height, X, Y[, FrameStyle])

### Parameters

#### *Width*

The width of the new frame expressed in Twips. Use a null string ("" ) if you want Word Pro to use the width specified in the frame style.

#### *Height*

The height of the new frame expressed in Twips. Use a null string ("" ) if you want Word Pro to use the height specified in the frame style.

#### *X*

The position, expressed in Twips, of the upper left corner of the frame on the X axis. Use a null string ("" ) if you want Word Pro to use the X coordinate specified in the frame style.

#### *Y*

The position, expressed in Twips, of the upper left corner of the frame on the Y axis. Use a null string ("" ) if you want Word Pro to use the Y coordinate specified in the frame style.

#### *FrameStyle*

A String, expressing the name of the style from which the new frame should be created. If you do not specify a frame style, Word Pro uses the default frame style.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

If you use the recorder to get the code for creating a new frame, the recorder uses NewFrame when you use the new frame icon and CreateFrame when you choose Create - Frame.

## Word Pro: NewItem method

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds a new menu item object to a named menu item object.

### Syntax

[objectreference].NewItem(Caption, [p2,] [After,] [TargetText])

### Parameters

#### *Caption*

The name of the new menu item; a String expression.

#### *p2*

An optional Variant parameter that initializes the Action property. Default is 0.

#### *prexAfter*

An optional Boolean expression that determines where the new item is inserted with the existing items in the parent menu item object. Default of True places the new item after last item in the parent menu item object. If the TargetText property is specified, the new item is inserted after the specified object. Setting the value of After to False places the new item before the first item in the parent menu item object or TargetText property.

#### *prexTargetText*

An optional String expression that allows you to specify a menu item object so you can place the new item before or after it. If you are trying to place your new item before or after an existing Word Pro item that displays an accelerator keystroke on the menu, you must build the TargetText string. Use the LotusScript function, Chr\$(8), with the text on the menu to attach the accelerator keystroke to an existing item. For example, to attach to the Save menu item accelerator keystroke (in the File menu) you can use:

```
MyTargetText$ = "&Save" & Chr$(8) & "Ctrl+S"
```

### Return value

MenuItem.

### Usage

This method is used to add new options to Word Pro menus. Each time the NewItem method is executed, a new menu item object is created and added to the Items property for the existing menu item.

## **Word Pro: NewWindow method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEWWINDOW\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens another window for the currently active document. The new window is not a copy or separate version of the document. It is the same document displayed in a different window. Changes made to the document in one window are immediately reflected in the other window(s). Equivalent to choosing Window - New Window.

### **Syntax**

[objectreference].NewWindow()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: NextItem method

{button ,AL(`H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_NEXTITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the current item to the next item.

### Syntax

[objectreference].NextItem(ObjectType, LayoutObjectType, [MarkerName])

### Parameters

#### *ObjectType*

Data type is Variant. The value of this parameter must be "\$LwpRevisionObjectTypeRevision" or its numeric equivalent (1713).

#### *LayoutObjectType*

Data type is Variant. The value of this parameter must be "\$LwpLayoutObjectTypeLayout" or its numeric equivalent (523).

#### *MarkerName*

An optional String expression that indicates the name of the marker of the next item.

### Return value

This method returns an Integer value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

### **Word Pro: NextToObject method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NEXTTOOBJECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Determines whether or not the insertion point is located next to an object of the specified type.

### **Syntax**

[objectreference].NextToObject(ObjectType)

### **Parameters**

ObjectType

### **Return value**

### **Usage**

**Word Pro: NextVersion method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NEXTVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the active version to the next version.

**Syntax**

[objectreference].NextVersion()

**Parameters****Return value****Usage**

## Word Pro: Next method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_NEXT_METHOD_EXSCRIPT',1)} See example
```

[Layout]

ClickHere, Text and TextMarker

Returns the next specified object type in a text or text marker object.

## Syntax

```
[objectreference.]Next(ObjectType, LayoutObjectType)
```

```
[objectreference.]Next(ObjectType, [SearchObjectType],[p3],[MarkerName.][ClassName.][SubClass.])
```

```
[objectreference.]Next(ObjectType, [SearchObjectType],[p3],[MarkerName.][ClassName.][SubClass.])
```

## Parameters

[Layout]

*ObjectType*

The type of object to be returned next. Data type is Variant. The value of this parameter must be "\$LwpRevisionObjectTypeRevision" or its code equivalent (1713).

[Layout]

*LayoutObjectType*

The type of layout object that contains the object type to be returned next. Data type is Variant. The value of this parameter must be "\$LwpLayoutObjectTypeLayout" or its code equivalent (523).

[Text, TextMarker, or ClickHere]

*ObjectType*

Used with a Text, TextMarker, or ClickHere object, this parameter must have one of the following values:

\$LwpNextObjectTypeFormatcheck (1529)

\$LwpNextObjectTypeRevision (1527)

\$LwpNextObjectTypeSearch (1526)

\$LwpNextObjectTypeTombstone (1528)

[Text, TextMarker, or ClickHere]

*SearchObjectType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpSelectObjectTypeBullet (1757)

\$LwpSelectObjectTypeChunk (1750)

\$LwpSelectObjectTypeDocument (1754)

\$LwpSelectObjectTypeLevel (1756)

\$LwpSelectObjectTypeObject (1748)

\$LwpSelectObjectTypeParagraph (1753)

\$LwpSelectObjectTypeSection (1759)

\$LwpSelectObjectTypeSentence (1752)

\$LwpSelectObjectTypeStream (1755)

\$LwpSelectObjectTypeTombstoneset (1758)

\$LwpSelectObjectTypeTuna (1751)

\$LwpSelectObjectTypeWord (1749)

*p3*

Data type is Variant. Optional parameter. Default is True.

*MarkerName*

Data type is String. Optional parameter.

*ClassName*

Data type is String. Optional parameter.

*SubClass*

Data type is String. Optional parameter.

### **Return value**

Integer

[Layout]

Returns one the String values below or its code equivalent.

\$LwpRevisionSelectTypeNone (1714)

\$LwpRevisionSelectTypeInsert (1715)

\$LwpRevisionSelectTypeDelete (1716)

\$LwpRevisionSelectTypeDontcare (1717)

### **Usage**



## **Word Pro: NormalText method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NORMALTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes local attributes from text in a document. This reverts the text attributes back to the text attributes defined in the paragraph style. Equivalent to choosing Text - Normal.

### **Syntax**

[objectreference].NormalText()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

### **Word Pro: NumberOfVersions method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMBEROFVERSIONS\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the number of versions of a document. Equivalent to choosing File - Versions.

### **Syntax**

[objectreference].NumberOfVersions(VersionObjectType)

### **Parameters**

*VersionObjectType*

Data type is Variant. The value of this parameter must be "\$LwpVersionObjectType" or its code equivalent (1961).

### **Return value**

### **Usage**

## **Word Pro: OpenDataFile method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENDATAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens the specified Merge data file.

### **Syntax**

[objectreference].OpenDataFile(DataFilePath, DataFileType, [Password])

### **Parameters**

#### *DataFilePath*

A String expression specifying the name and file path of the data file you want to open.

#### *DataFileType*

A String expression specifying file type of the data file you want to open.

#### *Password*

A String expression specifying the password for the data file. Use this parameter only if the data file is password-protected. Optional parameter.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: OpenDocument method

{button ,AL('H\_APPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENDOCUMENT\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens a document in Word Pro.

### Syntax

[objectreference].OpenDocument(DocName, [Location,] [Kind,] [Password,] [OpenRO,] [MakeVisible])

### Parameters

#### *DocName*

A String expression representing the path and name of the document you want to open.

#### *Location*

An optional String expression representing the path of the document you are opening. If the document is not in the current working directory, you must use this parameter.

#### *Kind*

An optional String expression representing the file format for the document you are opening. You must provide the file format if the document you are opening is not a Word Pro document.

#### *Password*

An optional String expression which provides the password for files which are password-protected.

#### *OpenRO*

An optional Boolean expression indicating whether you want to open the file as Read-write (False) or Read-only (True). Default is False.

#### *MakeVisible*

An optional Boolean expression indicating whether or not the document will be visible when opened. Optional parameter. Default is True.

### Return value

Long.

### Usage

Use this method to open an existing document in Word Pro.

## **Word Pro: OpenFromStorage method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENFROMSTORAGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens an embedded OLE Word Pro document from a specified IStorage.

### **Syntax**

[objectreference].OpenFromStorage(pIStorage, FileType, [Password])

### **Parameters**

#### *pIStorage*

A Numeric expression which specifies the IStorage space in which the embedded Word Pro object is stored. Data type is Long.

#### *FileType*

A String expression indicating the type of Word Pro or Ami Pro object being opened. The file types include:

Lotus Ami Pro

Lotus Ami Pro 3.x Macro

Lotus Ami Pro 3.x Styles

Lotus Word Pro

Lotus Word Pro SmartMaster

#### *Password*

A String expression which specifies the password for the embedded Word Pro document. Required only if the embedded Word Pro document is password protected.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: OpenObject method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENOBJECT\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens an object in a selected frame.

**Syntax**

[objectreference].OpenObject()

**Parameters****Return value****Usage**

## Word Pro: Open method

{button ,AL(^H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_NOTELAYOUT\_CLASS;H\_OLEOBJECT\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_OPEN\_METHOD\_EXSCRIPT',1)} [See example](#)

[StatusBar]

Opens/displays the status bar.

[ApplicationWindow]

Opens the application window and brings it to the top.

[NoteLayout]

Opens a specific note layout object.

## Syntax

[objectreference].NoteLayout.Open()

[objectreference].OleObject.Open(Verb)

[objectreference].Window.Open()

[objectreference].IconBarManager.Open()

[objectreference].StatusBar.Open()

[objectreference].ApplicationWindow.Open()

## Parameters

*Verb*

Data type is Long. Optional parameter. Default is 0.

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This method is not valid for IconBarManager.

## **Word Pro: OutlineMoveTextDown method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_OUTLINEMOVETEXTDOWN\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the currently active paragraph down one paragraph. The paragraph remains at the same outline level.

### **Syntax**

[objectreference].OutlineMoveTextDown()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**



## **Word Pro: OutlineMoveTextUp method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_OUTLINEMOVETEXTUP\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the currently active paragraph up one paragraph. The paragraph remains at the same outline level.

### **Syntax**

[objectreference].OutlineMoveTextUp()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: OutlineStyleSequence method**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINESTYLESEQUENCE\_METHOD\_EXSCRIPT',1)} [See example](#)

Applies paragraph styles to outline sequences.

**Syntax**

[objectreference].OutlineStyleSequence(ParagraphGetType)

**Parameters**

*ParagraphGetType*

Data type is Variant. The value of this parameter must be "\$LwpParagraphGetType" or its code equivalent (1628).

**Return value****Usage**

## **Word Pro: PageDown method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_PAGEDOWN\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point down one page.

### **Syntax**

[objectreference].PageDown()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: PageUp method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_PAGEUP\_METHOD\_EXSCRIPT',1)} [See example](#)

Moves the insertion point up one page.

### **Syntax**

[objectreference].PageUp()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: PasteLink method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PASTELINK\_METHOD\_EXSCRIPT',1)} [See example](#)

Pastes the contents of the Clipboard into the active Word Pro document as a linked OLE object.

### Syntax

[objectreference].PasteLink(PresentationFormat, [UpdateDataOnly,] [IconMetaFilePictHandle])

### Parameters

#### *PresentationFormat*

A String expression that indicates the Clipboard format you want to use for presenting the linked object in your Word Pro document. In most circumstances you should use "LWPOLEFormat". However, there may be times when you want to use one of the other formats to ensure that certain features will be available to you. For example, if you are pasting spreadsheet data into a Word Pro table and you want to update data without changing the cell formatting, you must use "Rich Text Format" for this parameter. Presentation formats include the following:

LWPOLEFormat	The normal setting for pasted links.
CF_METAFILEPICT	Windows Metafile
CF_BITMAP	Windows Bitmap
CF_DIB	Device Independent Bitmap
CF_TEXT	Text
Rich Text Format	Rich Text Format

#### *UpdateDataOnly*

An optional parameter that indicates whether or not you want the cell format of a linked Word Pro table to be updated along with the data in the cells. This option is only available for spreadsheet data which is pasted into a Word Pro table and linked through OLE to the original spreadsheet. A value of True (-1) will cause OLE to update only the data in the table cells. Any local formatting done to the Word Pro table will remain intact. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. Optional parameter. Default is False (0).

**Note** If you use a value of True for this parameter, you must use "Rich Text Format" as the value of the PresentationFormat parameter.

#### *IconMetaFilePictHandle*

An optional Numeric expression which allows you to specify which icon to use in representing the linked object in the Word Pro file. Using any value other than 0 automatically tells Word Pro to display the linked object as an icon. This value serves as a numeric handle (known as the HGLOBAL) to the metafile pict for that icon. You can get the HGLOBAL for a specific metafile pict by using the appropriate Windows API calls. Data type must be Long. Default is 0 which indicates that you want the contents of the linked OLE object to be displayed in the Word Pro document.

**Caution** If you record the process of choosing Edit - Paste Special and linking a file to be displayed as icon, Word Pro records a value for IconMetaFilePict that is valid only during the recording. When you play back the recorded script, the IconMetaFilePict value will be invalid and Word Pro will treat the value as if you passed a 0.

### Return value

### Usage

## Word Pro: PasteSpecial method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PASTESPECIAL\_METHOD\_EXSCRIPT',1)} [See example](#)

Pastes an OLE object into a Word Pro document. Equivalent to choosing Edit - Paste Special from the object's source application to the Word Pro application.

### Syntax

[objectreference].PasteSpecial(Format, IconMetaFilePictHandle)

### Parameters

#### *Format*

A String expression indicating the format of the OLE object.

<u>Value</u>	<u>Source of OLE object</u>
CF_TEXT	ASCII text
Rich Text Format	Rich text format
Biff	MS Excel versions earlier than 3
Biff3	MS Excel 3
Biff4	MS Excel 4
LotusChart	LotusChart
CF_METAFILEPICT	Windows Metafile
CF_DIB	Device independent bitmap
CF_BITMAP	Bitmap
WordProDraw	WordProDraw
WordProEquation	WordProEquation
WordProNative	Internal paste from AppFoundry
FileName	OLE file
Embed Source	Creates an OLE embedded object from any object that can become an OLE embedded object.
Embedded Object	Creates an OLE embedded object from an existing OLE embedded object.
Link Source	Creates a linked OLE object.
Link	Creates a DDE link.

#### *IconMetaFilePictHandle*

A Numeric expression which allows you to specify which icon to use in representing the new OLE object. Optional parameter. Using any value other than 0 automatically tells Word Pro to display the new object as an icon. This value serves as a numeric handle (known as the HGLOBAL) to the metafile pict for that icon. You can get the HGLOBAL for a specific metafile pict by using the appropriate Windows API calls. Data type must be Long. Default is 0, which indicates that you want Word Pro to display the contents of the new OLE object.

**Caution** If you record the process of creating a new OLE object to be displayed as icon, Word Pro records a value for IconMetaFilePict, which is valid only during the recording. When you play back the recorded script, the IconMetaFilePict value will be invalid and Word Pro will treat the value as if you passed 0.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: Paste method

{button ,AL('H\_DOCUMENT\_CLASS;H\_FOUNDRY\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PASTE\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is defined in the following classes:

[WPAApplication]

Pastes the contents of the Clipboard into the currently active document at the insertion point.

[Foundry]

Pastes all the object(s) contained in the specified Foundry object into the Foundry object from which the Paste method is called.

## Syntax

[objectreference].WPAApplication.Paste([MakeVisible])

[objectreference].Document.Paste([MakeVisible])

[objectreference].Foundry.Paste([FoundryType,][Clear])

## Parameters

[Document, WPAApplication]

*MakeVisible*

Do not use this parameter. This parameter is not implemented in this release of Word Pro.

*FoundryType*

Used when calling this method from a Foundry object, this parameter specifies which Foundry object you want to paste from. Data type is Variant, which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). Default is \$LwpFoundryTypeDocument.

\$LwpFoundryTypeApplication (346)

Indicates the Foundry object found in the AppFoundry property on WPAApplication. Pastes all objects currently in WordPro.AppFoundry into the Foundry object from which the script calls the Paste method. For example, if you use this value to call this method from a division called "Summary," all the objects in AppFoundry will be pasted into the Summary division's Foundry.

\$LwpFoundryTypeDocument (345)

Indicates the same Foundry object from which the script calls the Paste method. For example, if you use this value to call this method from a division called "Summary," all the objects in the Summary division's Foundry will be pasted into the Summary division's Foundry a second time, thus creating duplicates of each object in the Summary division's Foundry object.

\$LwpFoundryTypeTemporary (347)

Indicates the Foundry object found in the TempFoundry property on WPAApplication. Pastes all objects currently in WordPro.TempFoundry into the Foundry object from which the script calls the Paste method. For example, if you use this value to call this method from a division called "Summary," all the objects in TempFoundry will be pasted into the Summary division's Foundry.

*Clear*

Used when calling this method from a Foundry object, this parameter clears the source Foundry of the pasted object(s). For example, if FoundryType has a value of \$LwpFoundryTypeApplication (or 346), setting this parameter to False (0) would leave all the AppFoundry objects in place after the Paste method is executed. A value of True (-1) would cause all the objects in AppFoundry to be cleared after the Paste method was executed. Data type is Integer but the legal values are True (-1) and False (0). Default is True (-1).

## Return value

When used with WPAApplication and Document, this method returns a value of -1 (True) or 0 (False), indicating that the method succeeded or failed respectively.

Foundry objects return a value of type Variant.

## Usage

**Word Pro: PColConnectCells method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PCOLCONNECTCELLS\_METHOD\_EXSCRIPT',1)} [See example](#)

Connects the selected parallel column table cells.

**Syntax**

[objectreference].PColConnectCells()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**



## **Word Pro: PColConnectRows method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PCOLCONNECTROWS\_METHOD\_EXSCRIPT',1)} [See example](#)

Connects the cells in the parallel column row which has the focus. If cells from more than one row are selected, Word Pro connects the cells in the row which has the focus, and then moves the contents of the selected rows into the connected row. Word Pro leaves the rows which did not have the focus as disconnected rows of empty cells.

Equivalent to choosing Columns - Connect Across Row.

### **Syntax**

[objectreference].PColConnectRows()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: PColDisconnectCells method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PCOLDISCONNECTCELLS\_METHOD\_EXSCRIPT',1)} [See example](#)

Disconnects the selected parallel column table cells. Only works if there are connected cells or rows within the selection. Equivalent to choosing Columns -Disconnect Column Block.

**Syntax**

[objectreference].PColDisconnectCells()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

If you disconnect parallel column cells, the contents of the connected cell remain in a single cell rather than returning to their original separate cells.

## **Word Pro: PColSelectColumn method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PCOLSELECTCOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the parallel column which contains the cell that currently has the focus. If cells from more than one parallel column are in the selection, Word Pro selects all the parallel columns with cells in the selection. The insertion point must be in a parallel column table. Equivalent to choosing Columns - Select - Column Contents.

### **Syntax**

[objectreference].PColSelectColumn()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: PColSelectRow method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PCOLSELECTROW\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the parallel column row which contains the cell that currently has the focus. If cells from more than one row are in the selection, Word Pro selects all the rows with cells in the selection. The insertion point must be in a parallel column table. Equivalent to choosing Columns - Select - Row Contents.

**Syntax**

[objectreference].PColSelectRow()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: PColSelectTable method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PCOLSELECTTABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the parallel column table which currently has the focus. The focus must be include a parallel column table cell. Equivalent to choosing Table - Select - All Column Contents.

**Syntax**

[objectreference].PColSelectTable()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## Word Pro: Play method

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PLAY\_METHOD\_EXSCRIPT',1)} [See example](#)

[ApplicationWindow]

Determines what script to play and when to play it..

### Syntax

[objectreference].Play(p1)

### Parameters

*p1*

Data type is Variant. Optional parameter. This parameter has two parts:

1. String - Indicates the script filename (.LSS or .LWP).
2. Enumerated list - Indicates when the script will play.

\$LwpPlayWhenWordprolsExecuted = 1631

\$LwpPlayWhenDoslsOpened = 1632

\$LwpPlayWhenDoslsClosed = 1633

\$LwpPlayWhenNewdoc = 1634

\$LwpPlayWhenWordprolsExited = 1635

If this method contains no parameters, the current script will play when the method is called.

### Return value

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

Use this method to play a script.

## **Word Pro: PrepareToDestroy method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREPARETODESTROY\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].PrepareToDestroy()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: Present method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRESENT\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Present()

### **Parameters**

### **Return value**

### **Usage**



## Word Pro: PreviousItem method

```
{button ,AL('H_BASERTABLE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNOTETABLE_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GLOSSARY_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARALLEL_COLUMNS_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_PREVIOUSITEM_METHOD_EXSCRIPT',1)} See example
```

This method is defined in the following classes:

[Layout]

Internal Use Only.

[BaseTable]

Returns the last selected item in a glossary or base table.

## Syntax

```
[objectreference].BaseTable.PreviousItem(RevisionObjectType,LayoutObjectTyp
```

## Parameters

### *ObjectType*

The type of previously returned type object. Data type is Variant. The value of this parameter must be \$LwpRevisionObjectTypeRevision or its code equivalent (1713).

### *LayoutObjectType*

The type of layout object that contains the previously returned object type. Data type is Variant. The value of this parameter must be \$LwpLayoutObjectTypeLayout or its code equivalent (523).

### *MarkerName*

An optional String expression that indicates the name of the marker of the next item.

## Return value

This method returns an Integer value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

## Usage

## **Word Pro: PreviousVersion method**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREVIOUSVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].PreviousVersion()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: Previous method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREVIOUS\_METHOD\_EXSCRIPT',1)} [See example](#)

### Syntax

[objectreference].Previous(ObjectType,[SearchObjectType],[ p3,] [MarkerName,] [ClassName,] [SubClass])

### Parameters

#### *ObjectType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpNextObjectTypeFormatcheck (1529)

\$LwpNextObjectTypeRevision (1527)

\$LwpNextObjectTypeSearch (1526)

\$LwpNextObjectTypeTombstone (1528)

#### *SearchObjectType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent. Default is \$LwpSearchObjectTypeObject.

\$LwpSearchObjectTypeAlignment (1739)

\$LwpSearchObjectTypeIndent (1740)

\$LwpSearchObjectTypeObject (1736)

\$LwpSearchObjectTypeRevision (1743)

\$LwpSearchObjectTypeSpacing (1741)

\$LwpSearchObjectTypeStyle (1738)

\$LwpSearchObjectTypeTab (1742)

\$LwpSearchObjectTypeText (1737)

#### *p3*

Data type is Variant. Optional parameter. Default is True.

#### *MarkerName*

Data type is String. Optional parameter.

#### *ClassName*

Data type is String. Optional parameter.

#### *SubClass*

Data type is String. Optional parameter.

### Return value

### Usage

## Word Pro: PrintOut method

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTOUT\_METHOD\_EXSCRIPT',1)} [See example](#)

Prints a Word Pro document. This method is defined in several classes. See Usage for details on how this method behaves in each class.

**Note** This method is the same as the [Print](#) method defined in WPAApplication and TextDocument. We provide this alternative because the word "Print" is a reserved word in OLE automation. This reserved status prevents the Print method from working. If you are developing a script for use in OLE automation, you must use the PrintOut method instead of the Print method.

## Syntax

[objectreference].Print([From,][To,][Copies,][nodialog])

## Parameters

### *From*

An Integer specifying the first page to be printed. Optional parameter. Default is 1.

### *To*

An Integer specifying the last page to be printed. Optional parameter. Default is 9999.

### *Copies*

An Integer specifying the number of copies to be printed. Optional parameter. Default is 1.

### *nodialog*

Allows you to suppress or display the Print dialog box, which normally appears when you choose File - Print. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True, which suppresses the Print dialog box.

## Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

## Usage

The default values for this method print one copy of the entire document without displaying the Print dialog box.

## Word Pro: Print method

{button ,AL('H\_DOCUMENT\_CLASS;H\_PRINTMANAGER\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINT\_METHOD\_EXSCRIPT',1)} [See example](#)

Prints a Word Pro document. This method is defined in several classes. See Usage for details on how this method behaves in each class.

**Note** The word "Print" is a reserved word in OLE automation. This reserved status prevents the Print method from working. If you are developing a script for use in OLE automation, you must use the [PrintOut](#) method instead of the Print method.

### Syntax

[objectreference].WPAApplication.Print([From,][To,][Copies,][nodialog])

[objectreference].Document.Print([From,][To,][Copies,][nodialog])

[objectreference].PrintManager.Print(DocName)

### Parameters

#### *From*

An Integer specifying the first page to be printed. Optional parameter. Default is 1. This parameter is used only when using Print from WPAApplication or Document.

#### *To*

An Integer specifying the last page to be printed. Optional parameter. Default value is 9999. This parameter is used only when using Print from WPAApplication or Document.

#### *Copies*

An Integer specifying the number of copies to be printed. Optional parameter. Default is 1. This parameter is used only when using Print from WPAApplication or Document.

#### *nodialog*

Allows you to suppress or display the Print dialog box, which normally appears when you choose File - Print. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is True, which suppresses the Print dialog box. This parameter is used only when using Print from WPAApplication or Document.

#### *DocName*

A String expression specifying the name of the document you want to print. Use this parameter when using Print from PrintManager.

### Return value

When called from WPAApplication and TextDocument objects, this method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed, respectively.

PrintManager returns an Integer.

### Usage

When called from WPAApplication or a TextDocument, you can take the default values for the parameters that will print one copy of the entire document without displaying the Print dialog box.

When called from the PrintManager object, the Print method requires only the DocName parameter.

## **Word Pro: ProcessAccelKey method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PROCESSACCELKEY\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].ProcessAccelKey(WParam)

### **Parameters**

*WParam*

Data type is integer.

### **Return value**

Integer

### **Usage**

**Word Pro: PromoteOutlineLevel method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PROMOTEOUTLINELEVEL\_METHOD\_EXSCRIPT',1)} [See example](#)

Promotes the current paragraph to the next higher outline level. Equivalent to choosing Text - Outline - Promote.

**Syntax**

[objectreference].PromoteOutlineLevel()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: Promote method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PROMOTE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Promote()

### **Parameters**

### **Return value**

### **Usage**



**Word Pro: Purge method**

{button ,AL('H\_DIVISION\_CLASS;H\_FOUNDRY\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PURGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Gets rid of any objects in a Foundry object which were derived from a SmartMaster. Use this when you want to switch from one SmartMaster to another and get rid of the old objects created by the original SmartMaster.

**Syntax**

[objectreference].Purge()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: QueryDrop method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_QUERYDROP\_METHOD\_EXSCRIPT',1)} [See example](#)

Used during drag and drop operations, this method queries the currently active document window to determine if the items being dragged can be dropped at the location of the mouse cursor.

### Syntax

[objectreference].QueryDrop(WindowId,GeneralModifier, SpecificModifier, X, Y)

### Parameters

#### *WindowId*

An Integer which specifies the document window you are querying about the drop. Data type is Long.

#### *GeneralModifier*

Indicates which modifier keys are pressed during the drag operation. Data type is Variant. This allows this parameter to accommodate one of the string values below or its hexadecimal equivalent (in parentheses).

LwpGeneralModifierAlt (&H2) Indicates the ALT key is pressed.

LwpGeneralModifierCapslock (&H40) Indicates the CAPS LOCK key is pressed.

LwpGeneralModifierCommand (&H8) Indicates the COMMAND key is pressed. (Macintosh only)

LwpGeneralModifierCtrl (&H4) Indicates the CTRL key is pressed.

LwpGeneralModifierHelp (&H200) Indicates the HELP key is pressed.

LwpGeneralModifierNone (&H0) Indicates that no modifier keys are pressed.

LwpGeneralModifierNumlock (&H80) Indicates the NUMLOCK key is pressed.

LwpGeneralModifierOption (&H10) Indicates the OPTION key is pressed. (Macintosh only)

LwpGeneralModifierScrolllock (&H100) Indicates the SCROLL LOCK key is pressed.

LwpGeneralModifierShift (&H1) Indicates the SHIFT key is pressed.

LwpGeneralModifierSys (&H20) Indicates the SYSRQ key is pressed.

#### *SpecificModifier*

The value of this parameter should be zero (0). Data type is Variant. This allows this parameter to accommodate one of the string values below or its hexadecimal equivalent (in parentheses).

LwpSpecificModifierClicked (&H2)

LwpSpecificModifierComingup (&H4)

LwpSpecificModifierDamnait (&H1)

LwpSpecificModifierDoublebyte (&H1)

LwpSpecificModifierGoingdown (&H2)

LwpSpecificModifierMbutton1 (&H1)

LwpSpecificModifierMbutton2 (&H2)

LwpSpecificModifierMbutton3 (&H4)

LwpSpecificModifierMbuttondown (&H8)

LwpSpecificModifierMbuttonmask (&H7)

LwpSpecificModifierMbuttonup (&H10)

LwpSpecificModifierNomove (&H4)

LwpSpecificModifierNone (&H0)

LwpSpecificModifierNovolatile (&H8)

LwpSpecificModifierReset (&H10)

LwpSpecificModifierVirtual (&H8)

One of two coordinates indicating the position of the upper left corner of the window being queried. Data type is Long. Position of the X coordinate is expressed in Twips.

Y

One of two coordinates indicating the position of the upper left corner of the window being queried. Data type is Long. Position of the Y coordinate is expressed in Twips.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the dragged items can or cannot be dropped respectively.

**Usage**

**Word Pro: Query method**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_QUERY\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays a Windows Message box with a specified prompt and an edit box where you can enter string data to be sent back to the macro.

**Syntax**

[objectreference].Query(??)

**Parameters****Return value****Usage**

## Word Pro: QuickAlignFrame method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_QUICKALIGNFRAME\_METHOD\_EXSCRIPT',1)} [See example](#)

Aligns the currently active frame relative to the center or edges of the page on which the frame is located.

### Syntax

[objectreference].QuickAlignFrame(QuickLayoutAlign)

### Parameters

#### *QuickLayoutAlign*

Specifies how you want to align this frame in relation to the page. Data type is Variant, which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpQuickLayoutAlignLeft (1657) Aligns the left edge of the frame with the left edge of the page. This does not affect the vertical position of the frame.

\$LwpQuickLayoutAlignRight (1658) Aligns the right edge of the frame with the right edge of the page. This does not affect the vertical position of the frame.

\$LwpQuickLayoutAlignVertcenter (1659) Aligns the center point of the frame with the vertical center of the page. This does not affect the horizontal position of the frame.

\$LwpQuickLayoutAlignHorzcenter (1660) Aligns the center point of the frame with the horizontal center of the page. This does not affect the vertical position of the frame.

\$LwpQuickLayoutAlignTop (1661) Aligns the top of the frame with the bottom edge of the page. This does not affect the horizontal position of the frame.

\$LwpQuickLayoutAlignBottom (1662) Aligns the bottom of the frame with the bottom edge of the page. This does not affect the horizontal position of the frame.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

To center a frame on a page, call this method twice, once with the value, \$LwpQuickLayoutAlignVertcenter, and a second time with the value, \$LwpQuickLayoutAlignHorzcenter.

## Word Pro: QuickAlignTable method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_QUICKALIGNTABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Aligns the currently active table relative to the center or edges of the page(s) on which the table is located.

### Syntax

[objectreference].QuickAlignTable(QuickLayoutAlign)

### Parameters

#### *QuickLayoutAlign*

Specifies how you want to align this table in relation to the page. Data type is Variant, which allows the value of this parameter to be one of the string values listed below or its numeric equivalent (in parentheses). There is no default value.

\$LwpQuickLayoutAlignLeft (1657) Aligns the left edge of the table with the left edge of the page. This does not affect the vertical position of the table.

\$LwpQuickLayoutAlignRight (1658) Aligns the right edge of the table with the right edge of the page. This does not affect the vertical position of the table.

\$LwpQuickLayoutAlignVertcenter (1659) Aligns the center point of the table with the vertical center of the page. This does not affect the horizontal position of the table.

\$LwpQuickLayoutAlignHorzcenter (1660) Aligns the center point of the table with the horizontal center of the page. This does not affect the vertical position of the table.

\$LwpQuickLayoutAlignTop (1661) Aligns the top of the table with the bottom edge of the page. This does not affect the horizontal position of the table.

\$LwpQuickLayoutAlignBottom (1662) Aligns the bottom of the table with the bottom edge of the page. This does not affect the horizontal position of the table.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

To center a table on a page, call this method twice, once with the value, \$LwpQuickLayoutAlignVertcenter, and a second time with the value, \$LwpQuickLayoutAlignHorzcenter.



## **Word Pro: Read method**

{button ,AL('H\_BAG\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_READ\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Read(Length)

### **Parameters**

*Length*

Data type is Long.

### **Return value**

### **Usage**



## **Word Pro: Redo method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REDO\_METHOD\_EXSCRIPT',1)} [See example](#)

Reverses an action which has been undone using the Undo command. Equivalent to choosing Edit - Undo/Redo Special and using the Redo feature.

### **Syntax**

[Objectreference].WPApplication.Redo()

[Objectreference].TextDocument.Redo(Count)

### **Parameters**

#### *Count*

Used only when calling this method from a TextDocument object. The value of this parameter must be an Integer which specifies the number of undone actions you want to redo. Word Pro can only redo as many actions as have been undone. If this number is greater than the number of undone actions, Word Pro will redo all the undone actions and then stop. Optional parameter. Default is 1.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

Use this method any time you want to redo an action listed in the "Edits you can redo" list in the Undo/Redo dialog box.

## **Word Pro: Refresh method**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REFRESH\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Refresh()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: RegisterWPDataSet method

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_CLICK  
HERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H  
_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYO  
UT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HE  
ADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT  
CLASS;H_PARAGRAPHSTYLE_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLA  
YOUT_CLASS;H_RUBYLAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS  
;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLE  
MARKER_CLASS;H_TEXT_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTA  
BLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL(^H_REGISTERWPDATASET_METHOD_EXSCRIPT',1)} See example
```

Creates a WPDataSet and attaches it to the object from which you call this method.

### Syntax

```
[objectreference].RegisterWPDDataset(GroupName)
```

### Parameters

*GroupName*

A String expression representing the name given to the WPDataSet.

### Return value

Returns the newly created WPDataSet object. Assign this return value to a variable to gain easy access to the WPDataSet object.

### Usage

WPDataSet objects are useful tools that store data with a document. When you close a document that has one or more data sets attached to it, Word Pro saves the data set(s) with the document. Any time the document is open, you have access to the data sets created for that document.

When you register or unregister a WPDataSet on a Text object, that WPDataSet is assigned to the currently active paragraph.

This method creates and gives a name to a WPDataSet object. The new data set becomes attached to the object from which you call this method. This method returns the new object so you can assign it to a variable.

## Word Pro: Release method

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RELEASE\_METHOD\_EXSCRIPT',1)} [See example](#)

Gets rid of a SilverBullet or Style object in the Foundry object from which you call the Release method.

### Syntax

[objectreference].Release(FoundryReleaseType, p2, [ObjectName])

### Parameters

#### *Type*

Indicates whether you want to get rid of a SilverBullet or an object created from one of the following classes: CellLayout, CharacterStyle, FrameLayout, PageLayout, ParagraphStyle, TableLayout. Data type is Variant. The value of this parameter must be one of the strings below or its numeric equivalent (indicated in parentheses):

\$LwpFoundryReleaseTypeSilverbullet (344)

\$LwpFoundryReleaseTypeStyle (343)

#### *p2*

The value of this parameter depends on the value of the Type parameter. Data type is Variant. If the value of the Type parameter is \$LwpFoundryReleaseTypeSilverBullet (or 344), this parameter must be a string indicating the name of the SilverBullet object you want to release. If the value of the Type parameter is \$LwpFoundryReleaseTypeStyle (or 343), this parameter must be one of the following strings or its numeric equivalent (indicated in parentheses):

\$LwpStyleTypeCell (1834) Releases a CellLayout object.

\$LwpStyleTypeCharacter (1830) Releases a CharacterStyle object.

\$LwpStyleTypeDefault (1828)

\$LwpStyleTypeFrame (1832) Releases a FrameLayout object.

\$LwpStyleTypePage (1831) Releases a PageLayout object.

\$LwpStyleTypeParagraph (1829) Releases a ParagraphStyle object.

\$LwpStyleTypeTable (1833) Releases a TableLayout object.

#### *ObjectName*

If the value of the Type parameter is \$LwpFoundryReleaseTypeStyle (or 343), this parameter must be a string indicating the name of the object you want to release. Data type is String.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

**Word Pro: RemoveBookmark method**

{button ,AL('H\_BOOKMARKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEBOOKMARK\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes a bookmark. You must use the internal (hexidecimal) marker name for the bookmark in order to remove it. Any time this method is called, you must also remove the mark from the Markers collection located in the Foundry for the division.

**Syntax**

[objectreference].RemoveBookmark(MarkerName)

**Parameters**

*MarkerName*

Data type is String. Required parameter.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

To obtain an internal (hexidecimal) name, you can enumerate those names, or you can select a bookmark by its user name and ask for its internal name.

## Word Pro: RemoveChildFromLayout method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_REMOVECHILDFROMLAYOUT_METHOD_EXSCRIPT',1)} See example
```

Removes a specific layout object from its parent layout.

### Syntax

[objectreference].RemoveChildFromLayout()

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

Call this method from the child layout that you want to remove from its parent.

### **Word Pro: RemoveDataFile method**

{button ,AL(^H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REMOVEDATAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes a data file from a text document.

[MergeOptions]

Removes a Merge data file from the current document.

### **Syntax**

[objectreference].RemoveDataFile(MergeOptions)

### **Parameters**

### **Return value**

Long

### **Usage**

## **Word Pro: RemoveDdeLink method**

{button ,AL('H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEDDELINK\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes a Dde link from a document.

### **Syntax**

[objectreference].RemoveDdeLink(LinkInfo, [DdeNameType])

### **Parameters**

#### *LinkInfo*

Data type is String. Required parameter.

#### *DdeNameType*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpDdeNameTypeLinkinfo (176) Default.

\$LwpDdeNameTypeMarkername (175)

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: RemoveDepOnDocFile method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEDEPONDOCFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RemoveDepOnDocFile()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RemoveDivision method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes a division in a text document.

### **Syntax**

[objectreference].RemoveDivision(Name)

### **Parameters**

*Name*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: RemoveEditor method**

{button ,AL('H\_EDITORMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEEDITOR\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes an editor's name from a document.

### **Syntax**

[objectreference].RemoveEditor(EditorName)

### **Parameters**

*EditorName*

The name of the editor you want to delete. Data type is String.

### **Return value**

The return value for this method is always -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

This method returns True if Word Pro removes the editor's name from the document. This method returns False under the following conditions: the editor field is blank, an editor does not exist, the editor is the "All Others" editor, or the editor owns any edits in the document.

### **Usage**

Do not use the RemoveEditor method to delete "All Others", "Current", or "SmartMaster" editors.

## Word Pro: RemoveNamedProperty method

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYO  
UT_LAYOUT_CLASS;H_CONNECTEDLAYOOUT_CLASS;H_DIVISION_CLASS;H_DROPPLAYOOUT_CLASS;H_ENDNOTELAYO  
UT_LAYOUT_CLASS;H_FOOTERLAYOOUT_CLASS;H_FOOTNOTELAYOOUT_CLASS;H_FRAMEGROUPLAYOOUT_CLASS;H_FR  
AMELAYOOUT_CLASS;H_GROUPLAYOOUT_CLASS;H_HEADERLAYOOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_N  
OTELAYOOUT_CLASS;H_PAGELAYOOUT_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOOUT_CLASS;H_ROWLAYO  
UT_CLASS;H_RUBYLAYOOUT_CLASS;H_RBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOOUT_CLASS;H_SUPERTABLEL  
AYOOUT_CLASS;H_TABLELAYOOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_C  
LASS;H_TOCSUPERTABLELAYOOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_REMOVENAMEDPROPERTY_METHOD_EXSCRIPT',1)} See example
```

Removes the specified user-defined property from a division, layout, marker, or text document. This method only works on a WPDataSet property.

### Syntax

```
[objectreference].RemoveNamedProperty(PropertyName)
```

### Parameters

*PropertyName*

A String expression representing the name of the property you want to delete in a layout object.

### Return value

The return value for this method will always be -1.

### Usage

Since the RemoveNamedProperty method always returns -1, you must use the HasNamedProperty method to determine whether or not a specified named property actually exists. You will not receive a run-time error if you attempt to remove a named property that does not exist.

## **Word Pro: RemovePersistentAccelerators method**

{button ,AL('H\_ACCELERATORS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEPERSISTENTACCELERATORS\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RemovePersistentAccelerators()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RemovePowerField method**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEPOWERFIELD\_METHOD\_EXSCRIPT',1)} [See example](#)

Deletes a Power field from a document. Equivalent to clicking the Show/Hide Power Fields icon, selecting the Power field, and pressing DEL. Also equivalent to placing the insertion point in the Power field, clicking the right mouse button, and selecting Delete Field.

### **Syntax**

[objectreference].RemovePowerField(PowerFieldName)

### **Parameters**

*PowerFieldName*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: RemoveProperty method**

{button ,AL('H\_CHARACTERSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVEPROPERTY\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RemoveProperty(PropertyName)

### **Parameters**

*PropertyName*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: Remove method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REMOVE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Remove(RemoveType, [OthersInSet,] [DeleteParaData])

### **Parameters**

*RemoveType*

Data type is Variant.

\$LwpRemoveTypeHideshowprops (1672)

\$LwpRemoveTypeMultiParaTag (1673)

*OthersInSet*

Data type is Boolean. Optional parameter. Default is False.

*DeleteParaData*

Data type is Boolean. Optional parameter. Default is False.

### **Return value**

### **Usage**



## **Word Pro: RenderClipBitmap method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RENDERCLIPBITMAP\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RenderClipBitmap()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RenderClipDIB method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RENDERCLIPDIB\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RenderClipDIB()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RenderClipMetafile method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RENDERCLIPMETAFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RenderClipMetafile()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RenderClipPalette method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RENDERCLIPPALETTE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RenderClipPalette()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: Render method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RENDER\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Render(Format)

### **Parameters**

Format)

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: ReplaceAll method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_REPLACEALL\_METHOD\_EXSCRIPT',1)} [See example](#)

Executes a replace all, based on the settings found in the FindAndReplace object.

### **Syntax**

[objectreference].ReplaceAll()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: ReplaceCmd method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACECMD\_METHOD\_EXSCRIPT',1)} [See example](#)

Executes a replace, based on the settings found in the FindAndReplace object.

### **Syntax**

[objectreference].ReplaceCmd()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: Replace method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS',0)}

[See list of classes](#)

{button ,AL('H\_REPLACE\_METHOD\_EXSCRIPT',1)} [See example](#)

Finds and optionally replaces the first occurrence of the target text, special characters, or paragraph styles in a document..

### Syntax

[objectreference].Replace(ReplaceObjectType, ReplaceText)

### Parameters

*ReplaceObjectType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpReplaceObjectTypeCharacter (1674)

\$LwpReplaceObjectTypeChunk (1676)

\$LwpReplaceObjectTypeSentence (1677)

\$LwpReplaceObjectTypeWord (1675)

*Replace*

Text. Data type is String. On a ClickHere object, this method replaces the existing Content object with the content object (usually a Text object) named in the ReplaceText parameter.

### Return value

### Usage



## **Word Pro: RequestAndProcessData method**

{button ,AL('H\_DDELINKMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REQUESTANDPROCESSDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates a Dde link in Word Pro.

### **Syntax**

[objectreference].RequestAndProcessData(LinkInfo, [DdeSelection])

### **Parameters**

#### *LinkInfo*

Data type is String. Required parameter.

#### *DdeSelection*

Data type is Variant. Optional parameter. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpDdeSelectionDdelinkName (177)

\$LwpDdeSelectionMarkerName (178) Default.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: ResetFindAndReplace method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RESETFINDANDREPLACE\_METHOD\_EXSCRIPT',1)} [See example](#)

Resets the Find & Replace marks from the insertion point forward and allows you to continue the Find & Replace operation.

### **Syntax**

[objectreference].ResetFindAndReplace()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

If you begin a Find & Replace operation and interrupt the operation to make changes inside your document, Word Pro dismisses the Find & Replace marks and allows you to make the changes. You can then use this method to reset the Find & Replace marks and continue the operation. The operation will begin at the insertion point and continue to the end of the document. The operation then loops back to the beginning of the document and continues to the location of the insertion point.

### **Word Pro: Reset method**

{button ,AL('H\_BAG\_CLASS;H\_FINDANDREPLACE\_CLASS;H\_NUMERICFORMAT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESET\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the options back to the default setting (clear) for Find & Replace text.

### **Syntax**

[objectreference].Reset()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text, clicking Options and the Font button in either the "Find options" or "Replace options" section. If you choose a font name, attributes, style, or color, and then want to start again, click Clear to reset.

## **Word Pro: Resize method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

[ApplicationWindow]

Resizes the application window at the current position to the width and height specified by the parameters.

### **Syntax**

[objectreference].IconBarManager.Resize(Width, Height)

[objectreference].StatusBar.Resize(Width, Height)

[objectreference].ApplicationWindow(Width, Height)

### **Parameters**

#### *Width*

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

#### *Height*

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This method is not valid for IconBarManager and StatusBar.

[ApplicationWindow]

Use this method to change the size of the application window.

## **Word Pro: Restore method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESTORE\_METHOD\_EXSCRIPT',1)} [See example](#)

Restores the Word Pro application window to its previous size.

### **Syntax**

[objectreference].Restore()

### **Parameters**

None

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This method corresponds to the Restore button in the upper righthand corner of the Word Pro application window.

## **Word Pro: RetrieveInternetFileAndOpen method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RETRIEVEINTERNETFILEANDOPEN\_METHOD\_EXSCRIPT',1)} [See example](#)

Gets a file via WWW or FTP, stores it in your temporary directory, and opens the file for viewing in Word Pro. Equivalent to choosing File - Open and clicking Internet. The parameters for this method correspond to the options available in the Open from Internet dialog box.

### **Syntax**

[objectreference].RetrieveInternetFileAndOpen()

### **Parameters**

#### *URL*

A String expression specifying the URL for the document you want to open.

#### *UserID*

Used when retrieving a file from an FTP site, this parameter takes a String value representing the name of the user who has an account with the FTP server. If you are retrieving a WWW document, the value of this parameter should be a null string ("").

#### *Password*

Used when retrieving a file from an FTP site, this parameter takes a String value representing the password for the user named in UserID. If you are retrieving a WWW document, the value of this parameter should be a null string ("").

#### *Passive*

Set this value to True when you want to initiate the file transfer. Set it to False to allow the server to repond to your request when it is ready. Some FTP servers do not support this feature. The value of this parameter is usually False. If you are retrieving a WWW document, the value of this parameter should be a null string (""). Data type is Integer, but the value is always 0 (False) or -1 (True). You can use the LotusScript constants of True and False.

#### *Proxy*

A String expression specifying the DNS (for example, screen.companyname.com) or IP address (for example, 123.456.78.912). Do not include the "http:\\" in front of the the proxy value.

#### *ProxyPort*

An Integer which specifies the port number for the Proxy server. The value of this parameter is usually 8080 for the WWW and 21 for FTP, but you should check with your Internet Service Provider for your settings.

### **Return value**

A String representing the name of the file you retrieved and opened.

### **Usage**

You must have installed the Word Pro HTML filter to access most WWW documents with this method. This method will not work unless your machine is configured for Internet access. A standard Internet access configuration includes a WINSOCK compliant DLL.

The retrieved file will be stored in the directory you marked as your temporary directory. The temporary directory is identified in an environment variable, but most temporary directories are named "TEMP."

## **Word Pro: RetrievalInternetFile method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RETRIEVEINTERNETFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Gets a file via WWW or FTP and stores it in your temporary directory. Equivalent to choosing File - Open and clicking Internet. The parameters for this method correspond to the options available in the Open from Internet dialog box.

### **Syntax**

[objectreference].RetrievalInternetFile()

### **Parameters**

#### *URL*

A String expression specifying the URL for the document you want to open.

#### *UserID*

Used when retrieving a file from an FTP site, this parameter takes a String value representing the name of the user who has an account with the FTP server. If you are retrieving a WWW document, the value of this parameter should be a null string ("").

#### *Password*

Used when retrieving a file from an FTP site, this parameter takes a String value representing the password for the user named in UserID. If you are retrieving a WWW document, the value of this parameter should be a null string ("").

#### *Passive*

Set this value to True when you want to initiate the file transfer. Set it to False to allow the server to repond to your request when it is ready. Some FTP servers do not support this feature. The value of this parameter is usually False. If you are retrieving a WWW document, the value of this parameter should be a null string (""). Data type is Integer, but the value is always 0 (False) or -1 (True). You can use the LotusScript constants of True and False.

#### *Proxy*

A String expression specifying the DNS (for example, screen.companyname.com) or IP address (for example, 123.456.78.912). Do not include the "http:\\" in front of the the proxy value.

#### *ProxyPort*

An Integer which specifies the port number for the Proxy server. The value of this parameter is usually 8080 for the WWW and 21 for FTP, but you should check with your Internet Service Provider for your settings.

### **Return value**

A String representing the name of the file you retrieved.

### **Usage**

This method will not work unless your machine is configured for Internet access. A standard Internet access configuration includes a WINSOCK compliant DLL.

The retrieved file will be stored in the directory you marked as your temporary directory. The temporary directory is identified in an environment variable, but most temporary directories are named "TEMP."

**Word Pro: RevertToSaved method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVERTTOSAVED\_METHOD\_EXSCRIPT',1)} [See example](#)

Cancels changes made to the document since it was last saved and displays the previously saved version of the document. Equivalent to closing a file without saving the changes and then reopening the file.

**Syntax**

[objectreference].RevertToSaved()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**



## Word Pro: RevertToStyle method

```
{button ,AL('H_ALIGNMENT_CLASS;H_AMIKAKE_CLASS;H_ATTRIBUTES_CLASS;H_BREAKS_CLASS;H_BULLET_CLASS;H_CELLCONTAINER_CLASS;H_CHARACTERBORDER_CLASS;H_CLICKHERE_CLASS;H_COLOR_CLASS;H_DROPCAPCONTAINER_CLASS;H_FONT_CLASS;H_FRAMECONTAINER_CLASS;H_INDENT_CLASS;H_KINSOKU_CLASS;H_LANGUAGE_CLASS;H_NOTECONTAINER_CLASS;H_NUMBERING_CLASS;H_PAGECONTAINER_CLASS;H_PARAGRAPHBORDER_CLASS;H_RELATIVEINDENT_CLASS;H_RUBYCONTAINER_CLASS;H_SPACING_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABRACK_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL('H_REVERTTOSTYLE_METHOD_EXSCRIPT',1)} See example
```

Reverts an object to a style contained within the document's SmartMaster.

### Syntax

When called from a Color or a Language object:

```
[Objectreference].RevertToStyle()
```

When called from any other object:

```
[Objectreference].RevertToStyle(RevertType)
```

### Parameters

#### *RevertType*

Indicates the type of object which is being reverted to the original style. Always has a data type of Variant, but will accept only one value for each type of object.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

[Color]

Removes the current local color attribute and reverts to the original color attribute set for the associated style.

[Amikake]

If you are using an English language version of Word Pro, this method as a member of Amikake is not available.

[CellContainer, FrameContainer, SuperTableContainer]

Reverts all the properties of the container's layout to its style as indicated by the document's SmartMaster.

```
RevertToStyle:AlignStyleType RevertType=AlignStyleType;Alignment;Alignment;BaseObject
```

\$LwpAlignStyleType (2095)

```
RevertToStyle:AttributeProp RevertType=AttributeAll;Attributes;Attributes;BaseObject
```

\$LwpAttributePropAll (19)

\$LwpAttributePropBaseline (2200)

\$LwpAttributePropHidden (20)

\$LwpAttributePropHighlight (23)

\$LwpAttributePropHyphen (21)

\$LwpAttributePropLevel (26)

\$LwpAttributePropMisspelled (24)

\$LwpAttributePropProtected (22)

\$LwpAttributePropSkipped (25)

```
RevertToStyle:BreakProp RevertType=AllBreaksProperties;Breaks;Breaks;BaseObject
```

\$LwpBreakPropAllBreaks (59)

\$LwpBreakPropColumnBreakAfter (64)

\$LwpBreakPropColumnBreakBefore (63)

\$LwpBreakPropKeepWithNext (65)

\$LwpBreakPropKeepWithPrev (66)  
\$LwpBreakPropNextStyleName (68)  
\$LwpBreakPropPageBreakAfter (61)  
\$LwpBreakPropPageBreakBefore (60)  
\$LwpBreakPropPageBreakWithin (62)  
\$LwpBreakPropUseNextStyle (67)  
    RevertToStyle;BulletProperty RevertType=AllOverrides;Bullet;Bullet;BaseObject  
\$LwpBulletPropertyAll (82)  
\$LwpBulletPropertyBullet (83)  
\$LwpBulletPropertyEditable (86)  
\$LwpBulletPropertyRightalign (85)  
\$LwpBulletPropertyValidprop (84)  
    RevertToStyle;FontProperty RevertType=PropAll;Font;Font;BaseObject  
\$LwpFontPropertyAll (297)  
\$LwpFontPropertyAlllower (315)  
\$LwpFontPropertyAllupper (314)  
\$LwpFontPropertyBold (303)  
\$LwpFontPropertyCase (316)  
\$LwpFontPropertyDoubleunderline (308)  
\$LwpFontPropertyItalic (304)  
\$LwpFontPropertyName (298)  
\$LwpFontPropertyNextname (300)  
\$LwpFontPropertyOverstrike (312)  
\$LwpFontPropertySize (301)  
\$LwpFontPropertySmallcaps (313)  
\$LwpFontPropertySpacing (302)  
\$LwpFontPropertyStrikethru (311)  
\$LwpFontPropertySubscript (306)  
\$LwpFontPropertySuperscript (305)  
\$LwpFontPropertyUnderline (307)  
\$LwpFontPropertyWindowsname (299)  
\$LwpFontPropertyWorddoubleunderline (310)  
\$LwpFontPropertyWordunderline (309)  
    RevertToStyle;IndentProperty RevertType=IndentEvery;Indent;RelativeIndent;Indent, BaseObject  
    RevertToStyle;IndentProperty RevertType=IndentEvery;Indent;Indent;BaseObject  
\$LwpIndentPropertyAll (403)  
\$LwpIndentPropertyBodyonly (409)  
\$LwpIndentPropertyEnabled (410)  
\$LwpIndentPropertyEvery (402)  
\$LwpIndentPropertyFirst (404)  
\$LwpIndentPropertyHang (407)  
\$LwpIndentPropertyRelative (411)  
\$LwpIndentPropertyRest (405)  
\$LwpIndentPropertyRight (406)  
\$LwpIndentPropertySidesequal (408)  
    RevertToStyle;KinsokuProps RevertType=KinsokuAll;Kinsoku;Kinsoku;BaseObject  
\$LwpKinsokuPropsAll (429)

\$LwpKinsokuPropsEnabled (430)  
\$LwpKinsokuPropsHangover (431)  
\$LwpKinsokuPropsLevels (433)  
\$LwpKinsokuPropsSqueeze (432)  
    RevertToStyle;LineRevert RevertType;CharacterBorder;CharacterBorder;BaseObject  
    RevertToStyle;LineRevert RevertType=RevertAll;ParagraphBorder;ParagraphBorder;BaseObject  
\$LwpLineRevertAboveType (565)  
\$LwpLineRevertAboveWidth (563)  
\$LwpLineRevertAll (562)  
\$LwpLineRevertBelowType (566)  
\$LwpLineRevertBelowWidth (564)  
\$LwpLineRevertRightType (567)  
    RevertToStyle;none;Color;Color;BaseObject  
    RevertToStyle;none;Language;Language;BaseObject  
    RevertToStyle;NumberingProps RevertType=NumberingRevertAll;Numbering;Numbering;BaseObject  
\$LwpNumberingPropsRevertAll (1583)  
\$LwpNumberingPropsRevertHeading (1586)  
\$LwpNumberingPropsRevertLevel (1584)  
\$LwpNumberingPropsRevertPosition (1585)  
    RevertToStyle;RevertAll RevertType=RevertEverything;Amikake;Amikake;BaseObject  
\$LwpRevertAllEverything (1712)  
    RevertToStyle;SpacingProperty RevertType=SpacingAll;Spacing;Spacing;BaseObject  
\$LwpSpacingPropertyAboveamount (1784)  
\$LwpSpacingPropertyAbovenumber (1785)  
\$LwpSpacingPropertyAbovetype (1783)  
\$LwpSpacingPropertyAll (1778)  
\$LwpSpacingPropertyAlways (1779)  
\$LwpSpacingPropertyAmount (1787)  
\$LwpSpacingPropertyBelowamount (1781)  
\$LwpSpacingPropertyBelownumber (1782)  
\$LwpSpacingPropertyBelowtype (1780)  
\$LwpSpacingPropertyNumber (1788)  
\$LwpSpacingPropertyType (1786)  
    RevertToStyle;styleType RevertType;CellContainer;BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;ClickHere;Marker, BaseObject  
    RevertToStyle;styleType RevertType;NoteContainer;FrameContainer, BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;DropCapContainer;FrameContainer, BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;RubyContainer;FrameContainer, BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;FrameContainer;BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;PageContainer;BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;SubPageContainer;PageContainer, BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;SuperPageContainer;PageContainer, BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;SuperTableContainer;BaseContainer, BaseObject  
    RevertToStyle;styleType RevertType;Text;Content, BaseObject  
    RevertToStyle;styleType RevertType;TextMarker;Marker, BaseObject  
\$LwpStyleTypeCell (1834)  
\$LwpStyleTypeCharacter (1830)

\$LwpStyleTypeDefault (1828)

\$LwpStyleTypeFrame (1832)

\$LwpStyleTypePage (1831)

\$LwpStyleTypeParagraph (1829)

\$LwpStyleTypeTable (1833)

RevertToStyle;TabRackProperty RevertType=LwpTabRackPropAll;TabRack;TabRack;BaseObject

\$LwpTabRackPropertyAll (1859)

RevertToStyle;Variant RevertType=;WPApplication;WPApplication;Application, BaseObject

## **Word Pro: NumLinesOfSpaceAbove property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMLINESOFSPACEABOVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

numlinesofspaceabovevalue = [objectreference].NumLinesOfSpaceAbove

[objectreference].NumLinesOfSpaceAbove = numlinesofspaceabovevalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: NumLinesOfSpaceBelow property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMLINESOFSPACEBELOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

numlinesofspacebelowvalue = [objectreference].NumLinesOfSpaceBelow

[objectreference].NumLinesOfSpaceBelow = numlinesofspacebelowvalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: NumLinesOfSpace property**

{button ,AL('H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMLINESOFSPACE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

numlinesofspacevalue = [objectreference].NumLinesOfSpace

[objectreference].NumLinesOfSpace = numlinesofspacevalue

### **Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

**Word Pro: NumOfRecentFiles property**

{button ,AL(^H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NUMOFRECENTFILES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The maximum number of files allowed on the File menu.

**Data Type**

[Integer](#)

**Syntax**

numofrecentfilesvalue = [objectreference].NumOfRecentFiles

[objectreference].NumOfRecentFiles = numofrecentfilesvalue

**Legal values**

Integer between 0 and 5.

**Usage**

Equivalent to the "Recent files" option on the General panel of the Word Pro Preferences dialog box.



### **Word Pro: NumPagesInDoc property**

{button ,AL(^H\_DIVISION\_CLASS;H\_DOCINFO\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NUMPAGESINDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of pages in a document.

### **Data Type**

[Integer](#)

### **Syntax**

numpagesindocvalue = [objectreference].NumPagesInDoc

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumParagraphs property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMPARAGRAPHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to specify the number of paragraphs you want to sort.

### **Data Type**

[Integer](#)

### **Syntax**

numparagraphsvalue = [objectreference].NumParagraphs

[objectreference].NumParagraphs = numparagraphsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Text - Sort and selecting a number in the "Number of paragraphs/row in record" box. If you use this property in a table, it allows you to specify the number of rows you want to sort.

## Word Pro: NumRowsSpannedOneCell property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOOUT_CLASS;H_DROPCAPLAYOOUT_CLASS;H_ENDNOTELAYOOUT_CLASS;H_FOOTERLAYOOUT_CLASS;H_FOOTNOTELAYOOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOOUT_CLASS;H_PAGELAYOOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOOUT_CLASS;H_RUBYLAYOOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOOUT_CLASS;H_TABLEHEADINGLAYOOUT_CLASS;H_TABLELAYOOUT_CLASS;H_TOCSUPERTABLELAYOOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_NUMROWSSPANNEDONECELL_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) The number of table rows spanned by a connected cell.

## Data Type

[Integer](#)

## Syntax

numrowsspannedonecellvalue = [objectreference].NumRowsSpannedOneCell

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

In an unconnected cell, this property will contain a value of 1.

**Word Pro: NumRowsThatFit property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMROWSTHATFIT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

numrowsthatfitvalue = [objectreference].NumRowsThatFit

[objectreference].NumRowsThatFit = numrowsthatfitvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: NumRowsToFit property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMROWSTOFIT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

numrowstofitvalue = [objectreference].NumRowsToFit

[objectreference].NumRowsToFit = numrowstofitvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: NumRows property**

{button ,AL(^H\_BASETABLE\_CLASS;H\_CLICKHERE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASSES;H\_MARKER\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NUMROWS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of rows that have been assigned to hold contents.

### **Data Type**

[Integer](#)

### **Syntax**

numrowsvalue = [objectreference].NumRows

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumTabs property**

{button ,AL('H\_TABRACK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

numtabsvalue = [objectreference].NumTabs

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: NumUndoLevels property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMUNDOLEVELS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

numundolevelsvalue = [objectreference].NumUndoLevels

[objectreference].NumUndoLevels = numundolevelsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



**Word Pro: NumWindowsViewingDoc property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_NUMWINDOWSVIEWINGDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

numwindowsviewingdocvalue = [objectreference].NumWindowsViewingDoc

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: NumWordsInDoc property**

{button ,AL('H\_DOCINFO\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_NUMWORDSINDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of words in a document.

### **Data Type**

Long

### **Syntax**

numwordsindocvalue = [objectreference].NumWordsInDoc

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: ObjectType property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OBJECTTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Fribtype.

### **Data Type**

String

### **Syntax**

objectypevalue = [objectreference].ObjectType

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: Object property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OBJECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Variant](#)

**Syntax**

objectvalue = [objectreference].Object

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: Oblique property**

{button ,AL('H\_FONTMETRICS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OBLIQUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

obliquevalue = [objectreference].Oblique

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: Offset property**

{button ,AL('H\_INDEX\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OFFSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

offsetvalue = [objectreference].Offset

[objectreference].Offset = offsetvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: OleContainerDocName property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OLECONTAINERDOCNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

olecontainerdocnamevalue = [objectreference].OleContainerDocName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: OpenDocMacroName property**

{button ,AL('H\_AUTORUNMACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENDOCMACRONAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

opendocmacronamevalue = [objectreference].OpenDocMacroName

[objectreference].OpenDocMacroName = opendocmacronamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: OpenDocsVisible property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENDOCISIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Displays all documents that are opened in the current Word Pro session.

### **Data Type**

[Integer](#)

### **Syntax**

opendocsvisiblevalue = [objectreference].OpenDocsVisible

[objectreference].OpenDocsVisible = opendocsvisiblevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is 0.

### **Usage**

For some features, Word Pro creates, opens, modifies, and saves files without the user actually viewing them. An example of this functionality is Merge. When a user does a document merge, Word Pro creates, opens, modifies, and/or saves a Merge data file and the user never sees it.

If the value for this property is -1, the user can see all documents that are opened during the current session. If the value for this property is 0, files are not visible. To be able to see all files, you must set the property to -1.

## **Word Pro: OpenExistingFileInWelcomeBox property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENEXISTINGFILEINWELCOMEBOX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines which panel displays in the Welcome dialog box.

### **Data Type**

[Integer](#)

### **Syntax**

openexistingfileinwelcomeboxvalue = [objectreference].OpenExistingFileInWelcomeBox

[objectreference].OpenExistingFileInWelcomeBox = openexistingfileinwelcomeboxvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

In order for you to use this property, the value for the property, ShowNoWelcomeBox, must be False (0).

If the value for this property is True (-1), Word Pro displays the Welcome dialog box with the "Open an Existing Document" panel in the foreground. If the value for this property is False (0), Word Pro displays the Welcome dialog box with the "Create a New Document from a SmartMaster" panel in the foreground.

## **Word Pro: OpenReadOnly property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPENREADONLY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Opens a file as read-only.

### **Data Type**

[Integer](#)

### **Syntax**

openreadonlyvalue = [objectreference].OpenReadOnly

[objectreference].OpenReadOnly = openreadonlyvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Use this property to open any document in read-only mode.

**Word Pro: OpionDeclare property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPIONDECLARE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

opiondeclarevalue = [objectreference].OpionDeclare

[objectreference].OpionDeclare = opiondeclarevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: Options property

{button ,AL('H\_FORMATCHECKPREF\_CLASS;H\_MERGEOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[MergeOptions]

A flag that sets various options for a Merge.

## Data Type

Long (Enumerated Bitmask)

FormatCheckOptions

MergeOptFlg (Variant)

## Syntax

optionsvalue = [objectreference].Options

[objectreference].Options = optionsvalue

## Legal values

LwpFormatCheckOptionsAll (&H1F)

LwpFormatCheckOptionsFixacronyms (&H1)

LwpFormatCheckOptionsFixbulletlist (&H4)

LwpFormatCheckOptionsFixmargins (&H8)

LwpFormatCheckOptionsFixspaces (&H2)

LwpFormatCheckOptionsTwospaces (&H10)

LwpMergeOptFlgMergeAndPrint (&H1) Prints the Merge data file. If you set this flag, you cannot view any data records as you merge.

LwpMergeOptFlgMergeViewAndPrint (&H2) Allows you to view each data record in the Merge process.

LwpMergeOptFlgMergeAndSaveAs (&H4)

LwpMergeOptFlgMergeInsertFields (&H8) Indicates you inserted a Merge data field.

LwpMergeOptFlgMergeLabels (&H10)

LwpMergeOptFlgMergeMacro (&H20) Prevents the Merge bar from displaying during the Merge process.

## Usage

## **Word Pro: OrigFileType property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ORIGFILETYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

origfiletypevalue = [objectreference].OrigFileType

[objectreference].OrigFileType = origfiletypevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: OrigHeight property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ORIGHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Long](#)

**Syntax**

origheightvvalue = [objectreference].OrigHeight

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: OrigWidth property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ORIGWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Long](#)

### **Syntax**

origwidthvalue = [objectreference].OrigWidth

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: OutlineButtons property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINEBUTTONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

outlinebuttonsvalue = [objectreference].OutlineButtons

[objectreference].OutlineButtons = outlinebuttonsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: OutlineHeadingButtonsOnly property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINEHEADINGBUTTONSONLY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

outlineheadingbuttonsonlyvalue = [objectreference].OutlineHeadingButtonsOnly

[objectreference].OutlineHeadingButtonsOnly = outlineheadingbuttonsonlyvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: OutlineLevel property**

{button ,AL('H\_PRESENTATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINELEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

outlinelevelvalue = [objectreference].OutlineLevel

[objectreference].OutlineLevel = outlinelevelvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: OutlineOnlyHeadingsWhenCollapsed property**

{button ,AL('H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINEONLYHEADINGSWHENCOLLAPSED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

outlineonlyheadingswhencollapsedvalue = [objectreference].OutlineOnlyHeadingsWhenCollapsed

[objectreference].OutlineOnlyHeadingsWhenCollapsed = outlineonlyheadingswhencollapsedvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: Outline property**

{button ,AL('H\_FONTMETRICS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

outlinevalue = [objectreference].Outline

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: OutputToFile property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OUTPUTTOFILE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to print a document from a computer that does not have Word Pro installed.

### **Data Type**

[Integer](#)

### **Syntax**

outputtofilevalue = [objectreference].OutputToFile

[objectreference].OutputToFile = outputtofilevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing File - Print and selecting "Print to file."

## Word Pro: Override property

{button ,AL('H\_COLOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OVERRIDE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Determines what information will be used to define an object's color.

### Data Type

Variant (Enumerated)

### Syntax

overridevalue = [objectreference].Override

[objectreference].Override = overridevalue

### Legal values

\$LwpColorOverrideBlack (2017) Assigns the predefined WordPro color black to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideBlue (2024) Assigns the predefined WordPro color blue to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideDarkGray (2021) Assigns the predefined WordPro color dark gray to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideGray (2019) Assigns the predefined WordPro color gray to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideGreen (2023) Assigns the predefined WordPro color green to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideInvalid (2026)

\$LwpColorOverrideLightGray (2020) Assigns the predefined WordPro color light gray to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideRed (2022) Assigns the predefined WordPro color red to an object and ignores the red, green, and blue properties of the color object.

\$LwpColorOverrideReserved (2025) This value is not implemented.

\$LwpColorOverrideRgb (2016) Assigns the values in the red, green, and blue properties of the color object.

\$LwpColorOverrideWhite (2018) Assigns the predefined WordPro color white to an object and ignores the red, green, and blue properties of the color object.

### Usage

If the Override property value is set to RGB, then the object's color is defined by the values in the red, green, and blue properties of the color object. All other Override values use a predefined WordPro color and ignore the red, green, and blue properties of the color object.

**Word Pro: OverstrikeCharacter property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OVERSTRIKECHARACTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

overstrikecharactervalue = [objectreference].OverstrikeCharacter

[objectreference].OverstrikeCharacter = overstrikecharactervalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**



## **Word Pro: Overstrike property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_OVERSTRIKE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

overstrikevalue = [objectreference].Overstrike

[objectreference].Overstrike = overstrikevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: PageNo property**

{button ,AL('H\_USEWHEN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGENO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

pagenovalue = [objectreference].PageNo

[objectreference].PageNo = pagenovalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: PageNumberAsText property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PAGENUMBERASTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

pagenumberastextvalue = [objectreference].PageNumberAsText

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: PageNumberStyle property

{button ,AL('H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGENUMBERSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Variant (Enumerated)

NumberingStyle

### Syntax

pagenumberstylevalue = [objectreference].PageNumberStyle

[objectreference].PageNumberStyle = pagenumberstylevalue

### Legal values

\$LwpNumberingStyleBasic (1588)

\$LwpNumberingStyleChar (1593)

\$LwpNumberingStyleFullpitchbasic (1594)

\$LwpNumberingStyleFullpitchlowercase (1597)

\$LwpNumberingStyleFullpitchuppercase (1596)

\$LwpNumberingStyleFullpitchverbasic (1595)

\$LwpNumberingStyleLowercaseletters (1590)

\$LwpNumberingStyleLowercaseroman (1592)

\$LwpNumberingStyleNone (1587)

\$LwpNumberingStyleUppercaseletters (1589)

\$LwpNumberingStyleUppercaseroman (1591)

### Usage

## **Word Pro: PageNumber property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PAGENUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

pagenumbervalue = [objectreference].PageNumber

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: PageNumFirstPageShowing property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGENUMFIRSTPAGESHOWING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

pagenumfirstpageshowingvalue = [objectreference].PageNumFirstPageShowing

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: PageNumLastPageShowing property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGENUMLASTPAGESHOWING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

pagenumlastpageshowingvalue = [objectreference].PageNumLastPageShowing

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PageNum property**

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCO  
NTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGENUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the page number where the current container is located.

### **Data Type**

Integer

### **Syntax**

pagenumvalue = [objectreference].PageNum

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: PageOrder property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGEORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies if a document prints from front to back or from back to front.

### **Data Type**

Variant (Enumerated)

PageOrder

### **Syntax**

pageordervalue = [objectreference].PageOrder

[objectreference].PageOrder = pageordervalue

### **Legal values**

\$LtsPageOrderBackToFront (1056964852) Specifies if a document prints from back to front.

\$LtsPageOrderFrontToBack (1056964851) Specifies if a document prints from front to back.

### **Usage**

## **Word Pro: PageStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAGESTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

pagestylevalue = [objectreference].PageStyleName

[objectreference].PageStyleName = pagestylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: PageToUseLayoutOn property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_PAGETOUSELAYOUTON_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the page on which a specific layout will be used.

## Data Type

[Integer](#)

## Syntax

pagetouselayoutonvalue = [objectreference].PageToUseLayoutOn

[objectreference].PageToUseLayoutOn = pagetouselayoutonvaluew

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

## **Word Pro: PairKerning property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PAIRKERNING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

pairkerningvalue = [objectreference].PairKerning

[objectreference].PairKerning = pairkerningvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: ParagraphStyleName property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARAGRAPHSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The name of the paragraph style assigned to the current paragraph.

### **Data Type**

[String](#)

### **Syntax**

paragraphstylevalue = [objectreference].ParagraphStyleName

[objectreference].ParagraphStyleName = paragraphstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: ParagraphSymbolChar property**

{button ,AL('H\_CHARACTERSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARAGRAPHSYMBOLCHAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

paragraphsymbolcharvalue = [objectreference].ParagraphSymbolChar

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: ParentMenuHWND property**

{button ,AL('H\_MENUITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARENTMENUHWND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The handle to the parent window for a menu item.

**Data Type**

[Long](#)

**Syntax**

parentmenuhwndvalue = [objectreference].ParentMenuHWND

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

This property returns the handle of the parent window of a specific menu item. Use this handle to make API calls. This property is rarely used inside Word Pro under normal circumstances.

**Word Pro: ParentName property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARENTNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

parentnamevalue = [objectreference].ParentName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**



**Word Pro: Partial property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PARTIAL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

partialvalue = [objectreference].Partial

[objectreference].Partial = partialvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: PathName property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PATHNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

String

**Syntax**

pathnamevalue = [objectreference].PathName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: Path property**

{button ,AL(^H\_APPLICATION\_CLASS;H\_DOCUMENT\_CLASS;H\_INDEX\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The path in which the Word Pro executable is installed. Word Pro uses this path to find the .DLLs necessary for launching and running Word Pro.

### **Data Type**

String

### **Syntax**

pathnamevalue = [objectreference].Path

### **Legal values**

The value of this property cannot be set by a script.

### **Usage**

Some Word Pro users may install Word Pro in a path that is different from the default path provided during installation. You can use this property when you need the path to the Word Pro application or its components but you don't know if everyone installed to the same directory.

## Word Pro: Pattern property

{button ,AL('H\_BACKGROUND\_CLASS;H\_BORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PATTERN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

## Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

patternvalue = [objectreference].Pattern

[objectreference].Pattern = patternvalue

## Legal values

[Background]

\$LtsFillBarLeftDiag (1056964681)

\$LtsFillBarRightDiag (1056964688)

\$LtsFillBasket (1056964710)

\$LtsFillBigCheck (1056964717)

\$LtsFillBottomTopGrad (1056964729)

\$LtsFillBrick (1056964712)

\$LtsFillBubbles (1056964724)

\$LtsFillChevron (1056964708)

\$LtsFillCicles (1056964719)

\$LtsFillClumpedNarrowDiagHatch (1056964696)

\$LtsFillClumpedZs (1056964723)

\$LtsFillDarkNarrowDiagHatch (1056964693)

\$LtsFillDiagBasket (1056964711)

\$LtsFillDiagBrick (1056964713)

\$LtsFillDiagHatch (1056964695)

\$LtsFillDiamonds (1056964725)

\$LtsFillDottedDarkHatch (1056964853)

\$LtsFillDottedZigzag (1056964726)

\$LtsFillDoubleLeftDiag (1056964684)

\$LtsFillDoubleRightDiag (1056964690)

\$LtsFillGray1 (1056964669)

\$LtsFillGray10 (1056964678)

\$LtsFillGray2 (1056964670)

\$LtsFillGray3 (1056964671)

\$LtsFillGray4 (1056964672)

\$LtsFillGray5 (1056964673)

\$LtsFillGray6 (1056964674)

\$LtsFillGray7 (1056964675)

\$LtsFillGray8 (1056964676)

\$LtsFillGray9 (1056964677)

\$LtsFillHoriz (1056964699)

\$LtsFillHorizBar (1056964698)

\$LtsFillHorizCheckerboard (1056964716)

\$LtsFillIrregularDiagScales (1056964721)  
\$LtsFillLeftDiag (1056964682)  
\$LtsFillLeftNarrowDiagHatch (1056964694)  
\$LtsFillLeftRightGrad (1056964728)  
\$LtsFillNarrowDoubleLeftDiag (1056964685)  
\$LtsFillNarrowDoubleRightDiag (1056964854)  
\$LtsFillNarrowHoriz (1056964697)  
\$LtsFillNarrowVert (1056964701)  
\$LtsFillNeToSwDiagStripGrad (1056964738)  
\$LtsFillNeToSwGrad (1056964730)  
\$LtsFillNone (1056964667)  
\$LtsFillNwToSeDiagStripGrad (1056964739)  
\$LtsFillNwToSeGrad (1056964731)  
\$LtsFillRandomBar (1056964680)  
\$LtsFillRandomSquare (1056964679)  
\$LtsFillRegularCheck (1056964718)  
\$LtsFillRegularHatch (1056964706)  
\$LtsFillRightDiag (1056964689)  
\$LtsFillRtLeftGrad (1056964744)  
\$LtsFillRunningDash (1056964714)  
\$LtsFillScalesDown (1056964722)  
\$LtsFillScalesUp (1056964720)  
\$LtsFillSolid (1056964668)  
\$LtsFillSteel (1056964709)  
\$LtsFillTinyHatch (1056964705)  
\$LtsFillTopBottomGrad (1056964745)  
\$LtsFillTripleLeftDiag (1056964686)  
\$LtsFillTripleRightDiag (1056964691)  
\$LtsFillVert (1056964703)  
\$LtsFillVertBar (1056964702)  
\$LtsFillVertCheckerboard (1056964715)  
\$LtsFillWideHatch (1056964707)  
\$LtsFillWideHoriz (1056964700)  
\$LtsFillWideLeftDiag (1056964687)  
\$LtsFillWideRightDiag (1056964692)  
\$LtsFillWideVert (1056964704)  
\$LwpFillIndian3 (273)  
\$LwpFillPattern (2000)  
\$LwpFillPeachpie (274)

[BorderLines]

\$LtsBorderPatternDashDot (1056964659)  
\$LtsBorderPatternDashDotDot (1056964660)  
\$LtsBorderPatternDashed (1056964662)  
\$LtsBorderPatternDot (1056964663)  
\$LtsBorderPatternDouble (1056964666)

\$LtsBorderPatternLongDash (1056964661)  
\$LtsBorderPatternNone (1056964657)  
\$LtsBorderPatternSolid (1056964658)  
\$LwpBorderPattern13space (36)  
\$LwpBorderPattern31space (37)  
\$LwpBorderPatternButttdown (35)  
\$LwpBorderPatternButtonup (34)  
\$LwpBorderPatternCircle (41)  
\$LwpBorderPatternDbIThick (51)  
\$LwpBorderPatternDbIWavy (56)  
\$LwpBorderPatternDeco1 (44)  
\$LwpBorderPatternDeco2 (45)  
\$LwpBorderPatternDeco3 (50)  
\$LwpBorderPatternDiagonal (38)  
\$LwpBorderPatternPin (47)  
\$LwpBorderPatternRain (46)  
\$LwpBorderPatternRope (43)  
\$LwpBorderPatternRose (48)  
\$LwpBorderPatternStar (42)  
\$LwpBorderPatternSunf (49)  
\$LwpBorderPatternTaro (39)  
\$LwpBorderPatternThickDbIwavy (58)  
\$LwpBorderPatternThickThin (53)  
\$LwpBorderPatternThickWavy (57)  
\$LwpBorderPatternThinThick (54)  
\$LwpBorderPatternThinThickThin (52)  
\$LwpBorderPatternWavy (55)  
\$LwpLtsBorderPatternDot (40)

## **Usage**

## Word Pro: Percentage property

{button ,AL('H\_JOIN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PERCENTAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Defines the percentage property used in calculating the width and height of the bounding area of a scalable join object.

### Data Type

Integer

### Syntax

percentagevalue = [objectreference].Percentage

[objectreference].Percentage = percentagevalue

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage

A join object can be scalable or fixed.

#### Scalable join objects

You can set the ScaleMode property of a scalable join object to scaling or no scaling. Setting the ScaleMode property to scaling for a scalable join object causes the join object's width and height to be a function of the page, table layout, or frame container's height and width and the Percentage property. (For more information, see ScaleMode property.) You can only use the Percentage property on join objects whose ScaleMode property is set to scaling.

Setting the ScaleMode property to no scaling for a scalable join object causes the width and height of a join object to be the same as the width and height properties of the join object. The width and height of the join object does not change as the container's width and height changes.

The width and height of a scalable join with a no scaling setting are fixed. However, you can change the width and height properties of a join object.

#### Fixed join objects

A fixed join object has predefined Word Pro width and height properties. Therefore, you cannot change the width and height of a fixed join object. (For a list of which join objects are scalable or fixed, see JoinType property.)

The width and height of a scalable join object is determined by the following algorithm:

```
if (join percentage==100)
{
    joinwidth=container width/2
    joinheight=container height/2
}
else
{
    if containerHeight< containerWidth
        joinWidth=joinHeight=(containerHeight/200) * percentage
    elseif
        joinWidth=joinHeight=(containerWidth/200) * percentage
}
```

**Word Pro: PersonalData1 property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PERSONALDATA1\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

personaldata1value = [objectreference].PersonalData1

[objectreference].PersonalData1 = personaldata1value

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**



**Word Pro: PersonalData2 property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PERSONALDATA2\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

String

**Syntax**

personaldata2value = [objectreference].PersonalData2

[objectreference].PersonalData2 = personaldata2value

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: PersonalData3 property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PERSONALDATA3\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

personaldata3value = [objectreference].PersonalData3

[objectreference].PersonalData3 = personaldata3value

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: PersonalData4 property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PERSONALDATA4\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

personaldata4value = [objectreference].PersonalData4

[objectreference].PersonalData4 = personaldata4value

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: PhoneNumber property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PHONENUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

phonenum value = [objectreference].PhoneNumber

[objectreference].PhoneNumber = phonenum value

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PitchAndFamily property**

{button ,AL('H\_FONTMETRICS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PITCHANDFAMILY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

pitchandfamilyvalue = [objectreference].PitchAndFamily

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PointSize property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POINTSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Points](#)

### **Syntax**

pointsizevalue = [objectreference].PointSize

[objectreference].PointSize = pointsizevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PositionType property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POSITIONTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not the icon bar object is at the left, right, top, bottom, or floating position on the workspace.

### **Data Type**

Variant (Enumerated)

### **Syntax**

positiontypevalue = [objectreference].PositionType

[objectreference].PositionType = positiontypevalue

### **Legal values**

\$LwplconBarPosBottomOfWindow (390) Indicates if a set of SmartIcons is at the bottom of the workspace in a fixed position.

\$LwplconBarPosFloating (391) Indicates if a set of SmartIcons is in a floating position anywhere inside or outside the workspace.

\$LwplconBarPosLeftSideOfWindow (388) Indicates if a set of SmartIcons is at the left side of the workspace in a fixed position.

\$LwplconBarPosRightSideOfWindow (389) Indicates if a set of SmartIcons is at the right side of the workspace in a fixed position.

\$LwplconBarPosTopOfWindow (387) Indicates if a set of SmartIcons is at the top of the workspace in a fixed position.

### **Usage**

Allows you to move an icon bar object in the left, right, top, bottom, or floating position on the workspace.

**Word Pro: PositionXInContainer property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_POSITIONXINCONTAINER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The X coordinate of the insertion point relative to the top left corner of the container in which the insertion point is located.

**Data Type**

[Long](#)

**Syntax**

positionxincontainervalue = [objectreference].PositionXInContainer

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

A container can be a comment note, a footer, a page, a cell, a frame, and so on.



## Word Pro: PositionXOnPage property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_CLICKHERE\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_NOTECONTAINER\_CLASS;H\_OLEOBJECT\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASS;H\_TABLEONLYCONT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POSITIONXONPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The X coordinate of an object relative to the top left corner of the page.

### Data Type

Long

### Syntax

positionxonpagevalue = [objectreference].PositionXOnPage

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

**Word Pro: PositionYInContainer property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_POSITIONYINCONTAINER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The Y coordinate of the insertion point relative to the top left corner of the container in which the insertion point is located.

**Data Type**

Long

**Syntax**

positionyincontainervalue = [objectreference].PositionYInContainer

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

A container can be a comment note, a footer, a page, a cell, a frame, and so on.

## Word Pro: PositionYOnPage property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_CLICKHERE\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_NOTECONTAINER\_CLASS;H\_OLEOBJECT\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASS;H\_TABLEONLYCONT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POSITIONYONPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The Y coordinate of an object relative to the top left corner of the page.

### Data Type

Long

### Syntax

positionyonpagevalue = [objectreference].PositionYOnPage

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

**Word Pro: Position property**

{button ,AL('H\_ALIGNMENT\_CLASS;H\_NUMBERING\_CLASS;H\_OUTSEQITEM\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POSITION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates the position of an object in units of Twips.

**Data Type**

Long

**Syntax**

positionvalue = [objectreference].Position

[objectreference].Position = positionvalue

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

**Word Pro: PowerField property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_POWERFIELD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

powerfieldvalue = [objectreference].PowerField

[objectreference].PowerField = powerfieldvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: Prefix property**

{button ,AL('H\_NUMERICFORMATSUBSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREFIX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

prefixvalue = [objectreference].Prefix

[objectreference].Prefix = prefixvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PrettyPrinting property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRETTYPRINTING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

prettyprintingvalue = [objectreference].PrettyPrinting

[objectreference].PrettyPrinting = prettyprintingvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PreviousClickHere property**

{button ,AL(`H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_PREVIOUSCLICKHERE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name of the ClickHere block which precedes the current ClickHere in the division (uses Tab order.)

### **Data Type**

String

### **Syntax**

previousclickherevalue = [objectreference].PreviousClickHere

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: PreviousName property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREVIOUSNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

previousnamevalue = [objectreference].PreviousName

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PreviousNeighbor property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PREVIOUSNEIGHBOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

previousneighborvalue = [objectreference].PreviousNeighbor

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PrintDestination property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTDESTINATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

printdestinationvalue = [objectreference].PrintDestination

[objectreference].PrintDestination = printdestinationvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: PrintDocDescription property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTDOCDESCRIPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Prints the description of a document.

**Data Type**

[Integer](#)

**Syntax**

printdocdescriptionvalue = [objectreference].PrintDocDescription

[objectreference].PrintDocDescription = printdocdescriptionvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Setting this property prints the document description that displays in the "Contents" box when you select "Document Description" on the Fields panel in the Document Properties dialog box.

**Word Pro: PrinterName property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

printernamevalue = [objectreference].PrinterName

[objectreference].PrinterName = printernamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: PrintGraphics property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTGRAPHICS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to print a specific graphic in a document.

### **Data Type**

[Integer](#)

### **Syntax**

printgraphicsvalue = [objectreference].PrintGraphics

[objectreference].PrintGraphics = printgraphicsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: PrintInBackground property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTINBACKGROUND\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to perform other Word Pro tasks while a document prints.

### **Data Type**

[Integer](#)

### **Syntax**

printinbackgroundvalue = [objectreference].PrintInBackground

[objectreference].PrintInBackground = printinbackgroundvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: PrintPagesFrom property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTPAGESFROM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies the last page number within a range of pages that you want to print in a document.

**Data Type**

[Integer](#)

**Syntax**

printpagesfromvalue = [objectreference].PrintPagesFrom

[objectreference].PrintPagesFrom = printpagesfromvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to choosing File - Print and selecting a number in the "Pages from" box.



**Word Pro: PrintPagesTo property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTPAGESTO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies the last page within a range of pages that you want to print in a document.

**Data Type**

[Integer](#)

**Syntax**

printpagestovalue = [objectreference].PrintPagesTo

[objectreference].PrintPagesTo = printpagestovalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to choosing File - Print and selecting a number in the "Pages to" box.

## **Word Pro: PrintPageType property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTPAGETYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies the printing of odd and even pages, only even pages, or only odd pages in a document.

### **Data Type**

Variant (Enumerated)

PrintPage

### **Syntax**

printpagetypevalue = [objectreference].PrintPageType

[objectreference].PrintPageType = printpagetypevalue

### **Legal values**

\$LwpPrintPageEvenAndOddPages (1651) Allows you to specify the printing of odd and even pages in a document.

\$LwpPrintPageEvenPages (1652) Allows you to specify the printing of only even pages in a document.

\$LwpPrintPageOddPages (1653) Allows you to specify the printing of only odd pages in a document.

### **Usage**

Equivalent to choosing File - Print and selecting an option from the "Including" box.

## **Word Pro: PrintRange property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PRINTRANGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to specify which pages in a document you want to print. You can print all pages, the current page or division, specific pages or divisions, or a range of pages, such as pages 2 - 10.

### **Data Type**

Variant (Enumerated)

PrintRange

### **Syntax**

prinrangevalue = [objectreference].PrintRange

[objectreference].PrintRange = prinrangevalue

### **Legal values**

\$LtsPrintRangeAllPages (1056964848) Allows you to print all the pages in a document.

\$LtsPrintRangeCurrentPage (1056964849) Allows you to print only the current page in a document.

\$LtsPrintRangeSelectedPages (1056964850) Allows you to print only selected pages in a document.

\$LwpPrintRangeCurrentDivision (1655) Allows you to print only the current division in a document.

\$LwpPrintRangeRangeOfPages (1654) Allows you to specify a range of pages to print in a document.

\$LwpPrintRangeSelectedDivisions (1656) Allows you to specify a range of divisions to print in a document.

### **Usage**

Equivalent to choosing File - Print and selecting the desired options in the "Print range" section or the "Print" section in the Print dialog box.

## **Word Pro: Private property**

{button ,AL(^H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_PRIVATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

### **Data Type**

Integer

### **Syntax**

### **Legal values**

### **Usage**

**Word Pro: PromptHidden property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PROMPTHIDDEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) True makes the Prompt text hidden. False makes the Prompt text visible.

**Data Type**

[Integer](#)

**Syntax**

prompthiddenvalue = [objectreference].PromptHidden

[objectreference].PromptHidden = prompthiddenvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: ProtectedMode property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_PROTECTEDMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

protectedmodevalue = [objectreference].ProtectedMode

[objectreference].ProtectedMode = protectedmodevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: QueueName property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_QUEUE\_NAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

queuevalue = [objectreference].QueueName

[objectreference].QueueName =queuevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

### **Word Pro: ReadOnly property**

{button ,AL('H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_READONLY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

readonly value = [objectreference].ReadOnly

[objectreference].ReadOnly = readonlyvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



**Word Pro: Red property**

{button ,AL('H\_COLOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The red component of a color.

**Data Type**

[Integer](#)

**Syntax**

redvalue = [objectreference].Red

[objectreference].Red = redvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

You can set or read the current level of red in a specific object's color.

## Word Pro: RelativePageNum property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCO  
NTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RELATIVEPAGENUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

## Data Type

Integer

## Syntax

relativepagenumvalue = [objectreference].RelativePageNum

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

## Word Pro: RelativeType property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_RELATIVETYPE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Determines where the layout is anchored in the parent layout.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

relativetypevalue = [objectreference].RelativeType

[objectreference].RelativeType = relativetypevalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpRelativeTypeLytContent (1670)	The layout is anchored relative to the content box of the parent.
\$LwpRelativeTypeLytInline (1668)	The layout is anchored to the text flow as a single character and affects the line height accordingly.
\$LwpRelativeTypeLytInlineNewline (1669)	Places the layout on a new line by itself.
\$LwpRelativeTypeLytInlineVert (1671)	The layout is anchored so that it always moves vertically.
\$LwpRelativeTypeLytPara (1667)	The layout is anchored in the text relative to a paragraph.
\$LwpRelativeTypeLytParent (1666)	The coordinates of the layout are relative to the parent layout and the anchor position is on the parent container.

## Usage

## Word Pro: RelativeXDistance property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_RELATIVEXDISTANCE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The horizontal length of the anchor tether from a layout to its anchor point in the parent layout.

### Data Type

Long

### Syntax

relativexdistancevalue = [objectreference].RelativeXDistance

[objectreference].RelativeXDistance = relativexdistancevalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

## Word Pro: RelativeYDistance property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_RELATIVEYDISTANCE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The vertical length of the anchor tether from a layout to its anchor point in the parent layout.

## Data Type

Long

## Syntax

relativeydistancevalue = [objectreference].RelativeYDistance

[objectreference].RelativeYDistance = relativeydistancevalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

## Word Pro: Relative property

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RELATIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Variant (Enumerated)

IndentProperty

### Syntax

relativevalue = [objectreference].Relative

[objectreference].Relative = relativevalue

### Legal values

\$LwpIndentPropertyAll (403)

\$LwpIndentPropertyBodyonly (409)

\$LwpInden

tPropertyEnabled (410)

\$LwpIndentPropertyEvery (402)

\$LwpIndentPropertyFirst (404)

\$LwpIndentPropertyHang (407)

\$LwpIndentPropertyRelative (411)

\$LwpIndentPropertyRest (405)

\$LwpIndentPropertyRight (406)

\$LwpIndentPropertySidesequal (408)

### Usage

**Word Pro: RenderedPageNumber property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RENDEREDPAGENUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

renderedpagenumbervalue = [objectreference].RenderedPageNumber

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: ReplaceExactCase property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACEEXACTCASE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to replace the exact case of a word or phrase in Find & Replace.

### **Data Type**

[Integer](#)

### **Syntax**

replaceexactcasevalue = [objectreference].ReplaceExactCase

[objectreference].ReplaceExactCase = replaceexactcasevalue

### **Legal values**

Always contains an instance of the Text class. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property to check and manipulate the setting for replacing the exact case of a word or phrase in Find & Replace. If True, replaces the word that matches the user setting. Equivalent of choosing Edit - Find & Replace Text, clicking Options, and selecting "Exact case" in the "Replace options" section.



## Word Pro: Replacements property

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACEMENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Long (Enumerated Bitmask)

ReplacementChoices

### Syntax

replacementsvalue = [objectreference].Replacements

[objectreference].Replacements = replacementsvalue

### Legal values

LwpReplacementChoicesListbox22 (&H200000)

LwpReplacementChoicesListbox1 (&H1)

LwpReplacementChoicesListbox10 (&H200)

LwpReplacementChoicesListbox11 (&H400)

LwpReplacementChoicesListbox12 (&H800)

LwpReplacementChoicesListbox13 (&H1000)

LwpReplacementChoicesListbox14 (&H2000)

LwpReplacementChoicesListbox15 (&H4000)

LwpReplacementChoicesListbox16 (&H8000)

LwpReplacementChoicesListbox17 (&H10000)

LwpReplacementChoicesListbox18 (&H20000)

LwpReplacementChoicesListbox19 (&H40000)

LwpReplacementChoicesListbox2 (&H2)

LwpReplacementChoicesListbox20 (&H80000)

LwpReplacementChoicesListbox21 (&H100000)

LwpReplacementChoicesListbox22 (&H200000)

LwpReplacementChoicesListbox23 (&H400000)

LwpReplacementChoicesListbox24 (&H800000)

LwpReplacementChoicesListbox25 (&H1000000)

LwpReplacementChoicesListbox26 (&H2000000)

LwpReplacementChoicesListbox3 (&H4)

LwpReplacementChoicesListbox4 (&H8)

LwpReplacementChoicesListbox5 (&H10)

LwpReplacementChoicesListbox6 (&H20)

LwpReplacementChoicesListbox7 (&H40)

LwpReplacementChoicesListbox8 (&H80)

LwpReplacementChoicesListbox9 (&H100)

LwpReplacementChoicesListboxAll (&H3FFFFFF)

### Usage

## **Word Pro: ReplaceString property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACESTRING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to type what to replace in Find & Replace.

### **Data Type**

[String](#)

### **Syntax**

replacestringvalue = [objectreference].ReplaceString

[objectreference].ReplaceString = replacestringvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text and typing a text or character string in the "Replace with" box.

'Example: OleContainerDocName property

'This example script has not yet been created.

'Example: OleObjects property

'This example script has not yet been created.

'Example: OleObject property

'This example script has not yet been created.

```
'Example: OpenDataFile method
' This example opens the merge data file for the current merge document.
' RUNTIME DEPENDENCIES: You must have a document open that has been assigned a
' merge data file for this script to work.

DataFile = .ActiveDocument.MergeOptions.DataFileName
If DataFile <> False Then
    Print .OpenDataFile(DataFile,"Lotus Word Pro")
End If
```

'Example: OpenDocMacroName property  
'This example script has not yet been created.

'Example: OpenDocsVisible property

'This example script has not yet been created.



```
'Example: OpenDocument method
' This example uses an input box to get a file name from you and then opens
' the document with that file name. The default file name is "README95.LWP".
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim FileName as String
FileName = InputBox$ ("Enter a filename to open, e.g., README95.LWP:", "Example
Script", "")

.OpenDocument FileName, "", "", "", False, True
```

'Example: Opened event

'This example script has not yet been created.

'Example: OpenExistingFileInWelcomeBox property  
'This example script has not yet been created.

'Example: OpenFromStorage method

'This example script has not yet been created.

'Example: OpenObject method

'This example script has not yet been created.

'Example: OpenReadOnly property

'This example script has not yet been created.

'Example: Open method

'This example script has not yet been created.

'Example: OpionDeclare property

'This example script has not yet been created.



'Example: Options property

'This example script has not yet been created.

'Example: OrigFileType property

'This example script has not yet been created.

'Example: OrigHeight property

'This example script has not yet been created.

'Example: OrigWidth property

'This example script has not yet been created.

'Example: OutlineBorderLines property  
'This example script has not yet been created.

'Example: OutlineButtons property

'This example script has not yet been created.

'Example: OutlineHeadingButtonsOnly property  
'This example script has not yet been created.

'Example: OutlineLevel property

'This example script has not yet been created.



'Example: OutlineMoveTextDown method

'This example script has not yet been created.

'Example: OutlineMoveTextUp method

'This example script has not yet been created.

'Example: OutlineOnlyHeadingsWhenCollapsed property  
'This example script has not yet been created.

'Example: OutlineSeqItems property  
'This example script has not yet been created.

'Example: OutlineStyleSequences property  
'This example script has not yet been created.

'Example: OutlineStyleSequence method  
'This example script has not yet been created.

'Example: Outline property

'This example script has not yet been created.

'Example: OutputToFile property

'This example script has not yet been created.



'Example: Override property

'This example script has not yet been created.

'Example: OverstrikeCharacter property

'This example script has not yet been created.

'Example: Overstrike property

'This example script has not yet been created.

```
'Example: PageDown method
' This example simulates pressing the page down and page up keys in the
' current document;
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.PageDown
MessageBox "Click OK to pageup.",MB_OK,"Example Script"
.PageUp
```

'Example: PageNo property

'This example script has not yet been created.

'Example: PageNumberAsText property

'This example script has not yet been created.

'Example: PageNumberStyle property

'This example script has not yet been created.

'Example: PageNumber property

'This example script has not yet been created.



'Example: PageNumFirstPageShowing property  
'This example script has not yet been created.

'Example: PageNumLastPageShowing property

'This example script has not yet been created.

'Example: PageNum property

'This example script has not yet been created.

'Example: PageOrder property

'This example script has not yet been created.

'Example: PageStyleName property

'This example script has not yet been created.

'Example: PageStyles property

'This example script has not yet been created.

'Example: Pages property

'This example script has not yet been created.

'Example: PageToUseLayoutOn property

'This example script has not yet been created.



```
'Example: PageUp method
' This example simulates pressing the page down and page up keys in the
' current document;
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.PageDown
MessageBox "Click OK to pageup.",MB_OK,"Example Script"
.PageUp
```

'Example: Page property

'This example script has not yet been created.

'Example: PairKerning property

'This example script has not yet been created.

```
'Example: PaneColor property  
'
```

'Example: PaperNames property

'This example script has not yet been created.

'Example: ParagraphBorder property

'This example script has not yet been created.

'Example: ParagraphStyleName property

'This example script has not yet been created.

'Example: ParagraphStyles property  
'This example script has not yet been created.



'Example: ParagraphStyle property

'This example script has not yet been created.

'Example: ParagraphSymbolChar property

'This example script has not yet been created.

'Example: ParallelColumns property  
'This example script has not yet been created.

'Example: ParentMenuHWND property

'This example script has not yet been created.

'Example: ParentName property

'This example script has not yet been created.

'Example: Parent property

'This example script has not yet been created.

'Example: Partial property

'This example script has not yet been created.

'Example: PasteLink method

' This example pastes an Ole object from its source application and creates  
' a linked Ole object.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.PasteLink "LWPOLEFormat", False, 0



```
'Example: PasteSpecial method
' This example pastes any CF_TEXT data from the clipboard.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

FormatCategories = .Text.GetPasteFormatCategories
If (FormatCategories And &H1) Then
    .PasteSpecial "CF_TEXT"
End If
```

'Example: Paste method

' This example inserts some text into the current document. The text is then  
' selected and cut to the clipboard. After the message box is closed, the  
' text is then pasted back into the document.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "This is some sample text."

.Text.Select \$LwpSelectObjectTypeParagraph

.CutSelection

Messagebox "Click OK paste removed text.",MB\_OK,"Example Script"

.Paste

'Example: PathName property

'This example script has not yet been created.

'Example: Path property

'This example script has not yet been created.

'Example: Pattern property

'This example script has not yet been created.

```
'Example: PColConnectCells method
' This example creates a parallel column table with 3 columns. The first row
' is then selected and connected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, $LtsAlignmentHorizCenter
.PColSelectRow
.PColConnectCells
```

'Example: PColConnectRows method

' This example creates a parallel column table with 3 columns and then  
' connects the first row.

```
.CreateParallelColumns 3, $LtsAlignmentHorizCenter  
.PColConnectRows
```

```
'Example: PColDisconnectCells method
' This example creates a parallel column table with 3 columns. The first row
' is then selected and connected. After the message box is closed the cells
' are then disconnected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, $LtsAlignmentHorizCenter
.PColSelectRow
.PColConnectCells
MessageBox "Click OK to disconnect cells ",MB_OK,"Example Script"
.PColDisconnectCells
```



'Example: PColSelectColumn method

' This example creates a parallel column table with 3 columns and then selects  
' the first column.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.PColSelectColumn

'Example: PColSelectRow method

' This example creates a parallel column table with 3 columns and then selects  
' the first row.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.PColSelectRow

```
'Example: PColSelectTable method
' This example creates a parallel column table with 3 columns. The first row
' is then selected and connected. Text is typed into the selected row. A new
' row is inserted and then the entire table is selected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, $LtsAlignmentHorizCenter
.PColSelectRow
.PColConnectCells
.Type "This is a parallel column."
.BaseTable.InsertRowOrColumn $LwpTableInsTypeRow, True, 1
.PColSelectTable
```

'Example: Percentage property

'This example script has not yet been created.

'Example: PersonalData1 property

'This example script has not yet been created.

'Example: PersonalData2 property

'This example script has not yet been created.

'Example: PersonalData3 property

'This example script has not yet been created.

'Example: PersonalData4 property

'This example script has not yet been created.



'Example: PhoneNumber property

'This example script has not yet been created.

'Example: PitchAndFamily property

'This example script has not yet been created.

'Example: Play method

'This example script has not yet been created.

'Example: PointSize property

'This example script has not yet been created.

```
'Example: PositionType property
Dim CR As String*1
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager

IcnPallet = "Comment Tools"
CR = Chr(10)
Set IcnMgr = .ApplicationWindow.IconBarManager

With IcnMgr.IconBars(IcnPallet)
    MsgStr = "Height = " & .Height & CR
    MsgStr = MsgStr & "IconBarPositionState = " & .IconBarPositionState & CR
    MsgStr = MsgStr & "PositionType = " & .PositionType & CR
    MsgStr = MsgStr & "ScreenPositionX = " & .ScreenPositionX & CR
    MsgStr = MsgStr & "ScreenPositionY = " & .ScreenPositionY
    MessageBox MsgStr, 64, "Script Example - " & .Name
End With
```

'Example: PositionXInContainer property

'This example script has not yet been created.

'Example: PositionXOnPage property

'This example script has not yet been created.

'Example: PositionYInContainer property

'This example script has not yet been created.



'Example: PositionYOnPage property

'This example script has not yet been created.

'Example: Position property

'This example script has not yet been created.

'Example: PowerFields property

'This example script has not yet been created.

'Example: PowerField property

'This example script has not yet been created.

'Example: PreClose event

'This example script has not yet been created.

'Example: Preferences property

'This example script has not yet been created.

'Example: Prefix property

'This example script has not yet been created.

'Example: PrepareToDestroy method

'This example script has not yet been created.



'Example: PrePrint event

'This example script has not yet been created.

'Example: Presentation property

'This example script has not yet been created.

'Example: Present method

'This example script has not yet been created.

'Example: PrettyPrinting property

'This example script has not yet been created.

'Example: PreviousClickHere property

'This example script has not yet been created.

'Example: PreviousItem method

'This example script has not yet been created.

## **Word Pro: OutlineStyleSequence class**

The arrangement of styles in an outline.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: OutSeqItem class**

An item in an outline sequence.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

### **Usage**



## **Word Pro: PageContainer class**

The container object for pages. This object only exists for one page at a time and only when there is a page within the focus. When a PageContainer object is present, it is stored in the Page property on the WPAApplication object.

### **Base Classes**

BaseObject\BaseContainer

### **Derived Classes**

[SubPageContainer](#)

[SuperPageContainer](#)

### **Contained by**

[WPAApplication](#) in the [Page](#) Property

### **Usage**

The primary use for a PageContainer object is to provide quick and easy access to the PageLayout object for the currently active page. A PageContainer object always represents the page that currently has the focus. Thus, if you assign a PageContainer object to a variable, you can use that variable to access the currently active page. However, you must remember that the page referenced by the variable will change as the focus moves from one page to another. This is because the variable references the PageContainer object, and the PageContainer object always represents the page that has the focus. If there is no page within the focus, there is no PageContainer object. Thus, a variable that stores a PageContainer object will have a null value whenever the focus does not contain a page. There is never more than one PageContainer object at any given time.

For more information about container objects, see [BaseContainer](#).

**Word Pro: PageLayoutCollection class**

A collection of page layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Pages Property

Foundry in the PageStyles Property

**Usage**

Use this collection to access any of the page layout objects in the foundry of a specific division.

## **Word Pro: PageLayout class**

A page layout for a PageContainer object. This class inherits most of its members from the Layout class.

### **Base Classes**

BaseObject\Layout

### **Derived Classes**

None

### **Contained by**

### **Usage**

The PageLayout class provides you with a way to access and modify the format and appearance of PageLayout objects within your document.

Since the PageLayout class is derived from the Layout class, PageLayout objects can be stored within properties of the Layout type. For instance, the Layout property within the PageContainer class is of the Layout type. However, this property often stores objects of the PageLayout type. The Layout property is implemented in this way so that objects of other derived layout class types can be stored there as well.

PageLayout objects within a division are stored together in a collection. You can use the collection to access all PageLayout objects in the collection, or you can reference a particular PageLayout object in the collection. For example, by using the PageLayouts collection, you could modify each PageLayout object in the collection to have a specific top margin value. For more information on how to work with collections, see [Overview: Word Pro LotusScript Collection Classes](#).

At many locations within your document, multiple layouts are available. For example, your cursor may be in a frame on a page. In this case, the frame and the page both have associated layout objects. These layout objects may be combined with other objects into related groups, known as containers. For more information on containers and their associated layouts, see the Help topic titled [Word Pro: BaseContainer class](#).

## **Word Pro: ParagraphBorder class**

The border around a paragraph object.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [ParagraphBorder](#) Property

[ParagraphStyle](#) in the [ParagraphBorder](#) Property

[Text](#) in the [ParagraphBorder](#) Property

[TextMarker](#) in the [ParagraphBorder](#) Property

### **Usage**

**Word Pro: ParagraphStyleCollection class**

A collection of paragraph style objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the ParagraphStyles Property

**Usage**

Use this collection to access any of the paragraph style objects in the foundry of a specific division.

## **Word Pro: ParagraphStyle class**

The style assigned to a paragraph.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [ParagraphStyle](#) Property

[Formula](#) in the [ParagraphStyle](#) Property

[Text](#) in the [ParagraphStyle](#) Property

[TextMarker](#) in the [ParagraphStyle](#) Property

### **Usage**

**Word Pro: ParallelColsCollection class**

A collection of parallel column objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the ParallelColumns Property

**Usage**

Use this collection to access any of the parallel column objects in the foundry of a specific division.

**Word Pro: ParallelColsContainer class**

The container object for parallel columns. This object only exists for one parallel columns object at a time and only when there are parallel columns within the focus. When parallel columns are present, the ParallelColsContainer is stored in the TableContainer property on the WPApplication object.

**Base Classes**

BaseObject\BaseContainer\TableContainer

**Derived Classes**

None

**Contained by**

Not contained in any property.

**Usage**

The primary use for a ParallelColsContainer object is to provide quick and easy access to the CellLayout object for the currently active parallel columns. A ParallelColsContainer object always represents the parallel columns that currently have the focus. Thus, if you assign a ParallelColsContainer object to a variable, you can use that variable to access the currently active parallel columns. However, you must remember that the parallel columns referenced by the variable will change as the focus moves from one parallel columns object to another. This is because the variable references the ParallelColsContainer object, and the ParallelColsContainer object always represents the parallel columns that have the focus. If there are no parallel columns within the focus, there is no ParallelColsContainer object. Thus, a variable that stores a ParallelColsContainer object will have a null value whenever the focus does not contain parallel columns. There is never more than one ParallelColsContainer object at any given time.

For more information about container objects, see [BaseContainer](#).



## **Word Pro: ParallelColumns class**

A parallel (newspaper) column object in a document.

### **Base Classes**

BaseObject\Content\BaseTable

### **Derived Classes**

Glossary

### **Contained by**

WPApplication in the ParallelColumns Property

### **Usage**

**Word Pro: PowerFieldCollection class**

A collection of power field objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the PowerFields Property

**Usage**

Use this collection to access any of the power field objects in the foundry of a specific division.

## **Word Pro: PowerField class**

A power field object in a document.

### **Base Classes**

BaseObjectMarker

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: Preferences class**

Allows you to customize Word Pro by setting various options for file saving, file location, default files, and general and personal options.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

WPApplication in the Preferences Property

### **Usage**

## **Word Pro: Presentation class**

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

BaseContainer in the Presentation Property

### **Usage**

## **Word Pro: PrintManager class**

A tool to manage document printing.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

TextDocument in the PrintManager Property

### **Usage**

## **Word Pro: PrintSettings class**

The print settings displayed in the Print dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Document in the PrintSettings Property

### **Usage**

Use this class to display the current print settings for a specific document.

## **Word Pro: RelativeIndent class**

### **Base Classes**

BaseObject\Indent

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [RelativeIndent](#) Property

[ParagraphStyle](#) in the [RelativeIndent](#) Property

[Text](#) in the [RelativeIndent](#) Property

[TextMarker](#) in the [RelativeIndent](#) Property

### **Usage**



## **Word Pro: ReviewVersions class**

Allows you to review versions of a Word Pro document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the ReviewVersions Property

### **Usage**

## **Word Pro: RevisionDisplay class**

Allows you to display the Revision tool in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Preferences in the RevisionDisplay Property

### **Usage**

## **Word Pro: Revision class**

The Revision tool in the Word Pro application.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [RevisionMark](#) Property

[Text](#) in the [RevisionMark](#) Property

[TextMarker](#) in the [RevisionMark](#) Property

### **Usage**

**Word Pro: RowContainer class**

The container object for a table row. This object only exists for one row at a time and only when there is a table within the focus.

**Base Classes**

BaseObject\BaseContainer

**Derived Classes**

None

**Contained by**

Not contained in a property of any object.

**Usage**

The primary use for a RowContainer object is to provide quick and easy access to the RowLayout object for the currently active row. A RowContainer object always represents the row that currently has the focus. Thus, if you assign a RowContainer object to a variable, you can use that variable to access the currently active row. However, you must remember that the row referenced by the variable will change as the focus moves from one row to another. This is because the variable references the RowContainer object, and the RowContainer object always represents the row that has the focus. If there is no row within the focus, there is no RowContainer object. Thus, a variable that stores a RowContainer object will have a null value whenever the focus does not contain a row. There is never more than one RowContainer object at any given time.

For more information about container objects, see [BaseContainer](#).

**Word Pro: RowLayoutCollection class**

A collection of row layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Rows Property

**Usage**

Use this collection to access any of the row layout objects in the foundry of a specific division.

## **Word Pro: RowLayout class**

The layout for a row in a table object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

BaseTable in the CurrentRow Property

WPApplication in the CurrentRow Property

### **Usage**

## **Word Pro: RubyContainer class**

Not implemented in the US English version of Word Pro.

### **Base Classes**

BaseObject\BaseContainer\FrameContainer

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: RubyLayoutCollection class**

A collection of ruby layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the RubyLayouts Property

**Usage**

Use this collection to access any of the ruby layout objects in the foundry of a specific division.



## **Word Pro: RubyLayout class**

The ruby layout for a ruby object in a division.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: RubyMarker class**

A marker for a ruby object in a division.

### **Base Classes**

BaseObjectMarker

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: Ruler class**

A horizontal or vertical ruler object that indicates tab settings, indents, margins, and columns.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the HorzRuler Property

ApplicationWindow in the VertRuler Property

### **Usage**

## **Word Pro: ScriptDataSetCollection class**

A collection of script data sets in the ScriptDataSet class. This collection is currently not being used in the object model.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

[WPDataSetCollection](#)

### **Contained by**

### **Usage**

## **Word Pro: ScriptDataSet class**

A virtual base class that holds a collection of variables. Each variable has a name and a string value.

### **Base Classes**

BaseObject

### **Derived Classes**

WPDataSet

### **Contained by**

### **Usage**

## **Word Pro: Script class**

The object oriented programming language for Word Pro.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the Script Property

### **Usage**

**Word Pro: SectionCollection class**

A collection of section objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Sections Property

**Usage**

Use this collection to access any of the section objects in the foundry of a specific division.

**Word Pro: SectionTabs class**

The divider tabs used to indicate a section or a division.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

[ApplicationWindow](#) in the [SectionTabs](#) Property

**Usage**

Use the methods and properties of this class to create or modify a section or division object. You cannot use this class to delete a section or division object.



## **Word Pro: Section class**

Sections are areas of text in a document that reside within divisions.

### **Base Classes**

BaseObject

### **Derived Classes**

[IndexSection](#)

### **Contained by**

### **Usage**

## **Word Pro: SetTabsDialog class**

A dialog box used to set tabs in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the SetTabsDialog Property

### **Usage**

## **Word Pro: Shadow class**

The setting of shadow depth for text, frames, tables, headers, footers, and pages in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Layout in the Shadow Property

ParagraphBorder in the Shadow Property

### **Usage**

You can use the properties in this class to set the color or the depth of a shadow for text, frames, tables, headers, footers, and pages in a document.

**Word Pro: SilverBulletCollection class**

A collection of silver bullet objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the SilverBullets Property

**Usage**

Use this collection to access any of the silver bullet objects in the foundry of a specific division.

**Word Pro: SilverBullet class**

A silver bullet object in a document.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

Bullet in the SilverBullet Property

**Usage**

**Word Pro: SmartCorrect class**

SmartCorrect is a feature of Spell Check that instantly corrects errors as you type them.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

WPAApplication in the SmartCorrect Property

**Usage**

## **Word Pro: SortKey class**

This class allows you to set the sort options for each of the sort levels.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

SortOptions in the SortLevel1 Property

SortOptions in the SortLevel3 Property

SortOptions in the SortLevel2 Property

### **Usage**

You can use the properties in this class to specify the field or column that you want to sort, and in what order you want the data to sort.

## **Word Pro: SortOptions class**

The sort options displayed in the Sort dialog box.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the SortOptions Property

TextDocument in the SortOptions Property

### **Usage**



## **Word Pro: Spacing class**

The spacing of objects in Word Pro, such as text, tabs, columns, and so on.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [Spacing](#) Property

[Formula](#) in the [Spacing](#) Property

[ParagraphStyle](#) in the [Spacing](#) Property

[Text](#) in the [Spacing](#) Property

[TextMarker](#) in the [Spacing](#) Property

### **Usage**

**Word Pro: StatusBarButtonCollection class**

A collection of status bar button objects in the StatusBar class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

StatusBar in the StatusBarButtons Property

**Usage**

Use this collection to access any of the status bar button objects in the StatusBarButton class.

**Word Pro: StatusBarButton class**

A button on the status bar object in the Word Pro application.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

ApplicationWindow in the StatusBar Property

**Usage**

Use this class to set options for individual buttons located on the status bar.

## **Word Pro: StatusBar class**

The status bar object in the Word Pro application, visible at the bottom of the workspace.

### **Base Classes**

BaseObjectWindow

### **Derived Classes**

None

### **Contained by**

ApplicationWindow in the StatusBar Property

### **Usage**

When Word Pro is opened, the status bar also opens at the bottom of the workspace. You can view the status bar whenever Word Pro opens, but you cannot use it until you open a document.

## Word Pro: StringCollection class

A collection of strings.

### Base Classes

BaseObject\BaseCollection

### Derived Classes

None

### Contained by

[BaseTable](#) in the [RowLayouts](#) Property

[BaseTable](#) in the [ColumnLayouts](#) Property

[BaseTable](#) in the [CellLayouts](#) Property

[BaseTable](#) in the [ContentLayouts](#) Property

[Division](#) in the [DivisionNames](#) Property

[Filter](#) in the [TableExports](#) Property

[Filter](#) in the [TextandTableExports](#) Property

[Filter](#) in the [GraphicImports](#) Property

[Filter](#) in the [TextandTableImports](#) Property

[Filter](#) in the [GraphicExports](#) Property

[Filter](#) in the [TableImports](#) Property

[IconBarManager](#) in the [IconBarSets](#) Property

[Preferences](#) in the [SnapShotSaveOptions](#) Property

[PrintManager](#) in the [FaceNames](#) Property

[PrintManager](#) in the [PaperNames](#) Property

[PrintManager](#) in the [BinNames](#) Property

[PrintSettings](#) in the [Divisions](#) Property

[RevisionDisplay](#) in the [UndoLevels](#) Property

[ScriptDataSet](#) in the [DataNames](#) Property

[TextDocument](#) in the [DivisionNames](#) Property

[UserInterfacePrefs](#) in the [StylePaths](#) Property

[UserInterfacePrefs](#) in the [DocumentPaths](#) Property

[UserInterfacePrefs](#) in the [UserDictionaryPaths](#) Property

[UserInterfacePrefs](#) in the [BackupPaths](#) Property

[UserInterfacePrefs](#) in the [UserDictionaryFiles](#) Property

[UserInterfacePrefs](#) in the [Units](#) Property

[UserInterfacePrefs](#) in the [IconPaths](#) Property

[UserInterfacePrefs](#) in the [MacroPaths](#) Property

[UserInterfacePrefs](#) in the [GlossaryDataPaths](#) Property

[UserInterfacePrefs](#) in the [GlossaryDataFiles](#) Property

[Version](#) in the [Editors](#) Property

### Usage

Use this collection to access any of the string objects in the String class.

**Word Pro: SubPageContainer class**

The container object for sub pages. This object only exists for one sub page at a time and only when there is a page within the focus.

**Base Classes**

BaseObject\BaseContainer\PageContainer

**Derived Classes**

None

**Contained by**

Not contained in the property of any other object.

**Usage**

The primary use for a SubPageContainer object is to provide quick and easy access to the PageLayout object for the currently active sub page. A SubPageContainer object always represents the sub page that currently has the focus. Thus, if you assign a SubPageContainer object to a variable, you can use that variable to access the currently active sub page. However, you must remember that the sub page referenced by the variable will change as the focus moves from one sub page to another. This is because the variable references the SubPageContainer object, and the SubPageContainer object always represents the sub page that has the focus. If there is no sub page within the focus, there is no SubPageContainer object. Thus, a variable that stores a SubPageContainer object will have a null value whenever the focus does not contain a sub page. There is never more than one SubPageContainer object at any given time.

For more information about container objects, see [BaseContainer](#).

**Word Pro: SuperPageContainer class**

The container object for super pages. This object only exists for one super page at a time and only when there is a page within the focus.

**Base Classes**

BaseObject\BaseContainer\PageContainer

**Derived Classes**

None

**Contained by**

Not contained in the property of any other object.

**Usage**

The primary use for a SuperPageContainer object is to provide quick and easy access to the PageLayout object for the currently active super page. A SuperPageContainer object always represents the super page that currently has the focus. Thus, if you assign a SuperPageContainer object to a variable, you can use that variable to access the currently active super page. However, you must remember that the super page referenced by the variable will change as the focus moves from one super page to another. This is because the variable references the SuperPageContainer object, and the SuperPageContainer object always represents the super page that has the focus. If there is no super page within the focus, there is no SuperPageContainer object. Thus, a variable that stores a SuperPageContainer object will have a null value whenever the focus does not contain a super page. There is never more than one SuperPageContainer object at any given time.

For more information about container objects, see [BaseContainer](#).

**Word Pro: SuperTableCollection class**

A collection of super table objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the SuperTables Property

**Usage**

Use this collection to access any of the super table objects in the foundry of a specific division.



## **Word Pro: SuperTableContainer class**

The container object for super tables. This object only exists for one super table at a time and only when there is a table within the focus. When a SuperTableContainer object is present, it is stored in the SuperTableContainer property on the WPAApplication object.

To access the container object for a table, use the [TableOnlyContainer](#) property on WPAApplication.

To access the container object for parallel columns, use the [TableContainer](#) property on WPAApplication.

### **Base Classes**

BaseObject\BaseContainer

### **Derived Classes**

None

### **Contained by**

[WPAApplication](#) in the [SuperTableContainer](#) Property

### **Usage**

The primary use for a SuperTableContainer object is to provide quick and easy access to the SuperTableLayout object for the currently active super table. A SuperTableContainer object always represents the super table that currently has the focus. Thus, if you assign a SuperTableContainer object to a variable, you can use that variable to access the currently active super table. However, you must remember that the super table referenced by the variable will change as the focus moves from one super table to another. This is because the variable references the SuperTableContainer object, and the SuperTableContainer object always represents the super table that has the focus. If there is no super table within the focus, there is no SuperTableContainer object. Thus, a variable that stores a SuperTableContainer object will have a null value whenever the focus does not contain a super table. There is never more than one SuperTableContainer object at any given time.

For more information about container objects, see [BaseContainer](#).

## **Word Pro: SuperTableGroupLayout class**

A super table layout for a group of super tables.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: SuperTableLayoutCollection class**

A collection of super table layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the SuperTableLayouts Property

**Usage**

Use this collection to access any of the super table layout objects in the foundry of a specific division.

## **Word Pro: SuperTableLayout class**

The layout for a super table object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: SuperTable class**

A super table object in a document.

### **Base Classes**

BaseObject\Content

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: TableCollection class**

A collection of base table objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by****Usage**

Use this collection to access any of the base table objects in the foundry of a specific division.

## **Word Pro: TableContainer class**

An abstract container class that provides the basis for both the ParallelColsContainer and the TableOnlyCont classes. This class is used as the data type for the TableContainer property on WPAApplication. This allows the TableContainer property to contain either TableOnlyCont or ParallelColsContainer objects.

For more information on container classes, see [BaseContainer](#).

To access the container object for a table, use the [TableOnlyContainer](#) property on WPAApplication.

To access the container object for parallel columns, use the [TableContainer](#) property on WPAApplication.

### **Base Classes**

BaseObject\BaseContainer

### **Derived Classes**

[ParallelColsContainer](#)

[TableOnlyCont](#)

### **Contained by**

[WPAApplication](#) in the [TableContainer](#) Property

### **Usage**

Do not use this class for instantiating objects. You may use this class as the data type for a variable in which you wish to store objects created from either of this class' derived classes (TableOnlyCont or ParallelColsContainer).

## **Word Pro: TableFill class**

This class allows you to modify appearance properties of selected cell objects within a table.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

BaseTable in the TableFill Property

### **Usage**

The TableFill class gives you access to background and fill style information for table cells. It allows you to fill selected cells, or certain



**Word Pro: TableHeadingCollection class**

A collection of table heading objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the TableHeadings Property

**Usage**

Use this collection to access any of the table heading objects in the foundry of a specific division.

**Word Pro: TableHeadingLayoutCollection class**

A collection of table heading layout objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the TableHeadingLayouts Property

**Usage**

Use this collection to access any of the table heading layout objects in the foundry of a specific division.

## **Word Pro: TableHeadingLayout class**

The layout for a table heading in a division.

### **Base Classes**

BaseObject\Layout\TableLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: TableHeading class**

A heading for a table object.

### **Base Classes**

BaseObject\Content\BaseTable

### **Derived Classes**

None

### **Contained by**

### **Usage**

## **Word Pro: TableLayoutCollection class**

A collection of table layout objects in the foundry of a specific division.

### **Base Classes**

BaseObject\BaseCollection

### **Derived Classes**

None

### **Contained by**

Foundry in the TableLayouts Property

Foundry in the TableStyles Property

### **Usage**

Use this collection to access any of the table layout objects within the foundry of a specific division.

## **Word Pro: TableLayout class**

The layout for a table object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

[EndnoteLayout](#)

[FootnoteLayout](#)

[TableHeadingLayout](#)

### **Contained by**

### **Usage**

**Word Pro: TableLine class**

A borderline, outline, or diagonal line in a table object.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

BaseTable in the TableLine Property

**Usage**

Use this class to set or modify borderline, outline, or diagonal line styles in a table object.

**Word Pro: TableMarkerCollection class**

A collection of table marker objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the TableMarkers Property

**Usage**

Use this collection to access any of the table marker objects in the foundry of a specific division.



**Word Pro: TableMarker class**

A marker for a table object in a division.

**Base Classes**

BaseObjectMarker

**Derived Classes**

None

**Contained by****Usage**

**Word Pro: TableOnlyCollection class**

A collection of table only objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Tables Property

**Usage**

Use this collection to access any of the table only objects in the foundry of a specific division.

## **Word Pro: TableOnlyCont class**

The container object for tables. This object only exists for one table at a time and only when there is a table within the focus. When a TableOnlyCont object is present, it is stored in the TableOnlyContainer property on the WPAApplication object.

### **Base Classes**

BaseObject\BaseContainer\TableContainer

### **Derived Classes**

None

### **Contained by**

WPAApplication in the TableOnlyContainer Property

### **Usage**

The primary use for a TableOnlyCont object is to provide quick and easy access to the TableLayout object for the currently active table. A TableOnlyCont object always represents the table that currently has the focus. Thus, if you assign a TableOnlyCont object to a variable, you can use that variable to access the currently active table. However, you must remember that the table referenced by the variable will change as the focus moves from one table to another. This is because the variable references the TableOnlyCont object, and the TableOnlyCont object always represents the table that has the focus. If there is no table within the focus, there is no TableOnlyCont object. Thus, a variable that stores a TableOnlyCont object will have a null value whenever the focus does not contain a table. There is never more than one TableOnlyCont object at any given time.

The container classes for tables and parallel columns are derived from the same abstract class: TableContainer. However, only the table container object has its own property on WPAApplication. The container object for parallel columns, when it is present, is stored in the TableContainer property on WPAApplication.

For more information about container objects, see BaseContainer.

**Word Pro: Table class**

A table object in a document.

**Base Classes**

BaseObject\Content\BaseTable

**Derived Classes**

None

**Contained by**

WPAApplication in the Table Property

**Usage**

## **Word Pro: TabRack class**

Acts as the tab parent class for all tab objects in a document.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ClickHere](#) in the [TabRack](#) Property

[Layout](#) in the [TabRack](#) Property

[ParagraphStyle](#) in the [TabRack](#) Property

[Text](#) in the [TabRack](#) Property

[TextMarker](#) in the [TabRack](#) Property

### **Usage**

**Word Pro: TextCollection class**

A collection of text objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the Texts Property

**Usage**

Use this collection to access any of the text objects in the foundry of a specific division.

## Word Pro: TextDocument class

A Word Pro document, including all its divisions, sections, versions, pages, text objects, frames, tables, styles, print settings, scripts, and so on.

### Base Classes

BaseObject\Document

### Derived Classes

None

### Contained by

Application in the ActiveDocument Property

ApplicationWindow in the ActiveDocument Property

Division in the Master Property

DocWindow in the Document Property

TextDocument in the Master Property

### Usage

For each Lotus application that uses document-like objects, there is a corresponding subclass which is derived from the Document class. In Word Pro, the subclass is TextDocument. For Lotus 1-2-3, the subclass is Sheet. For Lotus Notes, the subclass is Note. The OpenDocument method on the WPAApplication class enables you to open an existing Word Pro document. WPAApplication also provides CreateDocument for creating new objects from the TextDocument class. It is also possible to get a TextDocument object associated with an already open document; use the LotusScript Bind operator. To get an object created by another Lotus application, use the GetObject operation.

The Name property (inherited from BaseObject) returns the file name (without the path) of the Word Pro document.

### Properties

Changed As Bool - Has the document changed since the last save?

Embedded As Bool - Is the document OLE embedded?

FullName As String - Name and path of this document.

IsOpen As Bool - Checks to see if a document is open, but since you cannot read a closed document, this value is always True.

Location As String - Path only of this document. In the future, this may be a variant.

Path As String - Path only of this document.

PrintSettings As PrintSettings - Allows you to set different print settings for each document.

ReadOnly As Bool - Allows you to set a document to ReadOnly or ReadWrite.

Saved As Bool - Has the document ever been saved?

### Methods

Activate - Makes this document active.

Close - Closes your document. All parameters are LOI except the CloseFile parameter which = If closing this and it is the last one, do you want a new untitled document to appear? CloseFile enum.

CopySelection - Copy to Clipboard.

CutSelection - Cut to Clipboard.

Paste - Pastes Clipboard. The MakeVisible parameter allows you to bring what is pasted into view when Paste occurs.

Print - LOI parameters; NoDialog added which inhibits the progress box.

PrintOut - Same as Print but exists for VB users.

Save - Saves the document.

SaveAs - Saves as another file type. Most parameters are LOI. AddToLastFileOpen list (recent files) added.

SaveCopyAs allows you to save a copy of an OLE launched object.

### Events

PreClose - Raised prior to closing.

Save - Pre-Save event.

SaveAs - Pre-SaveAs.

Saved - After Save.

SavedAs - After SaveAs.

Opened - After opening.

Created - After creating.

PrePrint - Before printing.

Keystroke WMCommand - Raised when a menu item is picked, a pre-defined icon is clicked, or certain push buttons cause another dialog box to display from a dialog box. This event passes the following information to the script event handler:

- menu ID
- icon ID
- ID of the menu command, WPBITMSK.LSS, containing all menu IDs



**Word Pro: TextMarkerCollection class**

A collection of text marker objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the TextMarkers Property

**Usage**

Use this collection to access any of the text marker objects in the foundry of a specific division.

**Word Pro: TextMarker class**

Marks an insertion point or a selection of text. Derives from Marker. A hidden object used to attach some data or functionality to text in the document.

**Base Classes**

BaseObjectMarker

**Derived Classes**

None

**Contained by****Usage**

**Word Pro: TextStyleCollection class**

A collection of paragraph and character style objects in the foundry of a specific division.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

Foundry in the TextStyles Property

**Usage**

Use this collection to access any of the text style objects in the foundry of a specific division.

## **Word Pro: Text class**

A text object in a document.

### **Base Classes**

BaseObject\Content

### **Derived Classes**

None

### **Contained by**

[Bullet](#) in the [Text](#) Property

[ClickHere](#) in the [Prompt](#) Property

[DivisionInfo](#) in the [FillerPageText](#) Property

[Graphic](#) in the [Text](#) Property

[NoteLayout](#) in the [Text](#) Property

[SilverBullet](#) in the [Text](#) Property

[WPApplication](#) in the [Text](#) Property

### **Usage**

## **Word Pro: TOCSuperTableLayout class**

A layout for a table of contents super table object.

### **Base Classes**

BaseObjectLayout

### **Derived Classes**

None

### **Contained by**

### **Usage**

**Word Pro: UnitCollection class**

A collection of Long integers. This collection is currently not being used in the object model.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by****Usage**

## **Word Pro: UserInterfacePrefs class**

The user interface preferences in Word Pro.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

[ApplicationWindow](#) in the [UserInterfacePrefs](#) Property

### **Usage**

You can use this class to set options in the Word Pro Preferences dialog boxes, including default paths, default directories, load options, display options, SpellCheck options, and so on.

## **Word Pro: UseWhen class**

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Layout in the UseWhen Property

### **Usage**



**Word Pro: VersionCollection class**

A collection of version objects in the VersionManager class.

**Base Classes**

BaseObject\BaseCollection

**Derived Classes**

None

**Contained by**

VersionManager in the Versions Property

**Usage**

Use this collection to access any of the version objects in the VersionManager class.

## **Word Pro: VersionManager class**

A tool to manage the Version tool in a division.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

Division in the VersionManager Property

TextDocument in the VersionManager Property

### **Usage**

**Word Pro: Version class**

The Version tool in the Word Pro application.

**Base Classes**

BaseObject

**Derived Classes**

None

**Contained by**

VersionManager in the CurrentVersion Property

**Usage**

## **Word Pro: Window class**

A window in the Word Pro application.

### **Base Classes**

BaseObject

### **Derived Classes**

[ApplicationWindow](#)

[DocWindow](#)

[IconBarManager](#)

[StatusBar](#)

### **Contained by**

### **Usage**

## **Word Pro: WinViewPrefs class**

The window view preferences in Word Pro.

### **Base Classes**

BaseObject

### **Derived Classes**

None

### **Contained by**

DocWindow in the WinViewPrefs Property

UserInterfacePrefs in the WinViewPrefs Property

### **Usage**

## Word Pro: WPAApplication class

The whole of the Word Pro application, including the application engine, workspace, and any documents created by the application. You can access any part of Word Pro or a Word Pro document through the members of the WPAApplication object.

### Base Classes

BaseObject\Application

### Derived Classes

None

### Contained by

BaseObject in the Application Property

### Usage

Each time you launch Word Pro, a single object is instantiated from the WPAApplication class. That object represents the Word Pro application. Under normal circumstances, there will be only one WPAApplication object active at any given time. WPAApplication gives you access to all the Word Pro-specific objects, features, events, and information which exist as part of Word Pro. You will use the WPAApplication object to access and control nearly every aspect of Word Pro and its components.

### Static and Current Context Properties

WPAApplication is unique in that it defines two types of properties: static and current context. The static properties, such as Name and AppViewPrefs, represent things which apply to the Word Pro application as a whole. Their contents are not dependent on which document or other object is active. The contents of static properties remain the same, regardless of which document may be active or where the focus of the application is directed.

The content of current context properties changes as the focus of Word Pro moves from one document or division to another. These properties are called current context properties because their contents depend on the current context of Word Pro. For example, in a document with two divisions named ChapterOne and ChapterTwo, the focus of Word Pro can move from one division to the other. While the focus is in the ChapterOne division, the current context of Word Pro is the ChapterOne Division object, and the Division property on WPAApplication contains the ChapterOne Division object, as long as the ChapterOne division is active. However, when you move the focus of Word Pro into the division named ChapterTwo, the current context changes and the contents of the Division property changes to the ChapterTwo Division object. It is important for you to keep current context in mind whenever you access the current context properties on the WPAApplication object.

The current context properties include:

ActiveDocument  
ActiveDocWindow  
ApplicationWindow  
BaseTable  
Cell  
Container  
Content  
CurrentCell  
CurrentColumn  
CurrentRow  
Division  
Divisions  
Documents  
Foundry  
Frame  
Graphic  
GraphicOleObject  
Layout  
OleObject  
Page  
ParallelColumns  
SuperTableContainer  
Table  
TableContainer

TableOnlyContainer  
Text

### The Abstract Application Class

Each Lotus application is represented in LotusScript by its own unique application object. Each application object is instantiated from a unique application class. But all application classes share a common abstract application class, called Application. This abstract class provides a common fundamental definition for all application objects. However, each Lotus application uses the abstract Application class as a starting point, and then defines additional properties, methods, and events as needed to create its own unique application class. In this way, each Lotus application class shares a common set of class members from the abstract class, while still providing the additional class members needed to represent the unique components of each application. The table below shows the name of each product's application class (each derived from the same abstract Application class), the name of the object instantiated from that application class, and the name of the global variable used for accessing that object.

Product Name	Application Class	Object Name	Global Variable
Word Pro	WPApplication	WordPro	CurrentApplication
Lotus 1-2-3	Application	1-2-3	CurrentApplication
Approach	Application	Approach	CurrentApplication
Freelance	Application	Freelance	

When referring to the WPApplication object from within a Word Pro script, you should use the global variable, CurrentApplication.

When referring to the WPApplication object from outside of Word Pro, you should use the GetObject function with the name, "WordPro."

### LotusScript and Multiple Sessions Of Word Pro

A single object (named WordPro) is instantiated from the WPApplication class each time you launch the Word Pro application. If you run multiple sessions of Word Pro simultaneously, each session of Word Pro is represented by its own WPApplication object. You can run multiple sessions of Word Pro and use LotusScript to manipulate the different instances of WPApplication. However, the memory requirements for running multiple sessions are quite high and great care must be taken to identify correctly the different WPApplication objects. A single error in managing multiple WPApplication objects could result in the loss of valuable data or unexpected changes in the default settings within Word Pro.

**Note** The documentation for the Word Pro LotusScript object model assumes that you never have more than one Word Pro session running at any time. While it is possible to run multiple Word Pro sessions and use LotusScript to manipulate them, it is not recommended.

If you do write scripts to control multiple sessions of Word Pro, you should keep in mind the following:

- The WPApplication object which is active when your script starts running is the "native" WPApplication object. All other WPApplication objects must be accessed through OLE automation, using the LotusScript GetObject or CreateObject functions.
- Word Pro reads default settings from .INI files as those default settings are used within a Word Pro session. For example, when the user opens the Set Tabs dialog box, Word Pro checks an .INI file for the current tab settings and displays those settings in the dialog box.
- Word Pro writes any changes to a default setting into an .INI file when the Word Pro session ends. For example, if the user changes the type of leader dot in the Set Tabs dialog box, that change will be written into an .INI file when the user ends the Word Pro session.
- Changes made to an .INI file by one Word Pro session may affect other simultaneous Word Pro sessions, if the first session ends and the other sessions make use of the settings changed in the first session. This is due to the fact that the other sessions read from the same .INI file as the first session and will reflect any changes made to the .INI file when the first session ended.

For example, while running two sessions of Word Pro simultaneously, you can change the value of the LeaderDotType property on the AppViewPrefs object in one session of Word Pro. You can then end that session, at which time the appropriate .INI file will be updated with the new LeaderDotType value. You can then open the Set Tabs dialog box from within the second session and see the new LeaderDotType value reflected in the dialog box settings. This could result in some unpredictable behavior in multiple sessions of Word Pro.

## **Word Pro: WPDataSetCollection class**

A collection of Word Pro data sets in the Word Pro application object, a document object or some object within a document.

### **Base Classes**

BaseObject\BaseCollection\ScriptDataSetCollection

### **Derived Classes**

None

### **Contained by**

[CharacterStyle](#) in the [WPDataSets](#) Property

[ClickHere](#) in the [WPDataSets](#) Property

[Division](#) in the [WPDataSets](#) Property

[Layout](#) in the [WPDataSets](#) Property

[Marker](#) in the [WPDataSets](#) Property

[ParagraphStyle](#) in the [WPDataSets](#) Property

[TextDocument](#) in the [WPDataSets](#) Property

[Text](#) in the [WPDataSets](#) Property

[TextMarker](#) in the [WPDataSets](#) Property

[WPApplication](#) in the [WPDataSets](#) Property

### **Usage**

Item ScriptDataSet is not implemented.

Use the WPDataSetCollection class to create and manage sets of variables attached to Word Pro objects.



**Word Pro: WPDataSet class**

The Word Pro data set that holds a collection of variables. Each variable has a name and a string value.

**Base Classes**

BaseObject\ScriptDataSet

**Derived Classes**

None

**Contained by****Usage**

Each WPDataSet can be attached to the following Word Pro objects: TextDocument, Layout, Marker, CharacterStyle, ParagraphStyle, WPAApplication, Division, and Text classes. If you close a document that has a data set attached to it, Word Pro saves the data set(s) with the document. When you reopen the document, Word Pro reads the data sets back in from the document.

## Word Pro: ParallelColumns class members

### Properties

Application AS WPAApplication  
CanEmbed AS Integer (Boolean)  
CellLayouts AS StringCollection  
ClassName AS String  
ColumnLayouts AS StringCollection  
ContentLayouts AS StringCollection  
ContentType AS ContentType  
CurrentCell AS CellLayout  
CurrentColumn AS Layout  
CurrentRow AS RowLayout  
DefaultLeftColumnStyleName AS String  
DefaultRightColumnStyleName AS String  
DefCellStyleName AS String  
DefColWidth AS Long (measured in Twips)  
DefRowHeight AS Long (measured in Twips)  
Description AS String  
EndingColOfSelection AS Integer  
EndingRowOfSelection AS Integer  
IsAutoGrow AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsParagraphNumberingDown AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsResetParagraphNumber AS Integer (Boolean)  
IsSizingViaMouse AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxBottomBorder AS Long (measured in Twips)  
MaxBottomGutter AS Long (measured in Twips)  
MaxLeftBorder AS Long (measured in Twips)  
MaxLeftGutter AS Long (measured in Twips)  
MaxNumColsAllowed AS Integer  
MaxNumRowsAllowed AS Integer  
MaxRightBorder AS Long (measured in Twips)  
MaxRightGutter AS Long (measured in Twips)  
MaxSplitCols AS Integer  
MaxSplitRows AS Integer  
MaxTopBorder AS Long (measured in Twips)  
MaxTopGutter AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
Parent AS BaseObject  
RowLayouts AS StringCollection  
SelectionType AS SelectionType

SingleCellSelected AS Integer (Boolean)

StartingColOfSelection AS Integer

StartingColStringOfSelection AS String

StartingRowOfSelection AS Integer

TableFill AS TableFill

TableLine AS TableLine

VersionID AS Long

### **Methods**

CellLayout

Connect

Copy

DeleteTable

DisconnectCells

DoesMarkerNameMatch

FindCellLayout

GetMarkerName

GoToTableCell

InsertRowOrColumn

Mark

NextItem

PreviousItem

SelectTableItem

Split

### **Events**

None

## **Word Pro: PowerFieldCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: PowerField class members**

### **Properties**

Application AS WPApplication  
ConvertOnNew AS Integer (Boolean)  
Description AS String  
DivisionInfo AS DivisionInfo  
Formula AS String  
HideFormula AS Integer (Boolean)  
IsChanged AS Integer (Boolean)  
IsMarkerValid AS Integer (Boolean)  
IsRegistered AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
LockResult AS Integer (Boolean)  
MarkerClass AS MarkerType  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
PageNumber AS Integer  
PageNumberAsText AS String  
Parent AS BaseObject  
Private AS Integer (Boolean)  
Result AS String  
StartColumns AS Integer  
StartRow AS Integer  
StateID AS Long  
Type AS ;\_  
UpdateOnLoad AS Integer (Boolean)  
UpdateOnLoadImmediate AS Integer (Boolean)  
VersionID AS Long  
WPDataSets AS WPDataSetCollection

### **Methods**

DeleteContents  
DeleteMarker  
ExtractText  
GetContents  
GetMarkedText  
GetNamedProperty  
GetParagraphNumber  
GoTo  
HasNamedProperty  
InsertMarker  
RegisterWPDataSet  
RemoveNamedProperty  
SetFieldFormula  
SetNamedProperty

UnregisterWPDataSet

Update

**Events**

None

## Word Pro: Preferences class members

### Properties

Address1 AS String  
Address2 AS String  
Application AS WPApplication  
BackgroundPrintingOn AS Integer (Boolean)  
BackgroundSpellingOn AS Integer (Boolean)  
CellStyleName AS String  
ChangeKeyboardToLanguage AS Integer (Boolean)  
ChangeTextToMatchkeyboard AS Integer (Boolean)  
CharacterSet AS CharacterSet  
CityState AS String  
Company AS String  
DdeEnabled AS Integer (Boolean)  
DefaultCellStyleDescription AS String  
DefaultColumnName AS String  
DefaultDropCapStyleDescription AS String  
DefaultFrameStyleDescription AS String  
DefaultLatinFont AS String  
DefaultLeftColumnName AS String  
DefaultNonLatinFont AS String  
DefaultPageStyleDescription AS String  
DefaultRightColumnName AS String  
DefaultTableStyleDescription AS String  
DefaultTextStyleDescription AS String  
Description AS String  
DragDropOn AS Integer (Boolean)  
DropCapStyleName AS String  
EMail AS String  
FaxNumber AS String  
FontMatching AS Integer (Boolean)  
FontName AS String  
FrameStyleName AS String  
HiLiteColor AS Color  
IndexPrimaryStyleName AS String  
IndexSecondaryStyleName AS String  
IndexSeparatorStyleName AS String  
InsertionMode AS Integer (Boolean)  
IsContents AS Integer  
IsSmartCorrectEnabled AS Integer (Boolean)  
IsSmartShadeEnabled AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
LinkGraphic AS Integer (Boolean)  
MouseButtonForManipulation AS MouseButton  
MouseButtonForSelection AS MouseButton  
MouseProperty AS MouseButton

MultiCellPasteOn AS Integer (Boolean)  
Name AS String  
NumberSequenceName AS String  
NumUndoLevels AS Integer  
OLEAutomation AS Integer (Boolean)  
OLEEnabled AS Integer (Boolean)  
PageStyleName AS String  
Parent AS BaseObject  
PersonalData1 AS String  
PersonalData2 AS String  
PersonalData3 AS String  
PersonalData4 AS String  
PhoneNumber AS String  
PointSize AS Points  
RevisionDisplay AS RevisionDisplay  
SaveSnapShot AS SnapShot  
SmallFileFormat AS Integer (Boolean)  
SnapShotSaveOptions AS StringCollection  
TableStyleName AS String  
TextStyleName AS String  
Title AS String  
UserInitials AS String  
UserName AS String  
VersionID AS Long  
ZipCode AS String

**Methods**

None

**Events**

None



## **Word Pro: Presentation class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

OutlineLevel AS Integer

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: PrintManager class members**

### **Properties**

Application AS WPAApplication  
BinNames AS StringCollection  
CanWePrint AS Integer (Boolean)  
DefaultBinName AS String  
DefaultPageHeight AS Long (measured in Twips)  
DefaultPageWidth AS Long (measured in Twips)  
Description AS String  
DriverName AS String  
FaceNames AS StringCollection  
IsValid AS Integer (Boolean)  
Name AS String  
PaperNames AS StringCollection  
Parent AS BaseObject  
PrintDestination AS String  
PrinterName AS String  
QueueName AS String  
UseDefaultPrinter AS Integer (Boolean)  
VersionID AS Long

### **Methods**

BinNameFromNumber  
Cancel  
EndPrinting  
GetDefaultPageSize  
GetEnvelopeDefaults  
GetFontType  
GetPrinterInfo  
Print  
Query  
ResetPrinting  
UpdatePageSizeChange  
UpdatePrinterBins  
UpdatePrinterChanges

### **Events**

None

## **Word Pro: PrintSettings class members**

### **Properties**

Application AS WPApplication  
BookletPrinting AS Integer (Boolean)  
Collate AS Integer (Boolean)  
Copies AS Integer  
Crop AS Integer (Boolean)  
Description AS String  
Divisions AS StringCollection  
DuplexType AS DuplexType  
IsCentered AS Integer (Boolean)  
IsPrePrintedForm AS Integer (Boolean)  
IsPrintClickHereBlocks AS Integer (Boolean)  
IsUpdateIndex AS Integer (Boolean)  
IsUpdateTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Name AS String  
NumCols AS Integer  
NumRowsToFit AS Integer  
OutputToFile AS Integer (Boolean)  
PageOrder AS PageOrder  
Parent AS BaseObject  
PrintDocDescription AS Integer (Boolean)  
PrintGraphics AS Integer  
PrintInBackground AS Integer (Boolean)  
PrintPagesFrom AS Integer  
PrintPagesTo AS Integer  
PrintPageType AS PrintPage  
PrintRange AS PrintRange  
SelectedPages AS String  
UpdateFields AS Integer (Boolean)  
VersionID AS Long  
VersionRemarks AS Integer (Boolean)  
ViewLevel AS Integer  
ViewType AS PresentationType  
WithComments AS Integer (Boolean)

### **Methods**

AddDivisionToPrint  
ClearDivisionList

### **Events**

None

## **Word Pro: RelativeIndent class members**

### **Properties**

All AS Long (measured in Twips)

Application AS WPAApplication

BodyOnly AS Integer (Boolean)

Description AS String

First AS Long (measured in Twips)

Hang AS Integer

IsBothSidesEqual AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

Relative AS IndentProperty

Rest AS Long (measured in Twips)

Right AS Long (measured in Twips)

UseRelative AS Integer (Boolean)

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: ReviewVersions class members**

### **Properties**

Application AS WPApplication

Description AS String

FilesToCompare AS String

IsValid AS Integer (Boolean)

Name AS String

NewFile AS Integer (Boolean)

NewVersion AS Integer (Boolean)

Parent AS BaseObject

VersionID AS Long

### **Methods**

ReviewVersions

### **Events**

None

## **Word Pro: RevisionDisplay class members**

### **Properties**

Application AS WPAApplication

DeleteFont AS Font

Description AS String

InsertFont AS Font

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

RevMarkCharacter AS Integer

RevMarkPosition AS MarkPosition

RevMarkType AS MarkType

TextAttributes AS Attributes

UndoLevels AS StringCollection

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Revision class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

RevisionType AS RevisionSelectType

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: RowContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide



IsPointWithin

LinkContainers

ShowContainers

Start

UnLinkContainers

**Events**

None

## Word Pro: RowGroupLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: RowLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: RowLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)



IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: RubyContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: RubyLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: RubyLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
Alignment AS Integer  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)



IsBreakable AS Integer (Boolean)  
IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout

LeftTopCellId AS Integer  
LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
Placement AS Integer  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)

TopLeftCellRowId AS Integer  
UseFooter AS Integer  
UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

#### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

**Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: RubyMarker class members**

### **Properties**

Application AS WPAApplication  
Description AS String  
DivisionInfo AS DivisionInfo  
IsChanged AS Integer (Boolean)  
IsMarkerValid AS Integer (Boolean)  
IsRegistered AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MarkerClass AS MarkerType  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
PageNumber AS Integer  
PageNumberAsText AS String  
Parent AS BaseObject  
StartColumns AS Integer  
StartRow AS Integer  
StateID AS Long  
VersionID AS Long  
WPDataSets AS WPDataSetCollection

### **Methods**

DeleteContents  
DeleteMarker  
GetContents  
GetMarkedText  
GetNamedProperty  
GetParagraphNumber  
GoTo  
HasNamedProperty  
InsertMarker  
RegisterWPDataSet  
RemoveNamedProperty  
SetNamedProperty  
UnregisterWPDataSet

### **Events**

None

## **Word Pro: Ruler class members**

### **Properties**

Application AS WPApplication

Description AS String

IsShowing AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

SelectTab AS Integer

VersionID AS Long

### **Methods**

DeselectRuler

SelectRuler

### **Events**

None

## **Word Pro: ScriptDataSetCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: ScriptDataSet class members**

### **Properties**

Application AS WPApplication

DataNames AS StringCollection

Description AS String

IsPersistent AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

GetData

IsDataNameUsed

SetData

### **Events**

None



## **Word Pro: Script class members**

### **Properties**

Application AS WPAApplication

CommentColor AS Color

Description AS String

DirectiveColor AS Color

ErrorColor AS Color

FontName AS String

FontSize AS Points

IdentifierColor AS Color

IsValid AS Integer (Boolean)

KeywordColor AS Color

Name AS String

OpionDeclare AS Integer

Parent AS BaseObject

PrettyPrinting AS Integer

TabWidth AS Integer

UndoEnable AS Integer

UndoLevels AS Integer

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: SectionCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: SectionTabs class members**

### **Properties**

Application AS WPAApplication

Description AS String

ExternalName AS String

IsShowing AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

Activate

AddSectionTabs

ConnectSectionTabs

Contract

Expand

InsertSection

### **Events**

None

## **Word Pro: Section class members**

### **Properties**

Application AS WPAApplication

Color AS Color

Description AS String

IsIndex AS Integer (Boolean)

IsValid AS Integer (Boolean)

LayoutName AS String

Name AS String

NextSection AS String

Parent AS BaseObject

ShowTabs AS Integer (Boolean)

UserName AS String

VersionID AS Long

### **Methods**

DeleteSection

GoToSection

### **Events**

None

## **Word Pro: SetTabsDialog class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

SelectTab AS Integer

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Shadow class members**

### **Properties**

Application AS WPApplication

Color AS Color

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

XPosition AS Long (measured in Twips)

YPosition AS Long (measured in Twips)

### **Methods**

None

### **Events**

None

## **Word Pro: SilverBulletCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: SilverBullet class members**

### **Properties**

Application AS WPApplication

Description AS String

IsPrivate AS Integer

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

RestartStyleName AS String

Text AS Text

VersionID AS Long

### **Methods**

GetArrayProp

SetArrayProp

### **Events**

None



## **Word Pro: SmartCorrectCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: SmartCorrect class members**

### **Properties**

Application AS WPAApplication

Description AS String

DoInitialCaps AS Integer (Boolean)

DoSmartQuotes AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddSmartCorrect

DeleteSmartCorrect

Get

### **Events**

None

## **Word Pro: SmartFillCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: SmartFill class members**

### **Properties**

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

ListCount AS Integer

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddStringToList

CreateEmptyList

GetListName

GetString

IsCaseExact

RemoveList

### **Events**

None

## **Word Pro: SortKey class members**

### **Properties**

Application AS WPAApplication

Description AS String

FieldNumber AS Integer

IsSortFromEnd AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

SortOrder AS Sort

SortType AS SortType

SortWord AS Integer

SortWordOption AS SortWhichWord

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: SortOptions class members**

### **Properties**

Application AS WPAApplication

Description AS String

FieldDelimiter AS DelimiterType

FieldDelimiterText AS String

IsValid AS Integer (Boolean)

Name AS String

NumParagraphs AS Integer

Parent AS BaseObject

SortLevel1 AS SortKey

SortLevel2 AS SortKey

SortLevel3 AS SortKey

SortNumbers AS SortNumberOrder

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: Spacing class members**

### **Properties**

Always AS Integer (Boolean)

Amount AS Long (measured in Twips)

AmountOfSpaceAbove AS Long (measured in Twips)

AmountOfSpaceBelow AS Long (measured in Twips)

Application AS WPApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

NumLinesOfSpace AS Long (measured in Twips)

NumLinesOfSpaceAbove AS Long (measured in Twips)

NumLinesOfSpaceBelow AS Long (measured in Twips)

Parent AS BaseObject

Type AS SpacingType

TypeAbove AS SpacingType

TypeBelow AS SpacingType

VersionID AS Long

### **Methods**

RevertToStyle

### **Events**

None

## **Word Pro: StatusBarButtonCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: StatusBarButton class members**

### **Properties**

P21012;Application;WPApplication;Read Only;BaseObject

Description AS String

GetButtonId AS Long

GetButtonType AS Long

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

AddPopupGraphicItem

AddPopupPointSizeItem

AddPopupTextItem

ClearPopupData

DeleteButton

InvalidateButton

SetButtonText

SetOverrideGraphic

SetOverrideText

SetPopupAlignment

SetPopupIndex

SetPopupWidth

SetPopupWidthType

SimulateButtonClick

### **Events**

StatusBarButtonClicked

StatusBarButtonFillPopupList

StatusBarButtonItemSelected

StatusBarButtonOverrideGraphic

StatusBarButtonOverrideText

StatusBarButtonOverrideTextAndGraphic

## **Word Pro: StatusBar class members**

### **Properties**

Active AS Integer (Boolean)

Application AS WPApplication

Caption AS String

Description AS String

Height AS Long (measured in Twips)

IsValid AS Integer (Boolean)

Left AS Long (measured in Twips)

Name AS String

Parent AS BaseObject

StatusBarButtons AS StatusBarButtonCollection

Top AS Long (measured in Twips)

VersionID AS Long

Visible AS Integer (Boolean)

Width AS Long (measured in Twips)

### **Methods**

Close

CreateNewButton

GetStandardButtonId

HideStatusBar

InvalidateWholeBar

Move

Open

Resize

ShowStatusBar

Update

### **Events**

Moved

StatusBarButtonClicked

StatusBarButtonFillPopupList

StatusBarButtonOverrideText

## **Word Pro: StringCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: SubPageContainer class members

### Properties

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### Methods

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: SuperPageContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: SuperTableCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## **Word Pro: SuperTableContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

RevertToStyle

SetStyle

ShowContainers

Start

UnLinkContainers

**Events**

None

## Word Pro: SuperTableGroupLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: SuperTableLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## Word Pro: SuperTableLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: SuperTable class members**

### **Properties**

Application AS WPApplication

CanEmbed AS Integer (Boolean)

ClassName AS String

ContentType AS ContentType

Description AS String

IsEmpty AS Integer (Boolean)

IsReplaceable AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: TableCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: TableContainer class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide



IsPointWithin

LinkContainers

ShowContainers

Start

UnLinkContainers

**Events**

None

## **Word Pro: TableFill class members**

### **Properties**

Application AS WPApplication

Background AS Background

Description AS String

FillStyle AS TableFillStyle

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

SetPattern

### **Events**

None

## **Word Pro: TableHeadingCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: TableHeadingLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: TableHeadingLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout



KeyStroke

MouseDown

MouseUp

## Word Pro: TableHeading class members

### Properties

Application AS WPAApplication  
CanEmbed AS Integer (Boolean)  
CellLayouts AS StringCollection  
ClassName AS String  
ColumnLayouts AS StringCollection  
ContentLayouts AS StringCollection  
ContentType AS ContentType  
CurrentCell AS CellLayout  
CurrentColumn AS Layout  
CurrentRow AS RowLayout  
DefCellStyleName AS String  
DefColWidth AS Long (measured in Twips)  
DefRowHeight AS Long (measured in Twips)  
Description AS String  
EndingColOfSelection AS Integer  
EndingRowOfSelection AS Integer  
IsAutoGrow AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsParagraphNumberingDown AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsResetParagraphNumber AS Integer (Boolean)  
IsSizingViaMouse AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxBottomBorder AS Long (measured in Twips)  
MaxBottomGutter AS Long (measured in Twips)  
MaxLeftBorder AS Long (measured in Twips)  
MaxLeftGutter AS Long (measured in Twips)  
MaxNumColsAllowed AS Integer  
MaxNumRowsAllowed AS Integer  
MaxRightBorder AS Long (measured in Twips)  
MaxRightGutter AS Long (measured in Twips)  
MaxSplitCols AS Integer  
MaxSplitRows AS Integer  
MaxTopBorder AS Long (measured in Twips)  
MaxTopGutter AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
Parent AS BaseObject  
RowLayouts AS StringCollection  
SelectionType AS SelectionType  
SingleCellSelected AS Integer (Boolean)  
StartingColOfSelection AS Integer

StartingColStringOfSelection AS String

StartingRowOfSelection AS Integer

TableFill AS TableFill

TableLine AS TableLine

VersionID AS Long

### **Methods**

CellLayout

Connect

Copy

DeleteTable

DisconnectCells

DoesMarkerNameMatch

FindCellLayout

GetMarkerName

GoToTableCell

InsertRowOrColumn

Mark

NextItem

PreviousItem

SelectTableItem

Split

### **Events**

None

## **Word Pro: TableLayoutCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: TableLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPAApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)  
IsBreakable AS Integer (Boolean)

IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS String  
Layer AS Layout  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout  
LeftTopCellId AS Integer

LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer

UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

### **Methods**

AddChildToLayout  
Backward  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty  
RevisionAcceptLayoutChange  
RevisionCancelLayoutChange  
SetAllMargins  
SetMinimumOrigin  
SetNamedProperty  
UnregisterWPDataSet  
Update

### **Events**

EnterLayout



KeyStroke

MouseDown

MouseUp

## **Word Pro: TableLine class members**

### **Properties**

Application AS WPAApplication

BorderLines AS BorderLines

Description AS String

DiagonalLines AS BorderLines

IsValid AS Integer (Boolean)

LineMix AS TableMix

Name AS String

OutlineBorderLines AS BorderLines

Parent AS BaseObject

VersionID AS Long

### **Methods**

BeginCustomLines

ChgLineStyle

EndCustomLines

GetLineMix

GetLineStyle

SetLineOneSide

SetLinesAllSides

### **Events**

None

## **Word Pro: TableMarkerCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: TableMarker class members

### Properties

Application AS WPApplication  
Description AS String  
DivisionInfo AS DivisionInfo  
IsChanged AS Integer (Boolean)  
IsMarkerValid AS Integer (Boolean)  
IsRegistered AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MarkerClass AS MarkerType  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
PageNumber AS Integer  
PageNumberAsText AS String  
Parent AS BaseObject  
StartColumns AS Integer  
StartRow AS Integer  
StateID AS Long  
VersionID AS Long  
WPDataSets AS WPDataSetCollection

### Methods

DeleteContents  
DeleteMarker  
GetContents  
GetMarkedText  
GetNamedProperty  
GetParagraphNumber  
GoTo  
HasNamedProperty  
InsertMarker  
RegisterWPDataSet  
RemoveNamedProperty  
SetNamedProperty  
UnregisterWPDataSet

### Events

None

## **Word Pro: TableOnlyCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: TableOnlyCont class members**

### **Properties**

AbsoluteTextOrientation AS Integer  
Application AS WPApplication  
ClientHeight AS Long (measured in Twips)  
ClientWidth AS Long (measured in Twips)  
ContentHeight AS Long (measured in Twips)  
ContentName AS String  
ContentWidth AS Long (measured in Twips)  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionName AS String  
Height AS Long (measured in Twips)  
IsFooter AS Integer  
IsHeader AS Integer  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxContentHeight AS Long (measured in Twips)  
MaxContentWidth AS Long (measured in Twips)  
Name AS String  
NumContainers AS Integer  
PageNum AS Integer  
Parent AS BaseObject  
PositionXOnPage AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
Presentation AS Presentation  
RelativePageNum AS Integer  
TextOrientation AS Integer  
VersionID AS Long  
Width AS Long (measured in Twips)

### **Methods**

Abandon  
Adopt  
Anchor  
Backward  
CanHaveFootnotes  
ConnectContainer  
DeleteContainer  
Disconnect  
Ending  
FindParent  
Forward  
GetObjectList  
GetPasteFormatCategories  
GoToContainer  
Hide

IsPointWithin

LinkContainers

ShowContainers

Start

UnLinkContainers

**Events**

None

## Word Pro: Table class members

### Properties

Application AS WPAApplication  
CanEmbed AS Integer (Boolean)  
CellEngine AS CellEngine  
CellLayouts AS StringCollection  
ClassName AS String  
ColumnLayouts AS StringCollection  
ContentLayouts AS StringCollection  
ContentType AS ContentType  
CurrentCell AS CellLayout  
CurrentColumn AS Layout  
CurrentRow AS RowLayout  
DefCellStyleName AS String  
DefColWidth AS Long (measured in Twips)  
DefRowHeight AS Long (measured in Twips)  
Description AS String  
EndingColOfSelection AS Integer  
EndingRowOfSelection AS Integer  
IsAutoGrow AS Integer (Boolean)  
IsEmpty AS Integer (Boolean)  
IsParagraphNumberingDown AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsResetParagraphNumber AS Integer (Boolean)  
IsSizingViaMouse AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Layout AS Layout  
MaxBottomBorder AS Long (measured in Twips)  
MaxBottomGutter AS Long (measured in Twips)  
MaxLeftBorder AS Long (measured in Twips)  
MaxLeftGutter AS Long (measured in Twips)  
MaxNumColsAllowed AS Integer  
MaxNumRowsAllowed AS Integer  
MaxRightBorder AS Long (measured in Twips)  
MaxRightGutter AS Long (measured in Twips)  
MaxSplitCols AS Integer  
MaxSplitRows AS Integer  
MaxTopBorder AS Long (measured in Twips)  
MaxTopGutter AS Long (measured in Twips)  
Name AS String  
NumCols AS Integer  
NumRows AS Integer  
Parent AS BaseObject  
RowLayouts AS StringCollection  
SelectionType AS SelectionType  
SingleCellSelected AS Integer (Boolean)



StartingColOfSelection AS Integer  
StartingColStringOfSelection AS String  
StartingRowOfSelection AS Integer  
TableFill AS TableFill  
TableLine AS TableLine  
VersionID AS Long

**Methods**

CellLayout  
Connect  
Copy  
DeleteTable  
DisconnectCells  
DoesMarkerNameMatch  
FindCellLayout  
GetMarkerName  
GoToTableCell  
InsertRowOrColumn  
Mark  
NextItem  
PreviousItem  
SelectTableItem  
Split  
Sum

**Events**

None

## **Word Pro: TabRack class members**

### **Properties**

Application AS WPAApplication

Description AS String

IsValid AS Integer (Boolean)

Name AS String

NumTabs AS Integer

Parent AS BaseObject

VersionID AS Long

### **Methods**

ClearAll

DeleteTab

GetOne

InsertOne

RevertToStyle

### **Events**

None

## **Word Pro: TextCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: TextDocument class members

### Properties

Application AS WPAApplication  
AsciiCRLFType AS AsciiLineEnding  
AutoRunMacro AS AutoRunMacro  
BookmarkManager AS BookmarkManager  
CanCreatePreviewBitmap AS Integer (Boolean)  
Changed AS Integer (Boolean)  
CreatePreviewBitmap AS Integer (Boolean)  
DdeLinkManager AS DdeLinkManager  
DdeOutboundInfo AS Long  
DemandLoad AS ReservedParam  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionNames AS StringCollection  
DivisionOptions AS DivisionOptions  
Divisions AS DivisionCollection  
DocControl AS DocControl  
DocInfo AS DocInfo  
DocOptions AS Options  
EditorManager AS EditorManager  
EditorName AS String  
Embedded AS Integer (Boolean)  
Epoch AS String  
FileType AS String  
FinishedSpellChecking AS Integer  
FirstChild AS String  
FirstDivision AS String  
FirstName AS String  
FirstPage AS Integer  
FootnoteOptions AS FootnoteOptions  
Foundry AS Foundry  
FullName AS String  
HasIndex AS Integer (Boolean)  
HasTOC AS Integer (Boolean)  
IsAsciiCRLF AS Integer (Boolean)  
IsAsciiKeepStyle AS Integer (Boolean)  
IsChanged AS Integer (Boolean)  
IsChangedOtherThanLinkTo AS Integer (Boolean)  
IsChangedSinceTimeSave AS Integer (Boolean)  
IsChangedToLinkTo AS Integer (Boolean)  
IsDivisionExternal AS Integer (Boolean)  
IsDocLoading AS Integer (Boolean)  
IsEmptyDoc AS Integer (Boolean)  
IsLockedForRevisions AS Integer (Boolean)  
IsOpen AS Integer (Boolean)

IsProtected AS Integer  
IsSpellBarUp AS Integer (Boolean)  
IsStyleSheet AS Integer (Boolean)  
IsUndoOn AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
LastChild AS String  
LastDivision AS String  
LastName AS String  
LastPage AS Integer  
LineNumberOptions AS LineNumberOptions  
Location AS String  
Locked AS Integer  
MailRouting AS MailRouting  
Master AS TextDocument  
MergeOptions AS MergeOptions  
Name AS String  
NextName AS String  
NextNeighbor AS String  
NonUserDocument AS Integer (Boolean)  
NumPagesInDoc AS Integer  
NumWindowsViewingDoc AS Integer  
NumWordsInDoc AS Long  
OleContainerDocName AS String  
Options AS Options  
OrigFileType AS String  
Parent AS BaseObject  
ParentName AS String  
Path AS String  
PathName AS String  
PreviousName AS String  
PreviousNeighbor AS String  
PrintManager AS PrintManager  
PrintSettings AS PrintSettings  
ReadOnly AS Integer (Boolean)  
RevisionMarkMode AS Integer  
Saved AS Integer (Boolean)  
SelectionType AS SelectionType  
SortOptions AS SortOptions  
StateID AS Long  
StyleSheetFullPath AS String  
StyleSheetName AS String  
StyleSheetPath AS String  
TabSpacing AS Long (measured in Twips)  
TitleBarDocNumber AS Integer  
UseContents AS Integer  
UsedFirstPageHeight AS Long (measured in Twips)  
VersionID AS Long

VersionManager AS VersionManager  
WPDataSets AS WPDataSetCollection

## **Methods**

Activate  
AddDivision  
AdviseOnRename  
AdviseOnSave  
AnyEdits  
BeginChange  
ChangeAllEditsToEditor  
Clear  
Close  
Copy  
CopySelection  
CreateDocument  
CutSelection  
Destroy  
EndChange  
ForceDocToLoad  
FXGetNotesString  
FXGetNotesWriteHandle  
FXSetNotesString  
GetActiveList  
GetNamedProperty  
GetNameFromPage  
HasNamedProperty  
Hit  
Link  
Localize  
Move  
Paste  
PrepareToDestroy  
Present  
Print  
PrintOut  
Purge  
Redo  
RegisterWPDataSet  
RemoveDepOnDocFile  
RemoveDivision  
RemoveNamedProperty  
SanityCheck  
Save  
SaveAs  
SaveDivision  
SetDocumentEpoch

SetNamedProperty

Undo

UndoRedo

Unlink

UnregisterWPDataSet

UpdateOle

UpdatePowerFields

UpdatePowerFieldsOnNew

## **Events**

Created

DocumentInsert

DocumentInserted

EnterClickHere

EnterLayout

ExitClickHere

ImportInsert

ImportInserted

KeyStroke

MouseDown

MouseUp

Opened

PreClose

PrePrint

Printed

Save

SaveAs

Saved

SavedAs

WMCommand

## **Word Pro: TextMarkerCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## Word Pro: TextMarker class members

### Properties

Alignment AS Alignment  
Amikake AS Amikake  
Application AS WPAApplication  
AtBeginning AS Integer (Boolean)  
AtBeginningOfLine AS Integer (Boolean)  
AtBeginningOfObject AS Integer (Boolean)  
AtBeginningOfParagraph AS Integer (Boolean)  
AtBeginningOfWord AS Integer (Boolean)  
AtEnd AS Integer (Boolean)  
AtEndOfLine AS Integer (Boolean)  
AtEndOfObject AS Integer (Boolean)  
AtEndOfParagraph AS Integer (Boolean)  
AtEndOfWord AS Integer (Boolean)  
Attributes AS Attributes  
Breaks AS Breaks  
Bullet AS Bullet  
CharacterBorder AS CharacterBorder  
CharacterStyle AS CharacterStyle  
CharacterStyleName AS String  
CodePage AS Integer  
Collapsible AS Integer  
ColumnNumber AS Integer  
ColumnWidth AS Long (measured in Twips)  
CurrentLanguage AS Languages  
Description AS String  
DivisionInfo AS DivisionInfo  
DivisionsRequired AS Integer  
EffectiveColumnWidth AS Long (measured in Twips)  
Expandable AS Integer  
FieldType AS String  
Font AS Font  
FormatCheckLevel AS Integer  
FormatCheckRule AS String  
FormatCheckSuggestion AS String  
HasLocalTabs AS CommandResponse  
Indent AS Indent  
IsChanged AS Integer (Boolean)  
IsInBulletEditMode AS Integer (Boolean)  
IsMarkerValid AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsParaDemandLoad AS Integer (Boolean)  
IsParagraphParent AS Integer (Boolean)  
IsPrivate AS Integer (Boolean)  
IsRegistered AS Integer (Boolean)

IsReplaceable AS Integer (Boolean)  
IsRevisionMark AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Kinsoku AS Kinsoku  
Language AS Language  
LastEditorName AS String  
Layout AS Layout  
LayoutName AS String  
MaintainEditor AS Integer  
MarkerClass AS MarkerType  
MultiCompareParaTag AS String  
MultiCompareParaTagSet AS Integer (Boolean)  
Name AS String  
NextText AS String  
NormalParagraph AS Integer (Boolean)  
Numbering AS Numbering  
NumberOfCharacters AS Long  
NumberOfRevisions AS Integer  
NumCharsInParagraph AS Long  
NumCols AS Integer  
NumRows AS Integer  
ObjectType AS String  
PageNumber AS Integer  
PageNumberAsText AS String  
ParagraphHasText AS Integer (Boolean)  
ParagraphBorder AS ParagraphBorder  
ParagraphStyle AS ParagraphStyle  
ParagraphStyleName AS String  
Parent AS BaseObject  
Partial AS Integer (Boolean)  
PositionXInContainer AS Long (measured in Twips)  
PositionXOnPage AS Long (measured in Twips)  
PositionYInContainer AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
RelativeIndent AS RelativeIndent  
RenderedPageNumber AS String  
RevisionMark AS Revision  
SectionName AS String  
SelectionHidden AS Integer (Boolean)  
SelectionType AS SelectionType  
Spacing AS Spacing  
StartColumns AS Integer  
StartRow AS Integer  
StateID AS Long  
StyleExceptions AS Long  
TabRack AS TabRack  
TextViewAttributes AS TextView

VersionID AS Long

WPDataSets AS WPDataSetCollection

WPDataSets AS WPDataSetCollection

## **Methods**

Add

AdjustShade

Backspace

Backward

Bisect

CalcSmartLevels

CalculateSmartLevels

Clear

ClearInternalSpellInfo

CloseObject

ContractOutlineLevel

DeleteChars

DeleteContents

DeleteMarker

Demote

Deselect

DivideText

Embed

ExpandOutline

Find

FormatCheckAction

FormatCheckReplace

Forward

GetContents

GetCopyFormatCategories

GetCount

GetCurrentMarkerName

GetMarkedText

GetMisspelledWord

GetNamedProperty

GetParagraphNumber

GetParaNumber

GetPasteFormatCategories

GetPosition

GetSpellStatus

GetSpellUserDictStatus

GetText

GetWordMisspelled

GoTo

HasNamedProperty

InsertBreak

InsertDocInfo

InsertHardSpace  
InsertMarker  
InsertNumber  
InsertPageNumber  
InsertTab  
InsertText  
InternalCopy  
InternalCut  
InternalPaste  
IsMarkerEqualToSelection  
IsPointWithin  
Mark  
MarkIndexAll  
MorphSelectionToTable  
MoveDown  
MoveParagraph  
MoveToEnd  
MoveToStart  
MoveUp  
Next  
OpenObject  
Previous  
Promote  
RegisterWPDataSet  
RegisterWPDataSet  
Remove  
RemoveNamedProperty  
Replace  
RevertToStyle  
RevisionAccept  
RevisionCancel  
Select  
SelectMarker  
SetNamedProperty  
SetStyle  
Shade  
ShowCursor  
Skip  
SortParagraphs  
SpellWord  
SplitParagraph  
SRReplace  
TerminateFormatCheck  
TextNumber  
UnregisterWPDataSet  
UnregisterWPDataSet

**Events**  
None

## **Word Pro: TextStyleCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## Word Pro: Text class members

### Properties

Alignment AS Alignment  
Amikake AS Amikake  
Application AS WPAApplication  
AtBeginning AS Integer (Boolean)  
AtBeginningOfLine AS Integer (Boolean)  
AtBeginningOfObject AS Integer (Boolean)  
AtBeginningOfParagraph AS Integer (Boolean)  
AtBeginningOfWord AS Integer (Boolean)  
AtEnd AS Integer (Boolean)  
AtEndOfLine AS Integer (Boolean)  
AtEndOfObject AS Integer (Boolean)  
AtEndOfParagraph AS Integer (Boolean)  
AtEndOfWord AS Integer (Boolean)  
Attributes AS Attributes  
Breaks AS Breaks  
Bullet AS Bullet  
CanEmbed AS Integer (Boolean)  
CharacterBorder AS CharacterBorder  
CharacterStyle AS CharacterStyle  
CharacterStyleName AS String  
ClassName AS String  
CodePage AS Integer  
ColumnNumber AS Integer  
ColumnWidth AS Long (measured in Twips)  
ContentType AS ContentType  
CurrentLanguage AS Languages  
Description AS String  
DivisionsRequired AS Integer  
EffectiveColumnWidth AS Long (measured in Twips)  
FieldType AS String  
Font AS Font  
FormatCheckLevel AS Integer  
FormatCheckRule AS String  
FormatCheckSuggestion AS String  
HasLocalTabs AS CommandResponse  
Indent AS Indent  
IsEmpty AS Integer (Boolean)  
IsInBulletEditMode AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsParaDemandLoad AS Integer (Boolean)  
IsParagraphParent AS Integer (Boolean)  
IsPrivate AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)  
IsReplaceable AS Integer (Boolean)

IsRevisionMark AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Kinsoku AS Kinsoku  
Language AS Language  
LastEditorName AS String  
LayoutName AS String  
MaintainEditor AS Integer  
MultiCompareParaTag AS String  
MultiCompareParaTagSet AS Integer (Boolean)  
Name AS String  
NextText AS String  
NormalParagraph AS Integer (Boolean)  
Numbering AS Numbering  
NumberOfCharacters AS Long  
NumberOfRevisions AS Integer  
NumCharsInParagraph AS Long  
ObjectType AS String  
ParagrapHasText AS Integer (Boolean)  
ParagraphBorder AS ParagraphBorder  
ParagraphStyle AS ParagraphStyle  
ParagraphStyleName AS String  
Parent AS BaseObject  
Partial AS Integer (Boolean)  
PositionXInContainer AS Long (measured in Twips)  
PositionXOnPage AS Long (measured in Twips)  
PositionYInContainer AS Long (measured in Twips)  
PositionYOnPage AS Long (measured in Twips)  
RelativeIndent AS RelativeIndent  
RenderedPageNumber AS String  
RevisionMark AS Revision  
SectionName AS String  
SelectionHidden AS Integer (Boolean)  
SelectionType AS SelectionType  
Spacing AS Spacing  
StyleExceptions AS Long  
TabRack AS TabRack  
TextViewAttributes AS TextView  
VersionID AS Long  
WPDataSets AS WPDataSetCollection

## **Methods**

Add  
AdjustShade  
Backspace  
Backward  
Bisect  
CalcSmartLevels



CalculateSmartLevels  
Clear  
ClearInternalSpellInfo  
CloseObject  
ContractOutlineLevel  
DeleteChars  
Demote  
Deselect  
DivideText  
Embed  
ExpandOutline  
Find  
FormatCheckAction  
FormatCheckReplace  
Forward  
GetCopyFormatCategories  
GetCount  
GetCurrentMarkerName  
GetMisspelledWord  
GetParaNumber  
GetPasteFormatCategories  
GetPosition  
GetSpellStatus  
GetSpellUserDictStatus  
GetText  
GetWordMisspelled  
InsertBreak  
InsertDocInfo  
InsertHardSpace  
InsertNumber  
InsertPageNumber  
InsertTab  
InsertText  
InternalCopy  
InternalCut  
InternalPaste  
IsMarkerEqualToSelection  
IsPointWithin  
Mark  
MarkIndexAll  
MorphSelectionToTable  
MoveDown  
MoveParagraph  
MoveToEnd  
MoveToStart  
MoveUp  
Next

OpenObject  
Previous  
Promote  
RegisterWPDataSet  
Remove  
Replace  
RevertToStyle  
RevisionAccept  
RevisionCancel  
Select  
SelectMarker  
SetStyle  
Shade  
ShowCursor  
Skip  
SortParagraphs  
SpellWord  
SplitParagraph  
SRReplace  
TerminateFormatCheck  
TextNumber  
UnregisterWPDataSet

**Events**

None

## Word Pro: TOCSuperTableLayout class members

### Properties

AbsoluteOn AS Integer (Boolean)  
AbsoluteXPos AS Long (measured in Twips)  
AbsoluteYPos AS Long (measured in Twips)  
AccessRights AS AccessRights  
AmtTether AS WhereType  
AmtToRotateContent AS Integer  
AmtToTetherFrom AS WhereType  
Application AS WPApplication  
Background AS Background  
BaseLineOffset AS Long (measured in Twips)  
BinName AS String  
BinNumber AS Integer  
BorderLines AS BorderLines  
BorderOffset AS Long (measured in Twips)  
BottomExternalMargin AS Long (measured in Twips)  
BottomIntArea AS Long (measured in Twips)  
Center AS Integer (Boolean)  
CenteredHorz AS Integer (Boolean)  
CenteredVert AS Integer (Boolean)  
ChildLayouts AS LayoutCollection  
ClassName AS String  
ColumnBalance AS Integer (Boolean)  
ColumnGap AS Long (measured in Twips)  
ConditionType AS ConditionType  
Content AS Variant  
ContentName AS String  
ContentStyleName AS String  
Definition AS Long  
Description AS String  
DirectionDown AS LayoutDirection  
DirectionLeft AS LayoutDirection  
DirectionRight AS LayoutDirection  
DirectionUp AS LayoutDirection  
DivisionName AS String  
DropCapPosition AS Integer  
EditorName AS String  
Footer AS Layout  
GridDistance AS Long (measured in Twips)  
GridType AS GridType  
Gutter AS Gutter  
Header AS Layout  
Height AS Long (measured in Twips)  
IsAdopted AS Integer (Boolean)  
IsBottomAligned AS Integer (Boolean)

IsBreakable AS Integer (Boolean)  
IsChildSpannable AS Integer (Boolean)  
IsCollapsed AS Integer (Boolean)  
IsCollapsible AS Integer (Boolean)  
IsColumnBreakable AS Integer (Boolean)  
IsComplex AS Integer (Boolean)  
IsConnected AS Integer (Boolean)  
IsContentReplaceable AS Integer (Boolean)  
IsErrorChecking AS Integer (Boolean)  
IsExpandDown AS Integer (Boolean)  
IsExpandLeft AS Integer (Boolean)  
IsExpandRight AS Integer (Boolean)  
IsExpandUp AS Integer (Boolean)  
IsHonorProtected AS Integer (Boolean)  
IsIntersectSiblings AS Integer (Boolean)  
IsLocal AS Integer (Boolean)  
IsLocked AS Integer (Boolean)  
IsMarginSameAsParent AS Integer (Boolean)  
IsMirrorImage AS Integer (Boolean)  
IsNotCopyable AS Integer (Boolean)  
IsNotGroupable AS Integer (Boolean)  
IsNoUICommAllowed AS Integer (Boolean)  
IsOverridden AS Integer (Boolean)  
IsOverride AS Integer (Boolean)  
IsPageBreak AS Integer (Boolean)  
IsPartOfGroup AS Integer (Boolean)  
IsPrintable AS Integer (Boolean)  
IsProtected AS Integer (Boolean)  
IsRevision AS Integer (Boolean)  
IsScripting AS Integer (Boolean)  
IsScrollable AS Integer (Boolean)  
IsSingleClickEntry AS Integer (Boolean)  
IsSizable AS Integer (Boolean)  
IsSnapTo AS Integer (Boolean)  
IsStyle AS Integer (Boolean)  
IsTableHeading AS Integer (Boolean)  
IsTOC AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
Join AS Join  
Justifiable AS Integer (Boolean)  
LandscapeMode AS Integer (Boolean)  
Layer AS Layout  
Layer AS String  
LeaderDotType AS LeaderDotType  
LeftExternalMargin AS Long (measured in Twips)  
LeftIntArea AS Long (measured in Twips)  
LeftPage AS Layout

LeftTopCellId AS Integer  
LineLocation AS Integer  
LinkFrame AS String  
MaintainAspectRatio AS Integer (Boolean)  
MarginBottom AS Long (measured in Twips)  
MarginLeft AS Long (measured in Twips)  
MarginRight AS Long (measured in Twips)  
MarginTop AS Long (measured in Twips)  
MasterName AS String  
MinBottomMargin AS Long (measured in Twips)  
MinHeight AS Long (measured in Twips)  
MinLeftMargin AS Long (measured in Twips)  
MinRightMargin AS Long (measured in Twips)  
MinTopMargin AS Long (measured in Twips)  
Name AS String  
NameBasedOnStyle AS String  
NumberOfLines AS Integer  
NumCols AS Integer  
NumColsSpannedOneCell AS Integer  
NumericFormat AS NumericFormat  
NumRowsSpannedOneCell AS Integer  
PageToUseLayoutOn AS Integer  
Parent AS BaseObject  
ParentName AS String  
RelativeType AS RelativeType  
RelativeXDistance AS Long (measured in Twips)  
RelativeYDistance AS Long (measured in Twips)  
RevisionType AS Integer  
RightExternalMargin AS Long (measured in Twips)  
RightIntArea AS Long (measured in Twips)  
RightPage AS Layout  
RotationAngle AS Integer  
ScaleHeight AS Long (measured in Twips)  
ScaleMode AS ScaleType  
ScalePercentage AS Long  
ScaleWidth AS Long (measured in Twips)  
ScriptName AS String  
SectionName AS String  
SelectType AS LayoutSelect  
Shadow AS Shadow  
Span AS Integer (Boolean)  
Style AS Layout  
StyleExceptions AS Long  
TabRack AS TabRack  
TextOrient AS TextOrient  
Tile AS Integer (Boolean)  
TOCNumEntries AS Integer

TOCRange AS String  
TOCSource AS GenerateFrom  
TopExternalMargin AS Long (measured in Twips)  
TopIntArea AS Long (measured in Twips)  
TopLeftCellRowId AS Integer  
UseFooter AS Integer  
UseHeader AS Integer (Boolean)  
UsePrinterSettings AS Integer (Boolean)  
UseWhen AS UseWhen  
VersionID AS Long  
VertAlign AS VertAlign  
WasDeletedInRevMarkMode AS Integer (Boolean)  
WasInsertedInRevMarkMode AS Integer (Boolean)  
Where AS WhereType  
Width AS Long (measured in Twips)  
WPDataSets AS WPDataSetCollection  
WrapType AS WrapType  
XOffset AS Long (measured in Twips)  
XPosition AS Long (measured in Twips)  
YOffset AS Long (measured in Twips)  
YPosition AS Long (measured in Twips)

#### **Methods**

AddChildToLayout  
AddTOCEntry  
Backward  
Clear  
DeleteContents  
DeleteLayout  
DoesMarkerNameMatch  
FindClass  
Forward  
GetMarkerName  
GetNamedProperty  
GetTOCProperties  
GoToLayout  
HasNamedProperty  
Layer  
Mark  
MirrorPage  
MoveToBack  
MoveToFront  
Next  
PreviousItem  
RegisterWPDataSet  
RemoveChildFromLayout  
RemoveNamedProperty

RevisionAcceptLayoutChange

RevisionCancelLayoutChange

SetAllMargins

SetMinimumOrigin

SetNamedProperty

SetTOCProperties

UnregisterWPDataSet

Update

**Events**

EnterLayout

KeyStroke

MouseDown

MouseUp

## **Word Pro: UnitCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None



## Word Pro: UserInterfacePrefs class members

### Properties

Application AS WPAApplication  
AutoBackup AS Integer (Boolean)  
AutoSave AS Integer (Boolean)  
AutoSaveMinutes AS Integer  
BackupPath AS String  
BackupPaths AS StringCollection  
BulletFonts AS String  
CheckForScriptEnumError AS Integer (Boolean)  
CheckForScriptPropertyError AS Integer (Boolean)  
CleanScreenMode AS Integer (Boolean)  
CleanScreenOnStartUp AS Integer (Boolean)  
Description AS String  
DocPath AS String  
DocumentPaths AS StringCollection  
FontUnitName AS String  
FontUnits AS ScaleMode  
GlossaryDataFileName AS String  
GlossaryDataFiles AS StringCollection  
GlossaryDataPaths AS StringCollection  
GlossaryPath AS String  
GoToIndexLastItemAccessed AS Integer  
GoToItemName AS String  
GoToPageLoadInBackground AS Integer (Boolean)  
GoToPageNumber AS Integer  
GoToSelection AS GoToType  
GrammarOptions AS Grammar  
HorizontalSplitWindow AS Integer (Boolean)  
IconPath AS String  
IconPaths AS StringCollection  
IconShowingBeforeCleanScreen AS Integer (Boolean)  
IsDisableWarningMessages AS Integer (Boolean)  
IsHighlightNote AS Integer (Boolean)  
IsPrintInBackground AS Integer (Boolean)  
IsReplacement AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
IsWorkingDir AS Integer (Boolean)  
LastDocPath AS String  
LastGraphicPath AS String  
LastGraphicType AS String  
LastMacroPath AS String  
LastUsedDateFormula AS String  
LeaderDotType AS LeaderDotType  
LoadFilesMaximized AS Integer (Boolean)  
MacroPath AS String

MacroPaths AS StringCollection  
MaximizeOnStartup AS Integer (Boolean)  
Name AS String  
NoteColor AS Color  
NumOfRecentFiles AS Integer  
OpenDocsVisible AS Integer (Boolean)  
OpenExisingFileInWelcomeBox AS Integer (Boolean)  
OpenReadOnly AS Integer (Boolean)  
Parent AS BaseObject  
RetainNameOfImportedFile AS Integer (Boolean)  
RunMacroOnDocEvents AS Integer (Boolean)  
RunMacroOnLoad AS Integer (Boolean)  
ShowFileNew AS Integer  
ShowGraphicPreview AS Integer (Boolean)  
ShowMailDisabled AS Integer (Boolean)  
ShowNoWelcomeBox AS Integer (Boolean)  
SizingUnitName AS String  
SizingUnits AS ScaleMode  
SpacingUnitName AS String  
SpacingUnits AS ScaleMode  
SpellCheckIncludesOtherTextStreams  
AS Integer (Boolean)  
SpellCheckInitialCaps AS Integer (Boolean)  
SpellCheckRepeatedWords AS Integer (Boolean)  
SpellCheckStartsAtBeginning AS Integer (Boolean)  
SpellCheckUserDictAlternatives AS Integer (Boolean)  
SpellCheckWordsWithNums AS Integer (Boolean)  
SplitPercentage AS Integer  
StatusSpellReplaceAll AS Integer (Boolean)  
StylePath AS String  
StylePaths AS StringCollection  
StyleSheetName AS String  
TabRelativeTo AS Integer  
TabType AS TabType  
ThesMaxNumSynonymsReturned AS Integer  
TimedSaveFileExtension AS String  
Units AS StringCollection  
UserDefinedFilter AS String  
UserDictFiles AS String  
UserDictionaryFiles AS StringCollection  
UserDictionaryPath AS String  
UserDictionaryPaths AS StringCollection  
UserDictStates AS String  
VersionID AS Long  
VerticalSplitWindow AS Integer (Boolean)  
WinViewPrefs AS WinViewPrefs  
WorkingType AS Integer (Boolean)

**Methods**

GetArrayProp

InsertPath

SetArrayProp

**Events**

None

## **Word Pro: UseWhen class members**

### **Properties**

Application AS WPApplication

Condition AS ConditionType

Description AS String

IsValid AS Integer (Boolean)

Name AS String

PageNo AS Integer

Parent AS BaseObject

Start AS StartType

VersionID AS Long

### **Methods**

None

### **Events**

None

## **Word Pro: VersionCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: VersionManager class members**

### **Properties**

Application AS WPAApplication

CurrentVersion AS Version

Description AS String

DocVersionID AS Long

GetRedoWhatDesc AS String

GetUndoWhatDesc AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

Versions AS VersionCollection

### **Methods**

CreateVersion

DeleteVersion

LeastRecentVersion

MostRecentVersion

NextVersion

NumberOfVersions

PreviousVersion

### **Events**

None

## **Word Pro: Version class members**

### **Properties**

Application AS WPApplication

AuthorName AS String

CreationDateString AS String

CreationTimeString AS String

DateCreatedValue AS Long

DateRevisedValue AS Long

Description AS String

DocVersionID AS Long

EditorName AS String

Editors AS StringCollection

IsValid AS Integer (Boolean)

ModifiedDateString AS String

ModifiedTimeString AS String

Name AS String

NumberOfRevisions AS Long

Parent AS BaseObject

VersionID AS Long

VersionName AS String

### **Methods**

CreateRemark

### **Events**

None

## **Word Pro: Window class members**

### **Properties**

Active AS Integer (Boolean)

Application AS WPApplication

Caption AS String

Description AS String

Height AS Long (measured in Twips)

IsValid AS Integer (Boolean)

Left AS Long (measured in Twips)

Name AS String

Parent AS BaseObject

Top AS Long (measured in Twips)

VersionID AS Long

Visible AS Integer (Boolean)

Width AS Long (measured in Twips)

### **Methods**

Close

Move

Open

Resize

Update

### **Events**

Moved



## Word Pro: WinViewPrefs class members

### Properties

Application AS WPAApplication  
CustomViewLevel AS Integer  
Description AS String  
IsDisplayMisspelled AS Integer (Boolean)  
IsHideHeaderFooter AS Integer (Boolean)  
IsHorizontalScrollBarCleanScrn AS Integer (Boolean)  
IsInDraft AS Integer (Boolean)  
IsInOutline AS Integer (Boolean)  
IsInPageSort AS Integer (Boolean)  
IsMarginsInColor AS Integer (Boolean)  
IsPageBreakMarks AS Integer (Boolean)  
IsSpellMode AS Integer (Boolean)  
IsValid AS Integer (Boolean)  
IsVerticalScrollBarCleanScrn AS Integer (Boolean)  
IsViewAnchor AS Integer (Boolean)  
IsViewBookmarks AS Integer (Boolean)  
IsViewClickHereBlocks AS Integer (Boolean)  
IsViewColGuides AS Integer (Boolean)  
IsViewColumnBreakMarks AS Integer (Boolean)  
IsViewDDELinks AS Integer (Boolean)  
IsViewHorzRuler AS Integer (Boolean)  
IsViewHorzScrollBar AS Integer (Boolean)  
IsViewMenuCleanScrn AS Integer (Boolean)  
IsViewNotes AS Integer (Boolean)  
IsViewOutlineFlowToScreen AS Integer (Boolean)  
IsViewOutlineIndent AS Integer (Boolean)  
IsViewPageLayoutMarks AS Integer (Boolean)  
IsViewParallelColumnBorder AS Integer (Boolean)  
IsViewPictures AS Integer (Boolean)  
IsViewReturnIconCleanScrn AS Integer (Boolean)  
IsViewReturns AS Integer (Boolean)  
IsViewRulerMarks AS Integer (Boolean)  
IsViewSectionBreakMarks AS Integer (Boolean)  
IsViewSectionTabs AS Integer (Boolean)  
IsViewSmartIconsCleanScrn AS Integer (Boolean)  
IsViewStatusBarCleanScrn AS Integer (Boolean)  
IsViewStatusSpell AS Integer (Boolean)  
IsViewTableGridLines AS Integer (Boolean)  
IsViewTableHeadings AS Integer (Boolean)  
IsViewTabs AS Integer (Boolean)  
IsViewThumbBarCleanScrn AS Integer (Boolean)  
IsViewTitleBarCleanScrn AS Integer (Boolean)  
IsViewVertRuler AS Integer (Boolean)  
IsViewVertScrollBar AS Integer (Boolean)

Name AS String

NumCols AS Integer

OutlineButtons AS Integer

OutlineHeadingButtonsOnly AS Integer

OutlineOnlyHeadingsWhenCollapsed

AS Integer

Parent AS BaseObject

VersionID AS Long

ViewType AS Views

### **Methods**

ClearSplits

ClearUpdate

Refresh

RestorePreviousView

### **Events**

None

## Word Pro: WPAApplication class members

### Properties

[ActiveDocument](#) AS [TextDocument](#)  
[ActiveDocWindow](#) AS [DocWindow](#)  
[AppFoundry](#) AS [Foundry](#)  
[Application](#) AS [WPAApplication](#)  
[ApplicationWindow](#) AS [ApplicationWindow](#)  
[AppViewPrefs](#) AS [AppViewPrefs](#)  
[BaseTable](#) AS Variant  
[Cell](#) AS [CellContainer](#)  
[Container](#) AS Variant  
[Content](#) AS [Content](#)  
[CurrentCell](#) AS [CellLayout](#)  
[CurrentColumn](#) AS [Layout](#)  
[CurrentRow](#) AS [RowLayout](#)  
[DefaultFilePath](#) AS String  
[Description](#) AS String  
[Division](#) AS [Division](#)  
[Divisions](#) AS [DivisionCollection](#)  
[Documents](#) AS [Documents](#)  
[FileProtection](#) AS [FileProtection](#)  
[FindAndReplace](#) AS [FindAndReplace](#)  
[Format](#) AS [FormatPreferences](#)  
[FormatCheckPreferences](#) AS [FormatCheckPref](#)  
[Foundry](#) AS [Foundry](#)  
[Frame](#) AS [FrameContainer](#)  
[FullName](#) AS String  
[GetHomeDirectory](#) AS String  
[Graphic](#) AS [Graphic](#)  
[GraphicOleObject](#) AS Variant  
[Interactive](#) AS Integer (Boolean)  
[IsValid](#) AS Integer (Boolean)  
[KeyboardLanguage](#) AS Integer  
[Language](#) AS String  
[Layout](#) AS [Layout](#)  
[Location](#) AS String  
[Name](#) AS String  
[OleObject](#) AS [OleObject](#)  
[Page](#) AS [PageContainer](#)  
[ParallelColumns](#) AS [ParallelColumns](#)  
[Parent](#) AS [BaseObject](#)  
[Path](#) AS String  
[Preferences](#) AS [Preferences](#)  
[SmartCorrect](#) AS [SmartCorrect](#)  
[SmartCorrects](#) AS [SmartCorrectCollection](#)  
[SmartFill](#) AS [SmartFillCollection](#)

[SuperTableContainer](#) AS [SuperTableContainer](#)  
[Table](#) AS [Table](#)  
[TableContainer](#) AS Variant  
[TableOnlyContainer](#) AS [TableOnlyCont](#)  
[TempFindAndReplace](#) AS [FindAndReplace](#)  
[TempFoundry](#) AS [Foundry](#)  
[Text](#) AS [Text](#)  
[VersionID](#) AS Long  
[Visible](#) AS Integer (Boolean)  
[WPDataSets](#) AS [WPDataSetCollection](#)

#### **Methods**

[AddIndexEntry](#)  
[AddTOCEntry](#)  
[AnswerMsgBox](#)  
[BeginChange](#)  
[Bold](#)  
[BringFrameToFront](#)  
[BringFrameToFrontOne](#)  
[CalculateSmartLevels](#)  
[CascadeWindow](#)  
[CellRevert](#)  
[ChangeSmartMaster](#)  
[ClearParaRevisionTags](#)  
[ClearTempFoundry](#)  
[Close](#)  
[CloseAll](#)  
[CloseDocWindow](#)  
[CloseMergeDataFile](#)  
[CombineDivisions](#)  
[CombineSections](#)  
[CompareFiles](#)  
[ConnectCells](#)  
[ConnectRows](#)  
[ContractOutlineLevel](#)  
[CopySelection](#)  
[CreateDataFile](#)  
[CreateDivision](#)  
[CreateDropCap](#)  
[CreateExternalDivision](#)  
[CreateFrame](#)  
[CreateGlossary](#)  
[CreateGlossaryEntry](#)  
[CreateGraphic](#)  
[CreateOleEmbeddedFile](#)  
[CreateOleLinkedFile](#)  
[CreateOleNew](#)

[CreateParallelColumns](#)  
[CreateTable](#)  
[CutSelection](#)  
[DblUnderline](#)  
[DeleteDivision](#)  
[DeleteKey](#)  
[DeleteParallelColumns](#)  
[DeleteTable](#)  
[DemoteOutlineLevel](#)  
[Deselect](#)  
[DisconnectCells](#)  
[EditClickHereLink](#)  
[Embed](#)  
[EndChange](#)  
[EnvelopePrint](#)  
[ExpandOutlineLevel](#)  
[FastFormat](#)  
[Find](#)  
[FormatCheck](#)  
[FormatCheckAction](#)  
[FrameRevert](#)  
[GetCopyFormatCategories](#)  
[GetCurrentMarkerName](#)  
[GetEnum](#)  
[GetPasteFormatCategories](#)  
[GetProfileString](#)  
[GlossaryInsert](#)  
[GlossaryOpen](#)  
[GoToBookmark](#)  
[GoToLayout](#)  
[GotoNextParallelColumn](#)  
[GoToObject](#)  
[GoToPage](#)  
[GroupDivision](#)  
[HandsOffStorage](#)  
[Help](#)  
[Hide](#)  
[HighlightToggle](#)  
[HourGlass](#)  
[ImportGraphic](#)  
[InitFindAndReplace](#)  
[InsertBullet](#)  
[InsertClickHere](#)  
[InsertClickHereLink](#)  
[InsertColumnBreak](#)  
[InsertDate](#)  
[InsertDocInfo](#)

[InsertDocument](#)  
[InsertField](#)  
[InsertFootnote](#)  
[InsertFrame](#)  
[InsertIndex](#)  
[InsertNote](#)  
[InsertOleDivision](#)  
[InsertPageBreak](#)  
[InsertPageLayout](#)  
[InsertPageNumber](#)  
[InsertRuby](#)  
[InsertSection](#)  
[InsertTOC](#)  
[InternalCopy](#)  
[InternalCut](#)  
[InternalPaste](#)  
[IsWMCommandValid](#)  
[Italic](#)  
[LowerCase](#)  
[MacroAppend](#)  
[MacroCancel](#)  
[MacroCompile](#)  
[MacroEndRecord](#)  
[MacroPlay](#)  
[MacroQuickPlay](#)  
[MacroQuickRecord](#)  
[MacroRecord](#)  
[MacroResume](#)  
[MacroRun](#)  
[MailDocument](#)  
[MakeTableFromText](#)  
[ManualFrame](#)  
[ManualTable](#)  
[Mark](#)  
[MarkRevisionInsert](#)  
[Merge](#)  
[MergeAddDataRecord](#)  
[MergeContinue](#)  
[MergeSetDataFile](#)  
[MergeStart](#)  
[Messages](#)  
[MoveDivision](#)  
[NewDivision](#)  
[NewDocument](#)  
[NewFrame](#)  
[NewWindow](#)  
[NormalText](#)

OpenDataFile  
OpenDocument  
OpenDocumentFromNotes  
OpenFromStorage  
OutlineMoveTextDown  
OutlineMoveTextUp  
PageDown  
PageUp  
Paste  
PasteLink  
PasteSpecial  
PColConnectCells  
PColConnectRows  
PColDisconnectCells  
PColSelectColumn  
PColSelectRow  
PColSelectTable  
Print  
PrintOut  
PromoteOutlineLevel  
QueryDrop  
QuickAlignFrame  
QuickAlignTable  
Quit  
Redo  
RegisterWPDataSet  
Render  
Replace  
ReplaceAll  
ReplaceCmd  
ResetFindAndReplace  
RetrieveInternetFile  
RetrieveInternetFileAndOpen  
RevertToSaved  
RevertToStyle  
ReviseAcceptAll  
ReviseCancelAll  
RunAutoNewMacro  
RunAutoOpenMacro  
RunScript  
Save  
SaveAs  
SaveAsToNotes  
SaveEnvelopeMaster  
SaveGlossary  
SaveThumbnailBitmap  
SaveToStorage

SaveToStorageComplete

SaveVersion

Select

SelectAStyle

SelectCell

SelectColumn

SelectDoc

SelectEntireCellRange

SelectEntireColumn

SelectEntirePCol

SelectEntirePColCellRange

SelectEntirePColColumn

SelectEntirePColRow

SelectEntireRow

SelectEntireTable

SelectParagraph

SelectPColCell

SelectRow

SelectSection

SelectSentence

SelectTable

SelectWord

SendFrameToBack

SendFrameToBackOne

SendMailSelectedText

SetStorage

SetStyle

SetUpEnvelopeMerge

SmallCaps

SmartSumColumn

SmartSumRow

SortParagraphs

SpellAddToUserDict

SpellClearSkippedWords

SpellMarkSkippedWords

SpellSkipAll

StartEditMergeData

StartEnvelopeDiv

StartFieldInsert

StoreInternetFile

StrikeThru

SubScript

SuperScript

TeamMail

TileWindowHorz

TileWindowVert

TimedSave



ToggleIconBar  
Type  
Underline  
Undo  
UnregisterWPDataSet  
UpdateFootersText  
UpdateHeadersText  
UpdateIndexSection  
UpdateOle  
UpdateTOC  
UpperCase  
WMCommand  
WordUnderline  
WriteProfileString

**Events**

DocumentClose  
DocumentClosed  
DocumentCreate  
DocumentCreated  
DocumentExport  
DocumentExported  
DocumentImport  
DocumentImported  
DocumentInsert  
DocumentInserted  
DocumentOpen  
DocumentOpened  
DocumentPrint  
DocumentPrinted  
DocumentSave  
DocumentSaveAs  
DocumentSaved  
DocumentSavedAs  
EnterClickHere  
EnterLayout  
ExitClickHere  
ImportInsert  
ImportInserted  
KeyStroke  
MouseDown  
MouseUp  
Quit  
WMCommand

## **Word Pro: WPDataSetCollection class members**

### **Properties**

Application AS WPApplication

Count AS Long

Description AS String

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

IsEmpty

Item

### **Events**

None

## **Word Pro: WPDataSet class members**

### **Properties**

Application AS WPApplication

DataNames AS StringCollection

Description AS String

IsPersistent AS Integer (Boolean)

IsValid AS Integer (Boolean)

Name AS String

Parent AS BaseObject

VersionID AS Long

### **Methods**

GetData

IsDataNameUsed

SetData

### **Events**

None

'Example: PreviousName property

'This example script has not yet been created.

'Example: PreviousNeighbor property

'This example script has not yet been created.

'Example: PreviousVersion method

'This example script has not yet been created.

'Example: Previous method

'This example script has not yet been created.

'Example: PrintDestination property  
'This example script has not yet been created.



'Example: PrintDocDescription property

'This example script has not yet been created.

'Example: Printed event

'This example script has not yet been created.

'Example: PrinterName property

'This example script has not yet been created.

'Example: PrintGraphics property

'This example script has not yet been created.

'Example: PrintInBackground property  
'This example script has not yet been created.

'Example: PrintManager property

'This example script has not yet been created.

'Example: PrintOut method

' This example prints the current page of the current document.

' RUNTIME DEPENDENCIES: You must have an open document configured

' to an available printer and for this script to work.

.ActiveDocument.PrintSettings.PrintRange = \$LtsPrintRangeCurrentPage

.PrintOut

'Example: PrintPagesFrom property

'This example script has not yet been created.



'Example: PrintPagesTo property

'This example script has not yet been created.

'Example: PrintPageType property

'This example script has not yet been created.

'Example: PrintRange property

'This example script has not yet been created.

'Example: PrintSettings property

'This example script has not yet been created.

'Example: Print method

' This example prints the current page of the current document.

' RUNTIME DEPENDENCIES: You must have an open document configured

' to an available printer and for this script to work.

.ActiveDocument.PrintSettings.PrintRange = \$LtsPrintRangeCurrentPage

**.Print**

'Example: Private property

'This example script has not yet been created.

'Example: ProcessAccelKey method

'This example script has not yet been created.

'Example: PromoteOutlineLevel method

'This example script has not yet been created.



'Example: Promote method

'This example script has not yet been created.

'Example: PromptHidden property

'This example script has not yet been created.

'Example: Prompt property

'This example script has not yet been created.

'Example: ProtectedMode property

'This example script has not yet been created.

'Example: Purge method

'This example script has not yet been created.

'Example: QueryDrop method

'This example script has not yet been created.

'Example: Query method

'This example script has not yet been created.

'Example: QueueName property

'This example script has not yet been created.



```
'Example: QuickAlignFrame method
' This example creates a frame. After the message box is closed, the
' frame is then horizontally centered.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim FrameWidth as Integer
Dim FrameHeight as Integer
Dim X as Integer
Dim Y as Integer
```

```
FrameWidth = 1440
FrameHeight = 1440
X = 1440
Y = 1440
```

```
.InsertFrame FrameWidth,FrameHeight, X, Y
Messagebox "Click OK to center frame.",MB_OK,"Example Script"
```

```
.QuickAlignFrame $LwpQuickLayoutAlignHorzcenter
```

```
'Example: QuickAlignTable method
' This example creates a table with 4 rows and 5 columns based upon the
' 'Default Table' style and then changes the alignment attribute to
' left aligned.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4

.QuickAlignTable $LwpQuickLayoutAlignLeft
```

'Example: Quit event  
,

```
'Example: Quit method  
' This example closes WordPro.  
' RUNTIME DEPENDENCIES: None.
```

```
.Quit
```

'Example: ReadOnly property

'This example script has not yet been created.

```
'Example: Read method
' This example creates a bag in the active division and then writes some data
' to the bag. The data from the created bag is read and printed to the Lotus
' Script Output panel. Next, data from all bags in the Bag Collection is
' printed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim BagName As String
Dim MyBag As Bag
Dim BagData As String
```

```
BagData = "This is data for the bag."
LenBagData = Len(BagData)
```

```
BagName = .Division.Foundry.Create($LwpFoundryCreateTypeBag)
Set MyBag = .Division.Foundry.Bags.Item(BagName)
```

```
Stat = MyBag.Write(BagData, LenBagData)
If Stat = True Then
    Print "BagData= " & MyBag.Read(LenBagData)
End If
```

```
Forall ThisBag In .Division.Foundry.Bags
    ThisBag.Reset
    Print "Name = " ThisBag.Name
    Print "Length = " ThisBag.Length
    Print ThisBag.Read(ThisBag.Length)
End Forall
```

'Example: Redo method

' This example types some text into the current document which is then

' undone and redone.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "Typing this text will be be undone"

.Undo

.Redo

'Example: Red property

'This example script has not yet been created.



'Example: Refresh method

'This example script has not yet been created.

```
'Example: RegisterWPDataSet method
' This example creates a dataset named 'PhoneNumbers' off of the application
' object. Two dataset items are added and then printed to the Script Editor
' Output panel. The dataset is then removed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim AppDataSet As WPDataSet
```

```
Set AppDataSet = .RegisterWPDataSet("PhoneNumbers")
```

```
AppDataSet.SetData "Mark", "555-1234"
```

```
AppDataSet.SetData "Peyton", "555-5678"
```

```
Print AppDataSet.GetData("Mark", " ")
```

```
Print AppDataSet.GetData("Peyton", " ")
```

```
.UnRegisterWPDataSet "PhoneNumbers"
```

'Example: RelativeIndent property

'This example script has not yet been created.

'Example: RelativePageNum property

'This example script has not yet been created.

'Example: RelativeType property

'This example script has not yet been created.

'Example: RelativeXDistance property

'This example script has not yet been created.

'Example: RelativeYDistance property

'This example script has not yet been created.

'Example: Relative property

'This example script has not yet been created.



```
'Example: Release method
' This example creates a new paragraph style and then deletes the style.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Dim StyleName As String
StyleName = "My New Char Style"
Style = .Division.Foundry.Create($LwpFoundryCreateTypeStyle, StyleName, 39)
With .Division.Foundry.CharacterStyles(Style)
    .Font.Underline = True
    .Font.FontColor.Blue = 128
    .Font.FontColor.Red = 128
    .Font.FontColor.Green = 0
End With
.Foundry.Release $LwpFoundryReleaseTypeStyle, 35, StyleName
```

```
'Example: RemoveBookmark method
' This example creates a new bookmark named "NewBookMark" and then removes it.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim MarkName As String

' Get the marker name for the currently selected text.
MarkName = .Mark($LwpMarkerTypeBookmark)

' Create the bookmark from the given marker name.
.Division.BookmarkManager.AddBookmark "NewBookMark", MarkName

MessageBox "Click OK to remove the bookmark. ",MB_OK,"Example Script"

' Get the marker name for this bookmark
MarkName = .Division.BookmarkManager.Bookmarks("NewBookMark").MarkerName

' The Bookmark Manager removes bookmarks by their marker name.
.Division.BookmarkManager.RemoveBookmark (MarkName)

' Since the bookmark has been deleted, remove its marker from the Markers
' collection.
.Division.Foundry.Markers(MarkName).DeleteMarker
```

'Example: RemoveChildFromLayout method

'This example script has not yet been created.

```
'Example: RemoveDataFile method
' This example creates a data file for the current document.  Two records are
' added and the data file is then removed from the current document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateDataFile "~|", "Name~Address~City~State~Zip|", False, "C:\mergedat.lwp"

'Add some records to the virtual datafile
.MergeAddDataRecord "Jane Doe~100 Main St.~ Atlanta~ GA~30319|"
.MergeAddDataRecord "John Doe~100 Main St.~ Atlanta~ GA~30319|"

.ActiveDocument.MergeOptions.RemoveDataFile
```

'Example: RemoveDdeLink method

'This example script has not yet been created.

'Example: RemoveDepOnDocFile method

'This example script has not yet been created.

```
'Example: RemoveDivision method  
' This example creates and then removes a new division in the current  
' document.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
DivisionId = .ActiveDocument.AddDivision("NewDivision")  
.ActiveDocument.RemoveDivision DivisionId
```

```
'Example: RemoveEditor method
' This example adds a new editor with read only rights to the current
' document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim NewEditorName As String
Dim NewEditorInitials As String
```

```
NewEditorName = "Lotus User"
NewEditorInitials = "LU"
.ActiveDocument.EditorManager.AddEditorManager NewEditorName,
    NewEditorInitials
.ActiveDocument.EditorManager.RemoveEditorManager NewEditorName
```



```
'Example: RemoveNamedProperty method
' This example creates a named property, 'ExampleProp' on the active document
' and assigns it a value which is then printed to the Lotus Script Output
' panel. The named property is then removed from the active document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
.Division.SetNamedProperty "ExampleProp", "Here is some data."
Print .Division.GetNamedProperty("ExampleProp")
MessageBox "Click OK to remove the named property. ",MB_OK,"Example Script"
.Division.RemoveNamedProperty "ExampleProp"
```

'Example: RemovePersistentAccelerators method  
'This example script has not yet been created.

'Example: RemovePowerField method

'This example script has not yet been created.

'Example: RemoveProperty method

'This example script has not yet been created.

'Example: Remove method

'This example script has not yet been created.

'Example: RenderClipBitmap method

'This example script has not yet been created.

'Example: RenderClipDIB method

'This example script has not yet been created.

'Example: RenderClipMetafile method  
'This example script has not yet been created.



'Example: RenderClipPalette method

'This example script has not yet been created.

'Example: RenderedPageNumber property

'This example script has not yet been created.

'Example: Render method

'This example script has not yet been created.

```
'Example: ReplaceAll method
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script replaces all the cats with dogs.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"
.Application.FindAndReplace.ReplaceString = "dog"
.InitFindAndReplace True
MessageBox "Click OK to replace all.",MB_OK,"Example Script"

.ReplaceAll
```

'Example: ReplaceAttributes property

'This example script has not yet been created.

```
'Example: ReplaceCmd method
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script finds the first 'cat' then replaces it with
' 'dog'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"
.Application.FindAndReplace.ReplaceString = "dog"
.InitFindAndReplace True
Messagebox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find

.ReplaceCmd
```

'Example: ReplaceExactCase property

'This example script has not yet been created.

'Example: ReplaceFont property

'This example script has not yet been created.



'Example: ReplaceLanguage property  
'This example script has not yet been created.

'Example: Replacements property

'This example script has not yet been created.

```
'Example: ReplaceString property
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script finds the first 'cat' then replaces it with
' 'dog'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"

.Application.FindAndReplace.ReplaceString = "dog"

.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find
.ReplaceCmd
```

'Example: ReplaceStyleName property  
'This example script has not yet been created.

'Example: ReplaceWithProperties property  
'This example script has not yet been created.

```
'Example: Replace method
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and the
' ReplaceString to "dog", and then displays a message box.
' When you click OK, the script finds the first 'cat' then replaces it with
' 'dog'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"
.Application.FindAndReplace.ReplaceString = "dog"
.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find

.Replace "$LwpReplaceObjectTypeWord","dog"
```

'Example: RequestAndProcessData method  
'This example script has not yet been created.

'Example: RequestRemarkOnClose property

'This example script has not yet been created.



'Example: RequireStartupScripts property  
'This example script has not yet been created.

```
'Example: ResetFindAndReplace method
' This example inserts three identical sentences into the current document,
' clears the FindAndReplace settings, sets the FindString to "cat" and then
' displays a message box. When you click OK, the script finds the first 'cat'.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim SentenceCount As Integer
For SentenceCount = 1 To 3
    .Text.InsertText "The sleep-deprived fox jumped over the gravity-challenged cat."
    .Text.SplitParagraph
Next
.Application.ResetFindAndReplace
.Application.FindAndReplace.FindString = "cat"
.InitFindAndReplace True
MessageBox "Click OK to find and then replace.",MB_OK,"Example Script"
.Find
```

'Example: ResetOnEachPage property

'This example script has not yet been created.

'Example: ResetWhen property

'This example script has not yet been created.

```
'Example: Reset method
' This example creates a bag in the active division and then writes some data
' to the bag. The data from the created bag is read and printed to the Lotus
' Script Output panel. Next, data from all bags in the Bag Collection is
' printed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim BagName As String
Dim MyBag As Bag
Dim BagData As String
```

```
BagData = "This is data for the bag."
LenBagData = Len(BagData)
```

```
BagName = .Division.Foundry.Create($LwpFoundryCreateTypeBag)
Set MyBag = .Division.Foundry.Bags.Item(BagName)
```

```
Stat = MyBag.Write(BagData, LenBagData)
If Stat = True Then
    Print "BagData= " & MyBag.Read(LenBagData)
End If
```

```
Forall ThisBag In .Division.Foundry.Bags
    ThisBag.Reset
    Print "Name = " ThisBag.Name
    Print "Length = " ThisBag.Length
    Print ThisBag.Read(ThisBag.Length)
End Forall
```

```
'Example: Resize method
' This example resizes the main application window.
' RUNTIME DEPENDENCIES: None
Dim AppWidth as Integer
Dim AppHeight as Integer
AppWidth = 8640
AppHeight = 5760
.ApplicationWindow.Resize AppWidth, AppHeight
```

'Example: RestartStyleName property  
'This example script has not yet been created.

```
'Example: Restore method  
' This example restores the main application window.  
.ApplicationWindow.Restore
```



'Example: Rest property

'This example script has not yet been created.

'Example: Result property

'This example script has not yet been created.

'Example: ResumePausedMacro property

'This example script has not yet been created.

```
'Example: RetrieveInternetFileAndOpen method  
' This example retrieves and opens Lotus' home page from the Internet.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim Location As String  
Dim Proxy As String  
Dim Port As Integer
```

```
Location = "http://www.lotus.com"  
Proxy = "123.456.78.910"  
Port = 1234  
.RetrieveInternetFileAndOpen Location, "", "", 1, Proxy, Port
```

```
'Example: RetrieveInternetFile method
' This example retrieves a stock symbol from the Internet.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim Location As String
Dim Proxy As String
Dim Port As Integer
```

```
Symbol = Inputbox ("What stock symbol","Retrieve Quote","IBM")
Location = "http://www.spacecom.com:8001/cgi-bin/getquote?TICKER=" + Symbol
Proxy = "123.456.78.910"
Port = 1234
```

```
FileName = .RetrieveInternetFile(Location,"","",1,Proxy,Port)
.OpenDocument Filename, "", "HTML", "", False, True
```

'Example: RevertToSaved method

' This example returns to the last saved version of the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.RevertToSaved

```
'Example: RevertToStyle method
' This example inserts some sample text into the current document, selects
' the inserted text, modifies the font and alignment properties of that text,
' and then displays a message box.
' When you close the message box, the script uses the RevertToStyle method to
' cause all the text properties to revert back to the paragraph style assigned
' to that text.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
With .Text
    .InsertText "This is some sample text."
    .Select $LwpSelectObjectTypeParagraph
    .Font.Size = 15.00
    .Font.Bold = True
    .Alignment.AlignmentType = $LtsAlignmentHorizCenter
    MessageBox "Click OK to revert text to style.",MB_OK,"Example Script"
    .RevertToStyle $LwpStyleTypeParagraph
End With
```

'Example: ReviewVersions method

'This example script has not yet been created.



'Example: ReviewVersions property

'This example script has not yet been created.

'Example: ReviseAcceptAll method

' This example accepts the revisions in the entire document for the specified ' author name.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ReviseAcceptAll False, "Peyton McManus", ""

'Example: ReviseCancelAll method

' This example accepts the revisions in the entire document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ReviseCancelAll 0, "", ""

'Example: RevisionAccept method

'This example script has not yet been created.

## **Word Pro: ReplaceStyleName property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACESTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to replace a paragraph style in Find & Replace.

### **Data Type**

[String](#)

### **Syntax**

replacestylevalue = [objectreference].ReplaceStyleName

[objectreference].ReplaceStyleName = replacestylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Edit - Find & Replace Text, typing the character, ^p, in the "Replace with" box, clicking Options, clicking the Font button in the "Replace options" section, and choosing a paragraph style in the "Style" list box.

## **Word Pro: ReplaceWithProperties property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REPLACEWITHPROPERTIES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to replace font properties in Find & Replace.

### **Data Type**

[Integer](#)

### **Syntax**

replacewithpropertiesvalue = [objectreference].ReplaceWithProperties

[objectreference].ReplaceWithProperties = replacewithpropertiesvalue

### **Legal values**

The legal values for this property are -1 and 0 but you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this property to replace font properties in Find & Replace. If True, replaces the font properties that match the user setting. Equivalent to choosing Edit - Find & Replace Text, clicking Options, selecting "Include properties," clicking the Font button in the "Replace options" section, and selecting properties.

**Word Pro: RequestRemarkOnClose property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REQUESTREMARKONCLOSE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not a dialog box displays when a document closes and requests the current editor to enter a remark.

**Data Type**

[Integer](#)

**Syntax**

requestremarkonclosevalue = [objectreference].RequestRemarkOnClose

[objectreference].RequestRemarkOnClose = requestremarkonclosevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to choosing File - TeamSecurity and selecting "Request Editor's remark on close" on the Editing Rights panel.

## **Word Pro: RequireStartupScripts property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REQUIRESTARTUPSCRIPTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not startup scripts are required to run when a file is opened.

### **Data Type**

[Integer](#)

### **Syntax**

requirestartupscriptsvalue = [objectreference].RequireStartupScripts

[objectreference].RequireStartupScripts = requirestartupscriptsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing File - TeamSecurity and selecting "Require running of startup scripts" on the Other Protection panel.



## **Word Pro: ResetOnEachPage property**

{button ,AL('H\_LINENUMBEROPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESETONEACHPAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

resetoneachpagevalue = [objectreference].ResetOnEachPage

[objectreference].ResetOnEachPage = resetoneachpagevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: ResetWhen property

{button ,AL('H\_ENDNOTEDIVISIONGROUPNUM\_CLASS;H\_ENDNOTEDIVISIONNUM\_CLASS;H\_ENDNOTEDOC\_NUM\_CLASS;H\_FOOTNOTENUMBERING\_CLASS;H\_FOOTNOTENUMOPT\_CLASS';0)} [See list of classes](#)

{button ,AL('H\_RESETWHEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Resets the numbering for endnotes.

### Data Type

Variant (Enumerated)

ResetOption

### Syntax

resetwhenevervalue = [objectreference].ResetWhen

[objectreference].ResetWhen = resetwhenevervalue

### Legal values

\$LwpResetOptionEachDivision (1710) Increases endnote numbers throughout a division and resets with the first endnote in the next division.

\$LwpResetOptionEachDivisiongroup (1711) Increases endnote numbers throughout a division group and resets with first endnote in the next division group.

\$LwpResetOptionEachDoc (1708) Increases endnote numbers each time you add a new endnote and continues increasing throughout the document.

\$LwpResetOptionEachPage (1709) Increases a footnote number on a page and resets with the first footnote on the next page. This value only resets footnote numbers on a page. It cannot be used to reset page numbers.

### Usage

Equivalent to choosing Create - Footnote/Endnote, clicking Options, and selecting an option from the "Reset footnote numbers on each" box.

**Word Pro: RestartStyleName property**

{button ,AL('H\_SILVERBULLET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESTARTSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

restartstylevalue = [objectreference].RestartStyleName

[objectreference].RestartStyleName = restartstylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: Rest property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REST\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

restvalue = [objectreference].Rest

[objectreference].Rest = restvalue

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

**Word Pro: Result property**

{button ,AL('H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESULT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

resultvalue = [objectreference].Result

[objectreference].Result = resultvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: ResumePausedMacro property**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RESUMEPAUSEDMACRO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

**Data Type**

[Integer](#)

**Syntax**

[objectreference].ResumePausedMacro = resumepausedmacrovalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: RevisionMarkMode property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISIONMARKMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

revisionmarkmodevalue = [objectreference].RevisionMarkMode

[objectreference].RevisionMarkMode = revisionmarkmodevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: RevisionType property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_REVISION\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISIONTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not a layout was inserted or deleted while in revision marking mode.

### Data Type

Data type for this property is [Integer](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

### Syntax

revisiontypevalue = [objectreference].RevisionType

### Legal values

<u>Value</u>	<u>Effect</u>
\$LwpRevisionTypeNone (1718)	No revisions occurred since the document was created.
\$LwpRevisionTypeInsert (1719)	Indicates that a layout was inserted.
\$LwpRevisionTypeDelete (1720)	Indicates that a layout was deleted.
\$LwpRevisionTypeDontcare (1721)	

### Usage



## **Word Pro: RevMarkCharacter property**

{button ,AL('H\_REVISIONDISPLAY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVMARKCHARACTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

revmarkcharactervalue = [objectreference].RevMarkCharacter

[objectreference].RevMarkCharacter = revmarkcharactervalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: RevMarkPosition property**

{button ,AL('H\_REVISIONDISPLAY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVMARKPOSITION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MarkPosition

### **Syntax**

revmarkpositionvalue = [objectreference].RevMarkPosition

[objectreference].RevMarkPosition = revmarkpositionvalue

### **Legal values**

\$LwpMarkPositionBothSides (585)

\$LwpMarkPositionLeft (583)

\$LwpMarkPositionRight (584)

### **Usage**

## **Word Pro: RevMarkType property**

{button ,AL('H\_REVISIONDISPLAY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVMARKTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

MarkType

### **Syntax**

revmarktypevalue = [objectreference].RevMarkType

[objectreference].RevMarkType = revmarktypevalue

### **Legal values**

\$LwpMarkTypeBars (587)

\$LwpMarkTypeChar (588)

\$LwpMarkTypeNone (586)

### **Usage**

## **Word Pro: RightAlign property**

{button ,AL('H\_BULLET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RIGHTALIGN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

CommandState

### **Syntax**

rightalignvalue = [objectreference].RightAlign

[objectreference].RightAlign = rightalignvalue

### **Legal values**

\$LwpCommandStateOff (151)

\$LwpCommandStateOn (152)

\$LwpCommandStateStyle (153)

### **Usage**

## Word Pro: RightExternalMargin property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_RIGHTEXTERNALMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the amount of margin space that is present to the right of a layout object.

### Data Type

Long

### Syntax

rightexternalmarginvalue = [objectreference].RightExternalMargin

[objectreference].RightExternalMargin = rightexternalmarginvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

This property cannot be set individually for frame layout objects within Word Pro. It is combined with all external margin values in the "Padding around border" setting, located on the Size & Margins panel of the InfoBox.

## Word Pro: RightIntArea property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_RIGHTINTAREA_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Not implemented.

## Data Type

Long

## Syntax

rightintareavalue = [objectreference].RightIntArea

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

**Word Pro: RightMousePropId property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RIGHTMOUSEPROPID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

rightmousepropidvalue = [objectreference].RightMousePropId

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: RightMousePropText property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RIGHTMOUSEPROPTXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[String](#)

### **Syntax**

rightmouseproptextvalue = [objectreference].RightMousePropText

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



**Word Pro: Right property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

rightvalue = [objectreference].Right

[objectreference].Right = rightvalue

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

## **Word Pro: RotationAngle property**

{button ,AL('ERROR:Entrynotfoundinindex',0)} [See list of classes](#)

{button ,AL('H\_ROTATIONANGLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Not Implemented. Indicates, in tenths of degrees, the amount by which the layout has been rotated, relative to its parent.

### **Data Type**

[Integer](#)

### **Syntax**

rotationanglevalue = [objectreference].RotationAngle

[objectreference].RotationAngle = rotationanglevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: RunMacroOnDocEvents property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUNMACROONDOCEVENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables scripts to run on the document open event.

### **Data Type**

[Integer](#)

### **Syntax**

runmacroondoceventsvalue = [objectreference].RunMacroOnDocEvents

[objectreference].RunMacroOnDocEvents = runmacroondoceventsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "Document open scripts" option in the "Disable" box on the General panel of the Word Pro Preferences dialog box. If the value of this property is True (-1), Word Pro allows scripts to run on the document open event. If the value of this property is False (0), Word Pro does not allow scripts to run on the document open event.

## **Word Pro: RunMacroOnLoad property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUNMACROONLOAD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables application startup scripts to run.

### **Data Type**

[Integer](#)

### **Syntax**

runmacroonloadvalue = [objectreference].RunMacroOnLoad

[objectreference].RunMacroOnLoad = runmacroonloadvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "Application startup scripts" option in the "Disable" box on the General panel of the Word Pro Preferences dialog box. If the value of this property is True (-1), Word Pro allows scripts to run when the application loads. If the value of this property is False (0), Word Pro does not allow scripts to run when the application loads.

## **Word Pro: RunOnCloseDoc property**

{button ,AL('H\_AUTORUNMACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUNONCLOSEDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

runoncloseddocvalue = [objectreference].RunOnCloseDoc

[objectreference].RunOnCloseDoc = runoncloseddocvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: RunOnNewDoc property**

{button ,AL('H\_AUTORUNMACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUNONNEWDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

runonnewdocvalue = [objectreference].RunOnNewDoc

[objectreference].RunOnNewDoc = runonnewdocvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: RunOnOpenDoc property**

{button ,AL('H\_AUTORUNMACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUNONOPENDOC\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

runonopenDocvalue = [objectreference].RunOnOpenDoc

[objectreference].RunOnOpenDoc = runonopenDocvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: Saved property**

{button ,AL(^H\_DOCUMENT\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SAVED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

## **Data Type**

[Integer](#)

## **Syntax**

savedvalue = [objectreference].Saved

## **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## **Usage**



## **Word Pro: SaveSnapShot property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVESNAPSHOT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

SnapShot

### **Syntax**

savesnapshotvalue = [objectreference].SaveSnapShot

[objectreference].SaveSnapShot = savesnapshotvalue

### **Legal values**

\$LwpSnapShotNosave (1768)

\$LwpSnapShotSave (1769)

### **Usage**

## Word Pro: ScaleHeight property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SCALEHEIGHT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The height of a graphic object that uses custom graphic scaling.

## Data Type

Long

## Syntax

scaleheightvalue = [objectreference].ScaleHeight

[objectreference].ScaleHeight = scaleheightvalue

## Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

A graphic object will be sized using the ScaleHeight property if the layout object uses custom graphic scaling. You can specify that a layout object use custom graphic scaling by setting the [ScaleMode](#) property to \$LwpScaleTypeCustom.

## Word Pro: ScaleMode property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_JOIN\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SCALEMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[JoinScaleType]

Determines how the width and height of a join is calculated.

[Layout]

Determines how the graphic content of a layout object is scaled.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

scalemodevalue = [objectreference].ScaleMode

[objectreference].ScaleMode = scalemodevalue

## Legal values

[JoinScaleType]

See the explanations for scaling (scalable) and no scaling (fixed) in the Usage section below.

<u>Value</u>	<u>Effect</u>
\$LwpJoinScaleTypeNoScaling (428)	
\$LwpJoinScaleTypeScaling (427)	

[Layout]

<u>Value</u>	<u>Effect</u>
\$LwpScaleTypeCustom (1725)	Custom rectangle (ScaleWdith and ScaleHeight) (checks MaintainAspectRatio)
\$LwpScaleTypeFitInFrame (1723)	Fits into the frame (checks MaintainAspectRatio)
\$LwpScaleTypeOriginalSize (1722)	The graphic is not scaled
\$LwpScaleTypePercentage (1724)	Scales by percentage (see ScalePercentage)

## Usage

A join object can be scalable or fixed.

### Scalable join objects

You can set the ScaleMode property of a scalable join object to scaling or no scaling. Setting the ScaleMode property to scaling for a scalable join object causes the join object's width and height to be a function of the page, table layout, or frame container's height and width and the Percentage property. For more information, see the Percentage property.

Setting the ScaleMode property to no scaling for a scalable join object causes the width and height of a join object to be the same as the width and height properties of the join object. The width and height of the join object does not change as the container's width and height changes.

The width and height of a scalable join with a no scaling setting are fixed. However, you can change the width and height properties of a join object.

#### Fixed join objects

A fixed join object has predefined Word Pro width and height properties. Therefore, you cannot change the width and height of a fixed join object. For a list of join objects that are scalable or fixed, see JoinType property.

## Word Pro: ScalePercentage property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_SCALEPERCENTAGE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates, in tenths of percentage points, the size of a layout object's graphic content in relation to its original size.

## Data Type

Long

## Syntax

scalepercentagevalue = [objectreference].ScalePercentage

[objectreference].ScalePercentage = scalepercentagevalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

A graphic object will be sized using the ScalePercentage property if the layout object uses a percentage graphic scaling setting. You can specify that a layout object use percentage graphic scaling by setting the [ScaleMode](#) property to \$LwpScaleTypePercentage.

## Word Pro: ScaleWidth property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SCALEWIDTH_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The width of a graphic object which uses custom graphic scaling.

### Data Type

Long

### Syntax

scalewidthvalue = [objectreference].ScaleWidth

[objectreference].ScaleWidth = scalewidthvalue

### Legal values

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

A graphic object will be sized using the ScaleWidth property if the layout object uses custom graphic scaling. You can specify that a layout object use custom graphic scaling by setting the [ScaleMode](#) property to \$LwpScaleTypeCustom.

**Word Pro: ScreenPositionX property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SCREENPOSITIONX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates that, if displayed, the icon bar object will be in a horizontal position.

**Data Type**

Long

**Syntax**

screenpositionxvalue = [objectreference].ScreenPositionX

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

Gives the coordinate of the icon bar's horizontal position on the workspace.

**Word Pro: ScreenPositionY property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SCREENPOSITIONY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates that, if displayed, the icon bar object will be in a vertical position.

**Data Type**

[Long](#)

**Syntax**

screenpositionyvalue = [objectreference].ScreenPositionY

**Legal values**

Data type of this property is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

Gives the coordinate of the icon bar's vertical position on the workspace.



## Word Pro: ScriptName property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SCRIPTNAME_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) For internal use only.

## Data Type

String

## Syntax

scriptnamevalue = [objectreference].ScriptName

[objectreference].ScriptName = scriptnamevalue

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

**Word Pro: SectionName property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS;0)} [See list of classes](#)

{button ,AL(^H\_SECTIONNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The internal name of the current section.

**Data Type**

String

**Syntax**

sectionnamevalue = [objectreference].SectionName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

If the insertion point is not located within the boundaries of a named section, this property returns a null string ("").

**Word Pro: Section property**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

sectionvalue = [objectreference].Section

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: SelectedPages property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTEDPAGES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to specify the page numbers that you want to print in a document.

### **Data Type**

String

### **Syntax**

selectedpagesvalue = [objectreference].SelectedPages

[objectreference].SelectedPages = selectedpagesvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

You cannot use this property unless the \$LtsPrintRangeSelectedPages value is set to the PrintRange property. For information, see the PrintRange property.

Equivalent to choosing File - Print, selecting "Selected pages only," clicking "Selected Pages," and selecting the desired options in the Selected Pages dialog box.

You must place quotes around the page numbers to indicate a string expression.

## **Word Pro: SelectionHidden property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTIONHIDDEN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates whether or not the selected text is marked as hidden.

### **Data Type**

[Integer](#)

### **Syntax**

selectionhiddenvalue = [objectreference].SelectionHidden

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

If the text at either end of the selection is marked as hidden, this property is True. If text in the middle of the selection is marked as hidden, but the text at both ends is not marked as hidden, this property returns a value of False.

## **Word Pro: SelectionType property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_CLICKHERE\_CLASS;H\_DIVISION\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS;H\_TEXT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTIONTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates what type of object is selected in the current text stream.

### **Data Type**

Variant (Enumerated)

### **Syntax**

selectiontypevalue = [objectreference].SelectionType

### **Legal values**

\$LwpSelectionTypeNone (1760)

\$LwpSelectionTypeText (1761)

\$LwpSelectionTypeCell ()

\$LwpSelectionTypeFrame ()

### **Usage**

If more than one of these types of objects is in the selection, this property indicates which type of object is uppermost in the focus.

## **Word Pro: SelectTab property**

{button ,AL('H\_RULER\_CLASS;H\_SETTABS\_DIALOG\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTTAB\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[Ruler]

Represents the currently selected tab on the ruler. In addition, this property coordinates the tab setting between the Set Tab dialog box and the ruler.

[SetTabsDialog]

Represents the currently selected tab in the Set Tabs dialog box. In addition, this property coordinates the tab setting between the Set Tab dialog box and the ruler.

## **Data Type**

Integer

## **Syntax**

selecttabvalue = [objectreference].SelectTab

[objectreference].SelectTab = selecttabvalue

## **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## **Usage**

[SetTabsDialog]

This property is an internal Word Pro setting that coordinates the tab setting between the Set Tabs dialog box and the ruler.

## Word Pro: SelectType property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SELECTTYPE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to control whether or not a specific type of layout is selected when the user enters that layout. For example, you can set a property to select the parent layout, the layout itself, or no layout.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

selectypevalue = [objectreference].SelectType

[objectreference].SelectType = selectypevalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpLayoutSelectNone (524)	No layout is selected when the layout is clicked. The currently selected layout or the default layout opens when the layout is clicked.
\$LwpLayoutSelect (525)	The layout itself is selected.
\$LwpLayoutSelectParent (2050)	The parent of the layout is selected when you click on one of the layouts which comprises a NoteLayout.

## Usage



## **Word Pro: ServerFormat property**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SERVERFORMAT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

serverformatvalue = [objectreference].ServerFormat

[objectreference].ServerFormat = serverformatvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: SetContextOfBar property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETCONTEXTOFBAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to query or specify the context in which an icon bar will be displayed when you are working in a specific context. Word Pro has several different contexts (always, in text, in a frame, in columns, in a table or table cell, in a drawing). You can specify in which of these contexts you want the bar displayed.

### **Data Type**

Integer

### **Syntax**

[objectreference].SetContextOfBar = setcontextofbarvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

You can set the display of a specific icon bar set. Equivalent to the value found in the "Bar can be displayed when context is" box in the SmartIcons Setup dialog box.

## **Word Pro: ShowDivisionTabs property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWDIVISIONTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not all divisions tabs, including hidden division tabs, are shown in a document.

### **Data Type**

[Integer](#)

### **Syntax**

showdivisiontabsvalue = [objectreference].ShowDivisionTabs

[objectreference].ShowDivisionTabs = showdivisiontabsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Word Pro can contain hidden tabs. For example, Word Pro always places scripts in a division, but the script division tab never displays unless you set the ShowDivisionTabs property.

Equivalent to choosing File - TeamSecurity and selecting "Display all division tabs in document" on the Other Protection panel.

## **Word Pro: ShowGraphicPreview property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWGRAPHICPREVIEW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables graphic preview in the Import Picture dialog box.

### **Data Type**

[Integer](#)

### **Syntax**

showgraphicpreviewvalue = [objectreference].ShowGraphicPreview

[objectreference].ShowGraphicPreview = showgraphicpreviewvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1). However, this property always retains the last value selected by the user.

### **Usage**

Equivalent to the "Preview" box on the Import Picture dialog box. If the value for this property is True (-1), Word Pro previews the chosen picture before you click OK. If the value for this property is False (0), Word Pro does not preview the chosen picture when you select it.

## **Word Pro: ShowHiddenText property**

{button ,AL('H\_DIVISIONOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWHIDDENTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

showhiddentextvalue = [objectreference].ShowHiddenText

[objectreference].ShowHiddenText = showhiddentextvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: ShowInContext property**

{button ,AL('H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWINCONTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly) Allows you to enable or disable the display of an icon bar object. This property acts as an on/off switch and, if selected, displays a specific set of icons whenever you are working in a specific part of a document. If turned on, the bar will display in its specified context; if turned off, the bar will never display.

### **Data Type**

[Integer](#)

### **Syntax**

[objectreference].ShowInContext = showincontextvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Used when you want to see an icon bar set in a specific context. Equivalent to the value found in the "Bar is enabled to display during its context" box in the SmartIcons Setup dialog box. You can turn it on (displays a specific set whenever you are working in a specific context) or off (turns off the display).

For example, if you turn off the Comment Tools icon bar and then use View - Show/Hide - Comment Tools to redisplay this bar, Comment Tools would then be checked (to display) in the SmartIcons Setup dialog box.

## **Word Pro: ShowMailDisabled property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWMAILDISABLED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the mail notification icon.

### **Data Type**

[Integer](#)

### **Syntax**

showmaildisabledvalue = [objectreference].ShowMailDisabled

[objectreference].ShowMailDisabled = showmaildisabledvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "New mail indicator" option in the "Disable field" on the General panel of the Word Pro Preferences dialog box. If the value of this property is True (-1), Word Pro does not display the new mail icon in the status bar. If the value of this property is False (0), Word Pro displays the new mail icon in the status bar. When you load Word Pro the first time, the mail icon is disabled to aid startup speed.

## **Word Pro: ShowNoWelcomeBox property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWNOWELCOMEBOX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the Welcome dialog box.

### **Data Type**

[Integer](#)

### **Syntax**

shownowelcomeboxvalue = [objectreference].ShowNoWelcomeBox

[objectreference].ShowNoWelcomeBox = shownowelcomeboxvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the "Welcome Dialog" option in the "Disable" field on the General panel of the Word Pro Preferences dialog box. If the value of this property is True (-1), Word Pro displays the Welcome dialog box. If the value of this property is False (0), Word Pro does not display the Welcome dialog box.



**Word Pro: ShowStatistics property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWSTATISTICS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) A flag that turns the compilation of document statistics that displays when Grammar Check finishes on or off.

**Data Type**

Integer

**Syntax**

showstatisticsvalue = [objectreference].ShowStatistics

[objectreference].ShowStatistics = showstatisticsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Use this property to set the flag on or off that displays the document statistics when Grammar Check completes. Equivalent to choosing Edit - Check Grammar, running Grammar Check, and reviewing the document statistics in the Readability Statistics dialog box, which automatically displays when Grammar Check is finished.

### **Word Pro: ShowTabs property**

{button ,AL('H\_DIVISIONINFO\_CLASS;H\_INDEXSECTION\_CLASS;H\_SECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWTABS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

showtabsvalue = [objectreference].ShowTabs

[objectreference].ShowTabs = showtabsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: SizeStyleName property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SIZESTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The size information for a font.

### **Data Type**

String

### **Syntax**

sizestylevalue = [objectreference].SizeStyleName

[objectreference].SizeStyleName = sizestylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: Size property**

{button ,AL('H\_FONT\_CLASS;H\_INDEX\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Points](#)

### **Syntax**

sizevalue = [objectreference].Size

[objectreference].Size = sizevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: SizingUnitName property

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SIZINGUNITNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The name and abbreviation for the name of the units in the SizingUnits property.

### Data Type

String

### Syntax

sizingunitnamevalue = [objectreference].SizingUnitName

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage

Equivalent to the SizingUnits property. The SizingUnits property can have four values; therefore, this property can have four values:

<u>SizingUnits property value</u>	<u>SizingUnitName property value (language specific)</u>
\$LtsScaleModeCentimeter (1056964840)	Centimeters (cm)
\$LtsScaleModeInch (1056964838)	Inches (in)
\$LtsScaleModePoint (1056964837)	Points (pts)
\$LtsScalePica (1728)	Picas (pi)

## **Word Pro: SizingUnits property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SIZINGUNITS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default scaling unit chosen by the user in the Word Pro Preferences dialog box.

### **Data Type**

Variant (Enumerated)

### **Syntax**

sizingunitsvalue = [objectreference].SizingUnits

[objectreference].SizingUnits = sizingunitsvalue

### **Legal values**

\$LtsScaleModeCentimeter (1056964840) Equivalent to "Centimeters (cm)."

\$LtsScaleModeInch (1056964838) Equivalent to "Inches (in)."

\$LtsScaleModePoint (1056964837) Equivalent to "Points (pts)."

\$LwpScaleModePica (1728) Equivalent to "Picas (pi)."

### **Usage**

Equivalent to the options in the "Measure in" list box on the General panel of the Word Pro Preferences dialog box.

This property does not correspond to the Custom measurement dialog boxes for line spacing and paragraph spacing.

## **Word Pro: Skipped property**

{button ,AL('H\_BULLET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SKIPPED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

CommandState

### **Syntax**

skippedvalue = [objectreference].Skipped

[objectreference].Skipped = skippedvalue

### **Legal values**

\$LwpCommandStateOff (151)

\$LwpCommandStateOn (152)

\$LwpCommandStateStyle (153)

### **Usage**

## **Word Pro: SkipWordMode property**

{button ,AL('H\_ATTRIBUTES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SKIPWORDMODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

skipwordmodevalue = [objectreference].SkipWordMode

[objectreference].SkipWordMode = skipwordmodevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



### **Word Pro: SmallCaps property**

{button ,AL('H\_FONT\_CLASS;H\_FONTMETRICS\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SMALLCAPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

smallcapsvalue = [objectreference].SmallCaps

[objectreference].SmallCaps = smallcapsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: SmallFileFormat property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SMALLFILEFORMAT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

smallfileformatvalue = [objectreference].SmallFileFormat

[objectreference].SmallFileFormat = smallfileformatvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: SmartFill property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SMARTFILL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) An instance of the SmartFillCollection class. Use this object to access the SmartFill object for the current session of Word Pro.

### **Data Type**

[SmartFillCollection](#)

### **Syntax**

[objectreference].SmartFill = smartfillvalue

smartfillvalue = [objectreference].SmartFill

### **Legal values**

Always contains an instance of the SmartCorrectCollection class.

### **Usage**

## **Word Pro: SmartLevel property**

{button ,AL('H\_NUMBERING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SMARTLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

CommandState

### **Syntax**

smartlevelvalue = [objectreference].SmartLevel

[objectreference].SmartLevel = smartlevelvalue

### **Legal values**

\$LwpCommandStateOff (151)

\$LwpCommandStateOn (152)

\$LwpCommandStateStyle (153)

### **Usage**

**Word Pro: SnapshotOffset property**

{button ,AL('H\_INDEX\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SNAPSHOTOFFSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

snapshotoffsetvalue = [objectreference].SnapshotOffset

[objectreference].SnapshotOffset = snapshotoffsetvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: SnapshotPath property**

{button ,AL('H\_INDEX\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SNAPSHOTPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

snapshotpathvalue = [objectreference].SnapshotPath

[objectreference].SnapshotPath = snapshotpathvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: SnapshotSize property**

{button ,AL('H\_INDEX\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SNAPSHOTSIZE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

snapshotsizevalue = [objectreference].SnapshotSize

[objectreference].SnapshotSize = snapshotsizevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: SortNumbers property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTNUMBERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies whether words or numbers are positioned first or last in a row.

### **Data Type**

Variant (Enumerated)

SortNumberOrder

### **Syntax**

sortnumbersvalue = [objectreference].SortNumbers

[objectreference].SortNumbers = sortnumbersvalue

### **Legal values**

\$LwpSortNumberOrderFirst (1770) Specifies if a word or number is positioned first in a row.

\$LwpSortNumberOrderLast (1771) Specifies if a word or number is positioned last in a row.

### **Usage**

This property specifies whether words or numbers are positioned first or last in a row. For example, if you select "First" in a Last-Name First-Name "Name" field, Word Pro lists the names alphabetically by last name. If you select "Last" in a Last-Name First-Name "Name" field, Word Pro lists the names alphabetically by first name.

Equivalent to choosing Sort - Text and selecting an option in the "Word" box.



**Word Pro: SortOrder property**

{button ,AL('H\_SORTKEY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to specify an ascending (A - Z or 0 - 9) or a descending (Z - A or 9 - 0) sort order.

**Data Type**

Variant (Enumerated)

Sort

**Syntax**

sortordervalue = [objectreference].SortOrder

[objectreference].SortOrder = sortordervalue

**Legal values**

\$LtsSortAscending (1056964770) Sorts the data from A to Z or 0 to 9.

\$LtsSortDescending (1056964772) Sorts the data from Z to A or 9 to 0.

**Usage**

Equivalent to choosing Text - Sort and selecting "Ascending" or "Descending" in the desired sort level "Order" box.

## **Word Pro: SortType property**

{button ,AL('H\_SORTKEY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

SortType

### **Syntax**

sorttypevalue = [objectreference].SortType

[objectreference].SortType = sorttypevalue

### **Legal values**

\$LwpSortTypeAlphanumeric (1772)

\$LwpSortTypeNumeric (1773)

### **Usage**

## **Word Pro: SortWordOption property**

{button ,AL('H\_SORTKEY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTWORDOPTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies whether to sort text by all words, the first or second word, or by any word you select.

### **Data Type**

Variant (Enumerated)

SortWhichWord

### **Syntax**

sortwordoptionvalue = [objectreference].SortWordOption

[objectreference].SortWordOption = sortwordoptionvalue

### **Legal values**

\$LwpSortWhichWordAllwords (1776) Word Pro sorts every character in the chosen field.

\$LwpSortWhichWordFirstword (1774) Word Pro sorts the first word.

\$LwpSortWhichWordLastword (1775) Word Pro sorts the last word.

\$LwpSortWhichWordOther (1777) Word Pro sorts on the word selected from the dialog box.

### **Usage**

Equivalent to choosing Text - Sort and selecting "First," "Last," "All," or "Other" from the desired sort level "Word" box.

**Word Pro: SortWord property**

{button ,AL('H\_SORTKEY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to sort the second word from the end or the second word from the beginning.

**Data Type**

[Integer](#)

**Syntax**

sortwordvalue = [objectreference].SortWord

[objectreference].SortWord = sortwordvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Equivalent to choosing Text - Sort, selecting "Other" from one of the desired "Word" boxes, and selecting a number from the "Word on which to sort" box.

## **Word Pro: SpaceAbove property**

{button ,AL(^H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS;0)} [See list of classes](#)

{button ,AL(^H\_SPACEABOVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

spaceabovevalue = [objectreference].SpaceAbove

[objectreference].SpaceAbove = spaceabovevalue

### **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: SpaceBelow property**

{button ,AL('H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPACEBELOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

spacebelowvalue = [objectreference].SpaceBelow

[objectreference].SpaceBelow = spacebelowvalue

### **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: SpacesBetweenSentences property**

{button ,AL('H\_GRAMMAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPACESBETWEENSENTENCES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates the number of spaces between sentences that will be allowed in a sentence in Grammar Check.

### **Data Type**

[Integer](#)

### **Syntax**

spacesbetweensentencesvalue = [objectreference].SpacesBetweenSentences

[objectreference].SpacesBetweenSentences = spacesbetweensentencesvalue

### **Legal values**

The legal values for this property are 1 or 2.

### **Usage**

Use this property to set the number of spaces between sentences, either 1 or 2. Equivalent to choosing Edit - Check Grammar, clicking Options, and 1 or 2 in the "Number of spaces between sentences" field, on the Grammatical Style panel.

## **Word Pro: SpacingStyleName property**

{button ,AL('H\_PARAGRAPHSTYLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPACINGSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

spacingstylevalue = [objectreference].SpacingStyleName

[objectreference].SpacingStyleName = spacingstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## Word Pro: SpacingUnitName property

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPACINGUNITNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The abbreviation for the name of the units in the SpacingUnits property.

### Data Type

String

### Syntax

spacingunitnamevalue = [objectreference].SpacingUnitName

### Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### Usage

Equivalent to the SpacingUnits property. The SpacingUnits property can have four values; therefore, this property can have four values:

<u>SpacingUnits property value</u>	<u>SpacingUnitName property value (language specific)</u>
\$LtsScaleModeCentimeter (1056964840)	cm
\$LtsScaleModeInch (1056964838)	in
\$LtsScaleModePoint (1056964837)	pts
\$LtsScalePica (1728)	pi

## Word Pro: SpacingUnits property

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPACINGUNITS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The scaling unit used for displaying spacing measurements in Word Pro.

### Data Type

Variant (Enumerated)

### Syntax

spacingunitsvalue = [objectreference].SpacingUnits

[objectreference].SpacingUnits = spacingunitsvalue

### Legal values

\$LtsScaleModeCentimeter (1056964840) Default for Japanese product.

\$LtsScaleModeInch (1056964838)

\$LtsScaleModePoint (1056964837)

\$LwpScaleModePica (1728) Default.

### Usage

Equivalent to the Custom measurement dialog boxes for line spacing and paragraph spacing. Not equivalent to the "Measure in" list box on the General panel of the Word Pro Preferences dialog box. To change the default scaling unit, use the ScalingUnits property on the UserInterfacePrefs class.

## **Word Pro: SpellCheckIncludesOtherTextStreams property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLCHECKINCLUDESOTHERTEXTSTREAMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is not used in Word Pro.

### **Data Type**

[Integer](#)

### **Syntax**

spellcheckincludesothertextstreamsvalue = [objectreference].SpellCheckIncludesOtherTextStreams

[objectreference].SpellCheckIncludesOtherTextStreams = spellcheckincludesothertextstreamsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This property is not used in Word Pro.

## **Word Pro: SpellCheckInitialCaps property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLCHECKINITIALCAPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables Spell Check to skip words that have the first letter capitalized.

### **Data Type**

[Integer](#)

### **Syntax**

spellcheckinitialcapsvalue = [objectreference].SpellCheckInitialCaps

[objectreference].SpellCheckInitialCaps = spellcheckinitialcapsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "Check words with initial caps" option in the Spell Check Options dialog box. If the value is False (0), Spell Check skips words with initial caps. If the value is True (-1), Spell Check checks words that have initial caps.

## **Word Pro: SpellCheckRepeatedWords property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLCHECKREPEATEDWORDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables Spell Check to skip duplicated words.

### **Data Type**

[Integer](#)

### **Syntax**

spellcheckrepeatedwordvalue = [objectreference].SpellCheckRepeatedWords

[objectreference].SpellCheckRepeatedWords = spellcheckrepeatedwordvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "Check for repeated words" option in the Spell Check Options dialog box. If the value is False (0), Spell Check skips repeated words if they are spelled correctly. If the value is True (-1), Spell Check flags duplicated words.

## **Word Pro: SpellCheckStartsAtBeginning property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLCHECKSTARTSATBEGINNING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is not used by Word Pro.

### **Data Type**

[Integer](#)

### **Syntax**

spellcheckstartsatbeginningvalue = [objectreference].SpellCheckStartsAtBeginning

[objectreference].SpellCheckStartsAtBeginning = spellcheckstartsatbeginningvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

This property is not used by Word Pro.

## **Word Pro: SpellCheckUserDictAlternatives property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLCHECKUSERDICTALTERNATIVES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables Spell Check to search the user dictionary for spelling alternatives.

### **Data Type**

[Integer](#)

### **Syntax**

spellcheckuserdictalternativesvalue = [objectreference].SpellCheckUserDictAlternatives

[objectreference].SpellCheckUserDictAlternatives = spellcheckuserdictalternativesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the "Include user dictionary alternatives" option in the Spell Check Options dialog box. If the value for this property is True (-1), Spell Check searches the user dictionary for spelling alternatives. If the value for this property is False (0), Spell Check searches only the current language dictionary for spelling alternatives.

## **Word Pro: SpellCheckWordsWithNums property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLCHECKWORDSWITHNUMS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables Spell Check to check words with numbers in them.

### **Data Type**

[Integer](#)

### **Syntax**

spellcheckwordswithnumsvvalue = [objectreference].SpellCheckWordsWithNums

[objectreference].SpellCheckWordsWithNums = spellcheckwordswithnumsvvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is True (-1).

### **Usage**

Equivalent to the "Check words with numbers" box in the Spell Check Options dialog box. If the value is True (-1), Spell Check checks words that include numbers (for example, te34st). If the value for this property is False (0), Spell Check skips words that include numbers.



## **Word Pro: SplitPercentage property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPLITPERCENTAGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) When you split the .MDI window for a special view, this property is the percentage of the .MDI window that you want the new window to use.

### **Data Type**

[Integer](#)

### **Syntax**

splitpercentagevalue = [objectreference].SplitPercentage

[objectreference].SplitPercentage = splitpercentagevalue

### **Legal values**

Default is from 1 to 99.

### **Usage**

Equivalent to the View - Split Top-Bottom and View - Split Left-Right menu options in Word Pro.

If the SplitWindow property is True (-1), then the window is split when it is created. You can use this property to set the percentage of the .MDI window for the new window. Other important properties include HorizontalSplitWindow and VerticalSplitWindow. You can also refer to the NewWindow method of WPAApplication.

### **Word Pro: StartColumns property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_STARTCOLUMNS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

startcolumnsvalue = [objectreference].StartColumns

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: StartingColOfSelection property**

{button ,AL(`H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_STARTINGCOLOFSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of the first column included in a selection of table cells.

**Data Type**

[Integer](#)

**Syntax**

startingcolofselectionvalue = [objectreference].StartingColOfSelection

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The column ID is a zero based value, which means that the first column in a table has a row ID value of zero.

### **Word Pro: StartingColStringOfSelection property**

{button ,AL('H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STARTINGCOLSTRINGOFSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the string containing the currently selected text for a range of cells.

### **Data Type**

String

### **Syntax**

startingcolstringofselectionvalue = [objectreference].StartingColStringOfSelection

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: StartingNumber property**

{button ,AL(`H\_ENDNOTEDIVISIONGROUPNUM\_CLASS;H\_ENDNOTEDIVISIONNUM\_CLASS;H\_ENDNOTEDOCNUM\_CLASS;H\_FOOTNOTENUMBERING\_CLASS;H\_FOOTNOTENUMOPT\_CLASS';0)} [See list of classes](#)

{button ,AL(`H\_STARTINGNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The first number used to reference an endnote in a document.

### **Data Type**

[Integer](#)

### **Syntax**

startingnumbervalue = [objectreference].StartingNumber

[objectreference].StartingNumber = startingnumbervalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

The first number used to reference an endnote number can start with any number. If you combine endnotes that are located in different documents into a single document, you can create a continuous endnote numbering scheme by setting the StartingNumber property to a specific endnote in the document.

Equivalent to choosing Create - Footnote/Endnote, clicking Options, and selecting a number in the "Starting at" box on the Numbering panel.

### **Word Pro: StartingRowOfSelection property**

{button ,AL(`H\_BASETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_STARTINGROWOFSELECTION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the number of the first row included in a selection of table cells.

### **Data Type**

[Integer](#)

### **Syntax**

startingrowofselectionvalue = [objectreference].StartingRowOfSelection

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

The row ID is a zero based value, which means that the first row in a table has a row ID value of zero.

### **Word Pro: StartRow property**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_MARKER\_CLASS;H\_POWERFIELD\_CLASS;H\_RUBYMARKER\_CLASS;H\_TABLEMARKER\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_STARTROW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

startrowvalue = [objectreference].StartRow

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: Start property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLASSES;H\_TABLEONLYCONT\_CLASS;H\_USEWHEN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_START\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

## Data Type

Variant (Enumerated)

StartType

## Syntax

startvalue = [objectreference].Start

[objectreference].Start = startvalue

## Legal values

\$LwpStartTypeNextevenpage (1827)

\$LwpStartTypeNextoddpager (1826)

\$LwpStartTypeNextpage (1824)

\$LwpStartTypeThispage (1825)

## Usage



## **Word Pro: StateID property**

```
{button ,AL(^H_CLICKHERE_CLASS;H_DIVISION_CLASS;H_MARKER_CLASS;H_POWERFIELD_CLASS;H_RUBYMARKER_CLASS;H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL(^H_STATEID_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only)

## **Data Type**

Long

## **Syntax**

stateidvalue = [objectreference].StateID

## **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## **Usage**

**Word Pro: StatusBarVisible property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBARVISIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A flag that tells you whether or not the StatusBar is turned on.

**Data Type**

[Integer](#)

**Syntax**

statusbarvisiblevalue = [objectreference].StatusBarVisible

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default value is True (-1).

**Usage**

If you are in Clean Screen mode, the value is False (0).

## **Word Pro: StatusSpellReplaceAll property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSSPELLREPLACEALL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables Spell Check to replace the currently highlighted word or to replace all words.

### **Data Type**

[Integer \(Bool\)](#)

### **Syntax**

statusspellreplaceallvalue = [objectreference].StatusSpellReplaceAll

[objectreference].StatusSpellReplaceAll = statusspellreplaceallvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the Replace and Replace All buttons on the Spell Check bar.

If the value is False (0), Spell Check replaces only the currently highlighted word. If the value is True (-1), Spell Check replaces all instances of this error with the selected word or with your edit in the "Word in question" box.

## Word Pro: StyleExceptions property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_CLICKHERE\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTEELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLEEXCEPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Provides a Long numeric value that indicates which style attributes of an object have been overridden.

## Data Type

[Long](#)

## Syntax

styleexceptionsvalue = [objectreference].StyleExceptions

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

[Layout]

LwpLayStyOverSize (&H1)  
LwpLayStyOverPlacement (&H2)  
LwpLayStyOverSizeAndPlacement (&H3)  
LwpLayStyOverMargins (&H4)  
LwpLayStyOverBorders (&H8)  
LwpLayStyOverBackground (&H10)  
LwpLayStyOverJoins (&H20)  
LwpLayStyOverShadow (&H40)  
LwpLayStyOverTabs (&H80)  
LwpLayStyOverScript (&H100)  
LwpLayStyOverNumerics (&H200)  
LwpLayStyOverColumns (&H400)  
LwpLayStyOverScaling (&H800)  
LwpLayStyOverRotation (&H1000)  
LwpLayStyOverLeaders (&H2000)  
LwpLayStyOverOrientation (&H4000)  
LwpLayStyOverMisc (&H8000)  
LwpLayStyOverChildren (&H10000)  
LwpLayStyOverContents (&H20000)

[Text]

None = &H0  
Face = &H1  
Size = &H2  
Attributes = &H4

bold italics strikethru super sub smallcaps underline single and double underline upper lower

Font = &H8

everything else hidden protected color

Alignment = &H10

Indent = &H20

Spacing = &H40

Paragraph and line

Borders = &H80

Breaks = &H100

Bullet = &H200

Numbering = &H400

Tabs = &H800

Kinsoku = &H1000

Charborder = &H2000

Amikake = &H4000

**Word Pro: StyleName property**

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_FORMATPREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

stylenamevalue = [objectreference].StyleName

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: StylePath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLEPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default path (drive and directory) for Word Pro SmartMaster templates.

### **Data Type**

String

### **Syntax**

stylepathvalue = [objectreference].StylePath

[objectreference].StylePath = stylepathvalue

### **Legal values**

A valid path including drive and directory.

### **Usage**

Equivalent to the "SmartMaster" value on the Locations panel of the Word Pro Preferences dialog box. In the Word Pro interface, "SmartMaster" can contain multiple paths. You can use this property to clear all paths before setting the default or first SmartMaster path. You can use the property, StylePaths, to read multiple SmartMaster paths entered by the user.

## **Word Pro: StyleSheetFullPath property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLESHEETFULLPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

stylesheetfullpathvalue = [objectreference].StyleSheetFullPath

[objectreference].StyleSheetFullPath = stylesheetfullpathvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## **Word Pro: StyleSheetName property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLESHEETNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The default SmartMaster file name.

### **Data Type**

String

### **Syntax**

stylesheetnamevalue = [objectreference].StyleSheetName

### **Legal values**

Valid .MWP file name. Can include a drive and directory.

### **Usage**

Equivalent to the "Plain Document SmartMaster" value on the Default files panel of the Word Pro Preferences dialog box.

## **Word Pro: StyleSheetPath property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLESHEETPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

String

### **Syntax**

stylesheetpathvalue = [objectreference].StyleSheetPath

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: ReviewVersions method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_REVIEWVERSIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVIEWVERSIONS\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].ReviewVersions()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: ReviseAcceptAll method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISEACCEPTALL\_METHOD\_EXSCRIPT',1)} [See example](#)

Accepts revisions according to the instructions provided in the parameters. You can accept revisions for the current paragraph or the entire document, from a single editor or all editors, and in one area of a document or the entire document.

### Syntax

[objectreference].ReviseAcceptAll(ReviseCurrentPara, [EditorName,] [MarkerName])

### Parameters

#### *ReviseCurrentPara*

Allows you to specify whether you want to accept the revisions for the current paragraph (True) or for the entire document (False). Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0).

#### *EditorName*

A String expression specifying the name of the editor whose revisions you want to accept. Optional parameter. If you do not provide a value for this parameter, Word Pro accepts all the revisions without regard to their source.

#### *MarkerName*

A String expression specifying the name of a marker which identifies a part of the document in which you want to accept revisions. Use this parameter when you mark an area of a document for revision and you want to accept revisions only in the marked area. Optional parameter.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: ReviseCancelAll method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISECANCELALL\_METHOD\_EXSCRIPT',1)} [See example](#)

Rejects revisions according to the instructions provided in the parameters. You can reject revisions for the current paragraph or the entire document, from a single editor or all editors, and in one area of a document or the entire document.

### Syntax

[objectreference].ReviseCancelAll(ReviseCurrentPara, [EditorName,] [MarkerName])

### Parameters

#### *ReviseCurrentPara*

Allows you to specify whether you want to reject the revisions for the current paragraph (True) or for the entire document (False). Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants of True (-1) and False (0).

#### *EditorName*

A String expression specifying the name of the editor whose revisions you want to reject. Optional parameter. If you do not provide a value for this parameter, Word Pro will reject all the revisions without regard to their source.

#### *MarkerName*

A String expression specifying the name of a marker which identifies a part of the document in which you want to reject revisions. Use this parameter when you mark an area of a document for revision and you want to reject revisions only in the marked area. Optional parameter.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: RevisionAcceptLayoutChange method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_REVISIONACCEPTLAYOUTCHANGE_METHOD_EXSCRIPT',1)} See example
```

Accepts revision markup changes made in a Layout object.

### Syntax

```
[objectreference].RevisionAcceptLayoutChange(ReviseAll, [EditorName])
```

### Parameters

#### *ReviseAll*

Data type is Integer. Indicates whether or not all revisions to a layout will be accepted.

#### *EditorName*

An optional String data type representing the name of the editor making the revisions to the Layout object.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: RevisionAccept method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISIONACCEPT\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RevisionAccept(ReviseAll, [EditorName,] [MarkerName])

### **Parameters**

*ReviseAll*

Data type is Boolean.

*EditorName*

Data type is String. Optional parameter.

*MarkerName*

Data type is String. Optional parameter.

### **Return value**

### **Usage**

## Word Pro: RevisionCancelLayoutChange method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_REVISIONCANCELLAYOUTCHANGE_METHOD_EXSCRIPT',1)} See example
```

Cancels revision markup changes made in a Layout object.

### Syntax

```
[objectreference].RevisionCancelLayoutChange(ReviseAll, [EditorName])
```

### Parameters

#### *ReviseAll*

Data type is Integer. Indicates whether or not all revisions to a layout will be accepted.

#### *EditorName*

An optional String data type representing the name of the editor making the revisions to the Layout object.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage



## **Word Pro: RevisionCancel method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_REVISIONCANCEL\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RevisionCancel(ReviseAll, [EditorName,] [MarkerName])

### **Parameters**

*ReviseAll*

Data type is Boolean.

*EditorName*

Data type is String. Optional parameter.

*MarkerName*

Data type is String. Optional parameter.

### **Return value**

### **Usage**

## **Word Pro: RunAutoNewMacro method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RUNAUTONEWMACRO\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RunAutoNewMacro()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RunAutoOpenMacro method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_RUNAUTOOPENMACRO\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].RunAutoOpenMacro()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: RunScript method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_RUNSCRIPT\_METHOD\_EXSCRIPT',1)} [See example](#)

Runs the specified Word Pro script.

### **Syntax**

[objectreference].RunScript(ScriptModule, ScriptFunction)

### **Parameters**

*ScriptModule*

A String expression specifying the name and path of the file which contains the script you want to run.

*ScriptFunction*

A String expression specifying the script you want to run.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: Run method**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_RUN\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Run(p1)

### **Parameters**

*p1*

Data type is Variant.

### **Return value**

### **Usage**

## Word Pro: SaveAsToNotes method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEASTONOTES\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves the active document with a different name or file type.

### Syntax

[objectreference].SaveAsToNotes([UID,] [Attached,] [Field,] [Database,][Server,] [FileType])

### Parameters

#### *UID*

An optional String value which indicates the .

#### *Attached*

An optional String value which indicates the .

#### *Field*

An optional String value which indicates the .

#### *Database*

An optional String value which indicates the .

#### *Server*

An optional String value which indicates the .

#### *FileType*

An optional String value which indicates the .

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## Word Pro: SaveAs method

{button ,AL('H\_DOCUMENT\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEAS\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves the active document with a different name or file type.

### Syntax

[objectreference].SaveAs([DocName,] [Location,] [DocType,] [Backup,][AddToLastFileOpenList,] [SaveCopyAs])

### Parameters

#### *DocName*

An optional String value which indicates the name under which the document will be saved.

#### *Location*

An optional String value which indicates the path or directory in which the document will be saved.

#### *DocType*

An optional String value that indicates the file type in which the document will be saved. A null string saves the document as a Word Pro file. Some of the usual file types are listed in the table below, but each user's list of available file types is derived from the list of text filters installed during the Word Pro installation.

DCA/RFT	Lotus Manuscript 2.x	MS Word for Windows 1.0
DIF	Lotus Organizer 1.x	MS Word for Windows 2.0
DisplayWrite	Lotus Word Pro	MS Word for Windows 6.0
HTML	Lotus Word Pro SmartMaster	MS Word for Windows95 7.0
Lotus 1-2-3	MS Excel	MS WordPad 1.0
Lotus 1-2-3 for OS/2	MS Excel 3.0	OfficeWriter 4,5,6
Lotus 1-2-3 R3	MS Excel 4.0	Rich Text Format(RTF)
Lotus 1-2-3 R4,5	MS Excel 5.0	SAMNA Word
Lotus 1-2-3 R6	MS Excel 7.0	WordPerfect 5.0
Lotus Ami Pro	MS Windows Write 3.x	WordPerfect 5.1
Lotus Ami Pro 3.x Macro	MS Word for DOS 3,4,5,6	WordPerfect 6.x
Lotus Ami Pro 3.x Styles	MS Word for OS/2	WordStar 2000 R3

#### *Backup*

An optional Integer value that indicates whether or not Word Pro should make a backup copy of this file before saving.

#### *AddToLastFileOpenList*

An optional Integer value that indicates whether or not Word Pro should add this file's name to list of recently opened files on the File menu. Default is False, indicating that the file name will not be added.

#### *SaveCopyAs*

An optional Integer value that indicates whether Word Pro should leave open the original document or the newly saved document. Default is False, indicating that Word Pro should close the original and leave open the newly saved file.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## **Word Pro: SaveData method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SaveData(P1)

### **Parameters**

*P1*

Data type is Long.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: SaveDivision method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEDIVISION\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves a division as a separate file. Equivalent to clicking the right mouse button on a division divider tab, choosing Division Properties, and clicking Save as File.

### **Syntax**

[objectreference].SaveDivision()

### **Parameters**

### **Return value**

### **Usage**

**Word Pro: SaveEnvelopeMaster method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEENVELOPEMASTER\_METHOD\_EXSCRIPT',1)} [See example](#)

Not implemented in Word Pro 97.

## Word Pro: SaveGlossary method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEGLOSSARY\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves the currently active Glossary data file. You can save the glossary or you can save a copy of the glossary under a different name or in another file type.

### Syntax

[objectreference].SaveGlossary(FilePath, FileType[, AddToLastFileOpenList] [, SaveCopyAs])

### Parameters

#### *FilePath*

A String expression specifying the path and name for the Glossary file.

#### *File Type*

A String expression specifying a file type. Use this to change the file type of the document when you want to use or read it in another application.

#### *AddToLastFileOpenList*

This parameter allows you to show or hide the glossary file from the last file opened list. Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0). Optional parameter. Default is False, which hides the Glossary file from the list.

#### *SaveCopyAs*

A String expression specifying a new name for the Glossary file. Use this to save a copy of the Glossary file under a different name.

### Return value

None

### Usage

## **Word Pro: SaveMacro method**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEMACRO\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SaveMacro()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: SaveSnapshot method**

{button ,AL('H\_GRAPHIC\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVESNAPSHOT\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SaveSnapshot(P1)

### **Parameters**

P1

Data type is Long.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: SaveThumbnailBitmap method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVETHUMBNAILBITMAP\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SaveThumbnailBitmap(Filename)

### **Parameters**

*Filename*

An optional String expression that indicates the name of the file.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SaveToStorageComplete method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVETOSTORAGECOMPLETE\_METHOD\_EXSCRIPT',1)} [See example](#)

Used to complete the use of the SaveToStorage method.

### **Syntax**

[objectreference].SaveToStorageComplete(pIStorage)

### **Parameters**

*pIStorage*

A Numeric expression which specifies the IStorage space to which you want to save the embedded Word Pro object. A null value indicates that you want to keep using the original IStorage space. Any other value specifies which IStorage you want to switch to once the save is complete. Data type is Long.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SaveToStorage method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVETOSTORAGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves an embedded Word Pro OLE object to the specified IStorage.

### **Syntax**

[objectreference].SaveToStorage(pIStorage, FileType, SameStorageAsLoad)

### **Parameters**

#### *pIStorage*

A Numeric expression which specifies the IStorage space to which you want to save the embedded Word Pro object. Data type is Long.

#### *FileType*

A String expression indicating the type of Word Pro or Ami Pro object being saved. The file types include:

- Lotus Ami Pro
- Lotus Ami Pro 3.x Macro
- Lotus Ami Pro 3.x Styles
- Lotus Word Pro
- Lotus Word Pro SmartMaster

#### *SameStorageAsLoad*

Data type is Boolean. Indicates whether the IStorage named in pIStorage is the same as the IStorage space from which Word Pro loaded this Word Pro OLE object.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

When you use this method, you must also call the SaveToStorageComplete method.



## **Word Pro: SaveUserDefaults method**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEUSERDEFAULTS\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves the following settings from the Word Pro Preferences dialog box and the Welcome dialog box:

- number of undo levels
- number of recent files
- all markup options
- list of recently used SmartMaster templates that display in the Open dialog box

### **Syntax**

[objectreference].SaveUserDefaults()

### **Parameters**

None

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: SaveVersion method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SAVEVERSION\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves the selected version as a file. Equivalent to choosing File - Versions and clicking Save as File.

### **Syntax**

[objectreference].SaveVersion(FilePath,Version)

### **Parameters**

*FilePath*

Data type is String.

*Version*

Data type is Long.

### **Return value**

### **Usage**

## **Word Pro: Save method**

{button ,AL(^H\_DOCUMENT\_CLASS;H\_DOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SAVE\_METHOD\_EXSCRIPT',1)} [See example](#)

Saves the currently active document. Equivalent to choosing File - Save.

### **Syntax**

[objectreference].Save()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SelectAStyle method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTASTYLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens the paragraph style menu on the status bar so the user can select a new style.

### **Syntax**

[objectreference].SelectAStyle()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: SelectCell method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTCELL\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the contents of the cell at the insertion point.

**Syntax**

[objectreference].SelectCell()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: SelectColumn method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTCOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the column at the insertion point. If cells from more than one column are selected, all the columns represented in that selection are selected.

**Syntax**

[objectreference].SelectColumn()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: SelectCustomIcon method**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTCUSTOMICON\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects a custom icon so that you can query or attach properties to it.

**Syntax**

[objectreference].SelectCustomIcon(GraphicPath, MacroPath)

**Parameters**

*GraphicPath*

Data type is String. Required parameter.

*MacroPath*

Data type is String. Required parameter.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You must select an icon using either this method or the SelectStandardIcon method before you can assign functional properties to it.

**Word Pro: SelectDoc method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTDOC\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the entire active document.

**Syntax**

[objectreference].SelectDoc()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**



## **Word Pro: SelectEntireCellRange method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTENTIRECELLRANGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the cell itself and the contents in the cell. The insertion point must be in the cell. If more than one cell is selected when you call this method, Word Pro selects all the cells and their contents.

### **Syntax**

[objectreference].SelectEntireCellRange()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: SelectEntireColumn method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTENTIRECOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the column which contains the currently active cell. If cells are selected from more than one column, this method selects the entire column for each selected cell.

**Syntax**

[objectreference].SelectEntireColumn()

**Parameters**

None.

**Return value**

None.

**Usage**

## **Word Pro: SelectEntirePColCellRange method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTENTIREPCOLCELLRANGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects both the parallel column cell and its contents. If more than one cell is in the focus, Word Pro selects all the cells and their contents.

### **Syntax**

[objectreference].SelectEntirePColCellRange()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: SelectEntirePColColumn method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTENTIREPCOLCOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the parallel column which has the focus. If text is selected in more than one parallel column, this method selects the entire parallel column for each text selection.

**Syntax**

[objectreference].SelectEntirePColColumn()

**Parameters**

None.

**Return value**

None.

**Usage**

**Word Pro: SelectEntirePColRow method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_SELECTENTIREPCOLROW\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the parallel column row which has the focus.

**Syntax**

[objectreference].SelectEntirePColColumn()

**Parameters**

None.

**Return value**

None.

**Usage**

**Word Pro: SelectEntirePCol method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTENTIREPCOL\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects all the columns and rows in the ParallelColumns object which has the focus. If there is no ParallelColumns object in the focus, this method returns 0, indicating failure.

**Syntax**

[objectreference].SelectEntirePCol()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: SelectEntireRow method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTENTIREROW\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the table row which has the focus and the contents of that row. If cells from more than one row are selected, Word Pro selects all the rows represented in the selection and their contents.

### **Syntax**

[objectreference].SelectEntireRow()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

### **Word Pro: SelectEntireTable method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTENTIRETABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the table which has the focus and its contents. Equivalent to choosing Table - Select - Entire Table.

### **Syntax**

[objectreference].SelectEntireTable()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**



### **Word Pro: SelectMarker method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTMARKER\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the specified marker object.

### **Syntax**

[objectreference].SelectMarker(MarkerName)

### **Parameters**

*MarkerName*

A String expression that specifies the name of the marker object you want to select.

### **Return value**

### **Usage**

## **Word Pro: SelectParagraph method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTPARAGRAPH\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the paragraph in which the insertion point is located.

### **Syntax**

[objectreference].SelectParagraph()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: SelectPColCell method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTPCOLCELL\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the cell in a parallel column in which the insertion point is located.

**Syntax**

[objectreference].SelectPColCell()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: SelectRow method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTROW\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the contents of the table row in which the insertion point is located. Equivalent to choosing Table - Select - Row Contents.

**Syntax**

[objectreference].SelectRow()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: SelectRuler method**

{button ,AL('H\_RULER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTRULER\_METHOD\_EXSCRIPT',1)} [See example](#)

Word Pro calls this method when the ruler displays and selects a ruler mouse filter context on the context stack to intercept mouse messages.

### **Syntax**

[objectreference].SelectRuler()

### **Parameters**

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

The mouse filter context filters mouse move messages and checks to see if a frame is currently being dragged. If a frame is being dragged, Word Pro displays special guides on the ruler.

## **Word Pro: SelectSection method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTSECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the contents within a specified section of the document. Equivalent to clicking the right mouse button on a section tab and choosing Select Section.

### **Syntax**

[objectreference].SelectSection([SectionName])

### **Parameters**

*SectionName*

An optional String expression which specifies the name of the section whose contents you want to select. If you do not provide this parameter, Word Pro selects the contents of the section in which the insertion point is located.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: SelectSentence method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SELECTSENTENCE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the sentence in which the insertion point is located.

**Syntax**

[objectreference].SelectSentence()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: SelectStandardIcon method**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTSTANDARDICON\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects a standard icon so that you can query or attach properties to it.

**Syntax**

[objectreference].SelectStandardIcon(MenuID)

**Parameters**

*MenuID*

Data type is Menu. Required parameter.

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

You must select an icon using either this method or the SelectCustomIcon method before you can assign functional properties to it.



## Word Pro: SelectTableItem method

{button ,AL(^H\_Basetable\_Class;H\_FOOTNOTETABLE\_Class;H\_GLOSSARY\_Class;H\_PARALLELCOLUMNS\_Class;H\_TABLE\_Class;H\_TABLEHEADING\_Class',0)} [See list of classes](#)

{button ,AL(^H\_SELECTTABLEITEM\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to select a specific type of table item. In order to use this method, the insertion point must be inside the table that contains the items to be selected.

### Syntax

[objectreference].SelectTableItem(TableSelection, [p2,] [p3,] [p4,] [EndCol,] [SelectWholeCell])

### Parameters

#### *TableSelection*

Data type is Variant, which allows its value to be one of the string constants below or its numeric equivalent. This parameter specifies which type of table item you want to select.

<u>Value</u>	<u>Effect</u>
\$LwpTableSelectionCell (1899)	The method moves the insertion point into a specified cell.
\$LwpTableSelectionRow (1900)	The method selects a specified range of rows.
\$LwpTableSelectionColumn (1901)	The method selects a specified range of columns.
\$LwpTableSelectionTable (1902)	The method selects the entire table.
\$LwpTableSelectionRange (1903)	The method selects a specified range of cells.

#### *p2*

Data type is Integer. Optional parameter. The purpose of this parameter varies according to the TableSelection parameter value. For details on each TableSelection parameter value, see the Usage section.

#### *p3*

Data type is Integer. Optional parameter. The purpose of this parameter varies according to the TableSelection parameter value. For details on each TableSelection parameter value, see the Usage section.

#### *p4*

Data type is Integer. Optional parameter. The purpose of this parameter varies according to the TableSelection parameter value. For details on each TableSelection parameter value, see the Usage section.

#### *EndCol*

Data type is Integer. Optional parameter. The purpose of this parameter varies according to the TableSelection parameter value. For details on each TableSelection parameter value, see the Usage section.

#### *SelectWholeCell*

Data type is Integer. Optional parameter. The purpose of this parameter varies according to the TableSelection parameter value. For details on each TableSelection parameter value, see the Usage section.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

#### **\$LwpTableSelectionCell (1899)**

When use you use this value in the TableSelection parameter, the method moves the insertion point into a specific cell. You must also provide values for the P2 and P3 parameters.

#### *P2*

This parameter is an optional Integer value which represents the Row ID of the cell into which the insertion point will move. There is no default value.

#### *P3*

This parameter is an optional Integer value which represents the Column ID of the cell into which the insertion point will move. There is no default value.

None of the other parameters are used when you use this value for the TableSelection parameter.

The Row ID and Column ID are zero based values. In order to select the first cell in a table, use a Row ID value of 0 and a Column ID value of 0.

#### **\$LwpTableSelectionRow (1900)**

When use you use this value in the TableSelection parameter, the method selects a range of rows within a table. You can also provide values for the P2, P3, and P4 parameters. When you do not supply a value for any other parameter, the contents of the cells within the current row are selected.

##### *P2*

This parameter is an optional Integer value which represents the ID of the first row to be included in the selection. Default is 0.

##### *P3*

This parameter is an optional Integer value which represents the ID of the last row to be included in the selection. Default is 0.

##### *P4*

This parameter is a Boolean value which specifies whether whole cells should be selected. Default is False.

None of the other parameters are used when you use this value for the TableSelection parameter.

The Row ID is a zero based value. In order to select the first row of a table, use a Row ID value of 0.

#### **\$LwpTableSelectionColumn (1901)**

When use you use this value in the TableSelection parameter, the method selects a range of columns within a table. You can also provide values for the P2, P3, and P4 parameters. When you do not supply a value for any other parameter, the contents of the cells within the current column are selected.

##### *P2*

This parameter is an optional Integer value which represents the ID of the first column to be included in the selection. Default is 0.

##### *P3*

This parameter is an optional Integer value which represents the ID of the last column to be included in the selection. Default is 0.

##### *P4*

This parameter is a Boolean value which specifies whether whole cells should be selected. Default is False.

None of the other parameters are used when you use this value for the TableSelection parameter.

The Column ID is a zero based value. In order to select the first column of a table, use a Column ID value of 0.

#### **\$LwpTableSelectionTable (1902)**

When use you use this value in the TableSelection parameter, the method selects an entire table. You can also provide a value for the P2 parameter.

##### *P2*

This parameter is an optional Boolean value that specifies whether whole cells should be selected. Default is False.

None of the other parameters are used when you use this value for the TableSelection parameter.

#### **\$LwpTableSelectionRange (1903)**

When use you use this value in the TableSelection parameter, the method selects a range of cells within a table. You can also provide values for the P2, P3, P4, EndCol, and SelectWholeCell parameters. When you do not supply a value for any other parameter, the contents of the current cell are selected.

##### *P2*

This parameter is an optional Integer value which represents the Row ID of the first cell to be selected. Default is 0.

*P3*

This parameter is an optional Integer value which represents the Column ID of the first cell to be selected. Default is 0.

*P4*

This parameter is an optional Integer value which represents the Row ID of the last cell to be selected. Default is 0.

*EndCol*

This parameter is an optional Integer value which represents the Column ID of the last cell to be selected. Default is 0.

*SelectWholeCell*

The SelectWholeCell parameter is an optional Boolean value which specifies whether whole cells should be selected. Default is False.

The Row ID and Column ID are zero based values, which means the first row and column of a table have an ID value of 0.

**Word Pro: SelectTable method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTTABLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the contents of the table in which the insertion point is located. Equivalent to choosing Table - Select - Entire Table Contents.

**Syntax**

[objectreference].SelectTable()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

**Word Pro: SelectWord method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SELECTWORD\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects the word at the insertion point.

**Syntax**

[objectreference].SelectWord()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

A word is comprised of a contiguous string of alphanumeric characters. Punctuation and spaces are seen as the end of a word. If the insertion point is between two spaces, Word Pro selects all the spaces on both sides of the insertion point, as well as the word preceding the spaces.

## Word Pro: Select method

```
{button ,AL(^H_CLICKHERE_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_CLASS',0)}
```

[See list of classes](#)

```
{button ,AL(^H_SELECT_METHOD_EXSCRIPT',1)} See example
```

Selects the type of object specified in the ObjectType parameter.

### Syntax

```
[objectreference].Select(ObjectType)
```

### Parameters

#### *ObjectType*

Specifies what type of object to select. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value.

`$LwpSelectObjectTypeBullet (1757)`

Selects the next bullet object in the text stream.

`$LwpSelectObjectTypeChunk (1750)`

A chunk is comprised of a single word (a group of characters with no spaces) and all the contiguous spaces following that word. If the insertion point is at the beginning, the end, or anywhere within a word, Word Pro selects that word and the spaces which follow it. If the insertion point is between two spaces, Word Pro selects all the spaces following the insertion point to the beginning of the next word. If there is no word between the spaces and the end of the paragraph, Word Pro selects to the end of the paragraph.

`$LwpSelectObjectTypeDocument (1754)`

Selects the contents of the entire document.

`$LwpSelectObjectTypeLevel (1756)`

Undefined.

`$LwpSelectObjectTypeObject (1748)`

Selects the next object of any type.

`$LwpSelectObjectTypeParagraph (1753)`

Selects the paragraph object in which the insertion point is located.

`$LwpSelectObjectTypeSection (1759)`

Selects the contents of the section in which the insertion point is located.

`$LwpSelectObjectTypeSentence (1752)`

Selects the sentence in which the insertion point is located.

`$LwpSelectObjectTypeStream (1755)`

Selects the text stream in which the insertion point is located. A text stream is comprised of all the paragraphs of text and the tables but none of the frames or OLE objects.

`$LwpSelectObjectTypeTombstoneset (1758)`

Undefined.

`$LwpSelectObjectTypeTuna (1751)`

Undefined.

`$LwpSelectObjectTypeWord (1749)`

Selects the word at the insertion point. In this case, a word is comprised of a contiguous string of alphanumeric characters. Punctuation and spaces are seen as the end of a word. If the insertion point is between two spaces, Word Pro selects all the spaces on both sides of the insertion point as well as the word preceding the spaces.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

**Word Pro: SendFrameToBackOne method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SENDFRAMETOBACKONE\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the priority of the selected frame so that it is one level back from its original position.

**Syntax**

[objectreference].SendFrameToBackOne()

**Parameters**

None.

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

To call this method, the "Place frame" option for the frame must be one of the following:

- On all pages
- On left/right pages
- On current page

You can access the "Place frame" option from the Placement panel in the Frame InfoBox.

## **Word Pro: SendFrameToBack method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_SENDFRAMETOBACK\_METHOD\_EXSCRIPT',1)} [See example](#)

Changes the priority of the currently selected frame so that it is behind all the other frames on the page.

### **Syntax**

[objectreference].SendFrameToBack()

### **Parameters**

None.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

To call this method, the "Place frame" option for the frame must be one of the following:

- On all pages
- On left/right pages
- On current page

You can access the "Place frame" option from the Placement panel in the Frame InfoBox.



### **Word Pro: SendMailSelectedText method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SENDMAILSELECTEDTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Uses the selected text as the body of a mail message which it mails using the MAPI application specified in your WIN.INI file. Equivalent to choosing File - TeamMail and selecting "Message with current selection's text as message body."

#### **Syntax**

[objectreference].SendMailSelectedText()

#### **Parameters**

None

#### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

#### **Usage**

## Word Pro: SetArrayProp method

{button ,AL('H\_SILVERBULLET\_CLASS;H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETARRAYPROP\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the properties of an array. This method is defined in the following classes:

[SilverBullet]

[UserInterfacePrefs]

### Syntax

[objectreference].SilverBullet.SetArrayProp(BulletArrayProp,Level,NewValue)

[objectreference].UserInterfacePrefs.SetArrayProp(PrefPropScope,Index,New)

### Parameters

#### *BulletArrayProp*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpBulletArrayPropCumulative (79)

\$LwpBulletArrayPropDivision (81)

\$LwpBulletArrayPropLesser (78)

\$LwpBulletArrayPropLesserspecific (77)

\$LwpBulletArrayPropSection (80)

#### *Level*

Data type is Integer.

#### *NewValue*

Data type is Integer.

#### *PrefPropScope*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpPrefPropScopeFindString (1640) Corresponds to the "Find" box in the Find & Replace bar.

\$LwpPrefPropScopeReplaceString (1641) Corresponds to the "Replace with" box in the Find & Replace bar.

#### *Index*

Data type is Integer. Legal values are 0, 1, 2, and 3..

#### *NewValue*

Data type is String. The strings contains the text that you want to set in the array, in this case, the text you want in the "Find" and "Replace with" boxes.

### Return value

Both SilverBullet and UserInterfacePrefs objects return a Boolean value.

### Usage

[SilverBullet]

[UserInterfacePrefs]

Use this method to set a value in the "Find" and "Replace with" boxes.

## Word Pro: SetButtonText method

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETBUTTONTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the text on a button.

### Syntax

[objectreference].SetButtonText(Button Text, IsHilite)

### Parameters

#### *Button Text*

The text to appear on the button. Data type is String.

#### *IsHilite*

Determines whether any special attributes will be shown in this button's text. Data type is Variant which allows the value to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these constants when you want Word Pro to combine the features listed below. Use the OR operator to combine constants. Valid parameters are:

LwpButtonContentsHilited (&H400) A value that allows the button contents to be highlighted (red in Word Pro).

LwpButtonContentsLeftAligned (&H40) A value that allows the button contents to be left-aligned.

LwpButtonContentsCenterAligned (&H80) A value that allows the button contents to be center-aligned.

LwpButtonContentsRightAligned (&H100) A value that allows the button contents to be right-aligned.

LwpButtonContentsGray (&H200) A value that allows the button contents to be grayed.

### Return value

Data type is Integer. Use 0 unless you want to override the defaults.

### Usage

Generally used in the script for a status bar button's override text event. Use when you want to set, change, or update the button's text.

## **Word Pro: SetData method**

{button ,AL('H\_SCRIPTDATASET\_CLASS;H\_WPDATASET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates a new data set variable and sets its value.

### **Syntax**

[objectreference].SetData(DataName, NewData)

### **Parameters**

#### *DataName*

The variable name in a data set. Data type is String.

#### *NewData*

The new data set item that you are creating. Data type is always String(s).

### **Return value**

String

### **Usage**

When you invoke the SetData method, the NewData parameter updates the value of the currently used DataName parameter. If a NewData parameter does not exist, then this method creates a new variable named DataName and gives it the value of NewData.

## **Word Pro: SetDocumentEpoch method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETDOCUMENTEPOCH\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SetDocumentEpoch()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: SetFieldFormula method

{button ,AL('H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETFIELDFORMULA\_METHOD\_EXSCRIPT',1)} [See example](#)

### Syntax

[objectreference].SetFieldFormula(Formula, [Type])

### Parameters

#### *Formula*

Data type is String.

#### *Type*

Data type is Variant. Optional parameter. Default is 0. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpPFTypeBookmark (2003)

\$LwpPFTypeButton (2007)

\$LwpPFTypeDde (2002)

\$LwpPFTypeDocvar (2011)

\$LwpPFTypeField (2004)

\$LwpPFTypeIndex (2009)

\$LwpPFTypeMarker (2010)

\$LwpPFTypeMergevar (2013)

\$LwpPFTypePrtescape (2008)

\$LwpPFTypeSeq (2005)

\$LwpPFTypeSet (2006)

\$LwpPFTypeToc (2012)

### Return value

### Usage

## **Word Pro: SetFocus method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETFOCUS\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SetFocus()

[objectreference].Setfocus(Control\_ID)

### **Parameters**

*Control\_ID*

Data type is Variant.

### **Return value**

Boolean

### **Usage**

**Word Pro: SetFormula method**

{button ,AL('H\_CELLENGINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETFORMULA\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a formula into the current table cell.

**Syntax**

[objectreference].SetFormula(Row, Column, Formula)

**Parameters***Row*

Allows you to set a formula in a specific row in a table object. Data type is Integer.

*Column*

Allows you to set a formula in a specific column in a table object. Data type is Integer.

*Formula*

Allows you to set the specific formula you want to insert in a table cell, row, or column. Data type is String.

**Return value****Usage**

Equivalent to choosing Table - Insert Formula.



## **Word Pro: SetLastUsedFilter method**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETLASTUSEDFILTER\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SetLastUsedFilter(Type, Filter)

### **Parameters**

#### *FilterType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpFilterTypeGraphic (280)

\$LwpFilterTypeTable (281)

\$LwpFilterTypeText (279)

#### *Filter*

Data type is String.

### **Return value**

### **Usage**

## Word Pro: SetLineOneSide method

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETLINEONESIDE\_METHOD\_EXSCRIPT',1)} [See example](#)

This method only displays in the Script Editor during a recording to reflect the selection of a line style for a specific side of a table.

### Syntax

[objectreference].SetLineOneSide(LinePlacement, LineStyle, LineWidth, LineColor, TableMix)

### Parameters

#### *LinePlacement*

Indicates which side of the table to set a specific line style. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

LwpLinePlacementAllsides (&HF)

LwpLinePlacementBottom (&H8)

LwpLinePlacementLeft (&H1)

LwpLinePlacementRight (&H2)

LwpLinePlacementTop (&H4)

#### *LineStyle*

Allows you to set one of the line styles below to a specific side of a table. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LtsBorderPatternDashDot (1056964659)

\$LtsBorderPatternDashDotDot (1056964660)

\$LtsBorderPatternDashed (1056964662)

\$LtsBorderPatternDot (1056964663)

\$LtsBorderPatternDouble (1056964666)

\$LtsBorderPatternLongDash (1056964661)

\$LtsBorderPatternNone (1056964657)

\$LtsBorderPatternSolid (1056964658)

\$LwpBorderPattern13space (36)

\$LwpBorderPattern31space (37)

\$LwpBorderPatternButtondown (35)

\$LwpBorderPatternButtonup (34)

\$LwpBorderPatternCircle (41)

\$LwpBorderPatternDblThick (51)

\$LwpBorderPatternDblWavy (56)

\$LwpBorderPatternDeco1 (44)

\$LwpBorderPatternDeco2 (45)

\$LwpBorderPatternDeco3 (50)

\$LwpBorderPatternDiagonal (38)

\$LwpBorderPatternPin (47)

\$LwpBorderPatternRain (46)

\$LwpBorderPatternRope (43)

\$LwpBorderPatternRose (48)

\$LwpBorderPatternStar (42)

\$LwpBorderPatternSunf (49)

\$LwpBorderPatternTaro (39)

\$LwpBorderPatternThickDblwavy (58)

\$LwpBorderPatternThickThin (53)

\$LwpBorderPatternThickWavy (57)

\$LwpBorderPatternThinThick (54) \$LwpBorderPatternThinThickThin (52)

\$LwpBorderPatternWavy (55)

\$LwpLtsBorderPatternDot (40)

#### *LineWidth*

Specifies the width for the line style of a specific side of a table. Data type is Twips.

#### *LineColor*

Specifies the line color for the line style of a specific side of a table. Data type is Long.

#### *TableMix*

Allows you to specify if a table contains more than one type of line style. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTableMixAllmixed (1893) Setting this value allows you to assign a different line style to each line of a table cell.

\$LwpTableMixBottommixed (1890) Setting this value allows you to assign a line style to the bottom line of a table cell that is different from the line style of the top, right, and left line of a table cell.

\$LwpTableMixLeftmixed (1891) Setting this value allows you to assign a line style to the left line of a table cell that is different from the line style of the top, right, and bottom line of a table cell.

\$LwpTableMixRightmixed (1892) Setting this value allows you to assign a line style to the right line of a table cell that is different from the line style of the top, bottom, and left line of a table cell.

\$LwpTableMixTopmixed (1889) Setting this value allows you to assign a line style to the top line of a table cell that is different from the line style of the bottom, right, and left line of a table cell.

#### **Return value**

#### **Usage**

## Word Pro: SetLinesAllSides method

{button ,AL('H\_TABLELINE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETLINESALLSIDES\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to simultaneously set a specific line style to all sides of an entire table, or a specific row, column or cell.

### Syntax

[objectreference].SetLinesAllSides(LinesAroundCells, LineStyle, LineWidth, LineColor, OutlineLineStyle, OutlineLineWidth, OutlineLineColor)

### Parameters

#### *LinesAroundCells*

Allows you to specify where the line style should be applied in the table object. Data type is Variant.

- \$LwpTableLineStyleAll (1878)
- \$LwpTableLineStyleCols (1882)
- \$LwpTableLineStyleCustom (1886)
- \$LwpTableLineStyleInnercols (1887)
- \$LwpTableLineStyleInnerRowscols (1888)
- \$LwpTableLineStyleMixed (1885)
- \$LwpTableLineStyleNone (1877)
- \$LwpTableLineStyleOutline (1879)
- \$LwpTableLineStyleOutlineall (1880)
- \$LwpTableLineStyleOutlinecols (1884)
- \$LwpTableLineStyleOutlinerows (1883)
- \$LwpTableLineStyleRows (1881)

#### *LineStyle*

Allows you set one of the lines styles listed below to a table object. Data type is Variant.

- \$LtsBorderPatternDashDot (1056964659)
- \$LtsBorderPatternDashDotDot (1056964660)
- \$LtsBorderPatternDashed (1056964662)
- \$LtsBorderPatternDot (1056964663)
- \$LtsBorderPatternDouble (1056964666)
- \$LtsBorderPatternLongDash (1056964661)
- \$LtsBorderPatternNone (1056964657)
- \$LtsBorderPatternSolid (1056964658)
- \$LwpBorderPattern13space (36)
- \$LwpBorderPattern31space (37)
- \$LwpBorderPatternButttdown (35)
- \$LwpBorderPatternButtonup (34)
- \$LwpBorderPatternCircle (41)
- \$LwpBorderPatternDbIThick (51)
- \$LwpBorderPatternDbIWavy (56)
- \$LwpBorderPatternDeco1 (44)
- \$LwpBorderPatternDeco2 (45)
- \$LwpBorderPatternDeco3 (50)
- \$LwpBorderPatternDiagonal (38)
- \$LwpBorderPatternPin (47)
- \$LwpBorderPatternRain (46)
- \$LwpBorderPatternRope (43)

\$LwpBorderPatternRose (48)  
\$LwpBorderPatternStar (42)  
\$LwpBorderPatternSunf (49)  
\$LwpBorderPatternTaro (39)  
\$LwpBorderPatternThickDbwavy (58)  
\$LwpBorderPatternThickThin (53)  
\$LwpBorderPatternThickWavy (57)  
\$LwpBorderPatternThinThick (54)  
\$LwpBorderPatternThinThickThin (52)  
\$LwpBorderPatternWavy (55)  
\$LwpLtsBorderPatternDot (40)

#### *LineWidth*

Specifies the width for the selected line style. Data type is Twips.

#### *LineColor*

Specifies the color for the selected line style. Data type is Long.

#### *OutlineLineStyle*

Specifies the line style for the outline. Data type is Variant.

\$LtsBorderPatternDashDot (1056964659)  
\$LtsBorderPatternDashDotDot (1056964660)  
\$LtsBorderPatternDashed (1056964662)  
\$LtsBorderPatternDot (1056964663)  
\$LtsBorderPatternDouble (1056964666)  
\$LtsBorderPatternLongDash (1056964661)  
\$LtsBorderPatternNone (1056964657)  
\$LtsBorderPatternSolid (1056964658)  
\$LwpBorderPattern13space (36)  
\$LwpBorderPattern31space (37)  
\$LwpBorderPatternButttdown (35)  
\$LwpBorderPatternButtonup (34)  
\$LwpBorderPatternCircle (41)  
\$LwpBorderPatternDbIThick (51)  
\$LwpBorderPatternDbIWavy (56)  
\$LwpBorderPatternDeco1 (44)  
\$LwpBorderPatternDeco2 (45)  
\$LwpBorderPatternDeco3 (50)  
\$LwpBorderPatternDiagonal (38)  
\$LwpBorderPatternPin (47)  
\$LwpBorderPatternRain (46)  
\$LwpBorderPatternRope (43)  
\$LwpBorderPatternRose (48)  
\$LwpBorderPatternStar (42)  
\$LwpBorderPatternSunf (49)  
\$LwpBorderPatternTaro (39)  
\$LwpBorderPatternThickDbwavy (58)  
\$LwpBorderPatternThickThin (53)  
\$LwpBorderPatternThickWavy (57)

\$LwpBorderPatternThinThick (54)

\$LwpBorderPatternThinThickThin (52)

\$LwpBorderPatternWavy (55)

\$LwpLtsBorderPatternDot (40)

*OutlineLineWidth*

Specifies the width of the outline. Data type is Twips.

*OutlineLineColor*

Specifies the color of the outline. Data type is Long.

**Return value**

**Usage**

## Word Pro: SetLinkSource method

{button ,AL('H\_GRAPHIC\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETLINKSOURCE\_METHOD\_EXSCRIPT',1)} [See example](#)

### Syntax

[objectreference].SetLinkSource(LinkDisplayName, FileNameLength, ValidateSource, LinkCookie)

[objectreference].SetLinkSource(LinkDisplayName, FileNameLength, ValidateSource)

### Parameters

*LinkDisplayName*

Data type is String.

*FileNameLength*

Data type is Long.

*ValidateSource*

Data type is Integer.

*LinkCookie*

Data type is Long.

### Return value

Long

### Usage

## Word Pro: SetMinimumOrigin method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_SETMINIMUMORIGIN_METHOD_EXSCRIPT',1)} See example
```

Moves layout to make the length of the anchor tether as short as the anchor position allows.

### Syntax

```
[objectreference].SetMinimumOrigin()
```

### Parameters

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage



## Word Pro: SetNamedProperty method

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_CLICK_HERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SETNAMEDPROPERTY_METHOD_EXSCRIPT',1)} See example
```

Creates and assigns a value to a named property.

### Syntax

```
[objectreference].SetNamedProperty(PropertyName, NewValue)
```

### Parameters

#### *PropertyName*

A String expression representing the name of the property to which you are assigning a value.

#### *NewValue*

A String expression representing the new value you want to assign to the property.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

A named property is a user-defined property assigned to an object. Unlike variables, named properties are persistent. They continue to exist when a script stops executing, and when a document is closed and reopened.

If you want to modify the value of an existing named property, pass its name to the method in the *PropertyName* parameter. The named property will then be assigned the string value that you specify in the *NewValue* parameter.

**Word Pro: SetOverrideGraphic method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETOVERRIDEGRAPHIC\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the graphic on a button. This method is called when responding to the StatusBarButtonOverrideGraphic event or the StatusBarButtonOverrideTextAndGraphic event.

**Syntax**

[objectreference].SetOverrideGraphic(hGraphic)

**Parameters**

*hGraphic*

Data type is Long and represents the handle to the bitmap that you want to appear on the button. The bitmap is deleted for you, so do not use the bitmap again after you make this call.

**Return value**

Integer

**Usage**

This method must be used within the StatusBarButtonOverrideGraphic event. If you want to force the graphic to change, you can use the InvalidateButton method to force the StatusBarButtonOverrideGraphic event.

### **Word Pro: SetOverrideText method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETOVERRIDE TEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the text on a button. This method is called when responding to the StatusBarButtonOverrideText event or the StatusBarButtonOverrideTextAndGraphic event.

### **Syntax**

[objectreference].SetOverrideText(Text)

### **Parameters**

*Text*

Data type is String and represents the text you want to appear on the button. The text will be deleted for you, so do not use it again after you make this call.

### **Return value**

Integer

### **Usage**

This method must be used within the StatusBarButtonOverrideText event. If you want to force the text to change, you can use the InvalidateButton method to force the StatusBarButtonOverrideText event.

**Note** If the text on the status bar button is never going to change, you can use the LwpButtonNoTextFromHost (&H800) parameter when the button is created.

## Word Pro: SetPattern method

{button ,AL('H\_TABLEFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETPATTERN\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to simultaneously set the following: the specific part of the table you want to fill, the background fill style, the background color, and the pattern color.

### Syntax

[objectreference].SetPattern(TableFillStyle, BackgroundFill, BackgroundColor, PatternColor)

### Parameters

#### *TableFillStyle*

Allows you to set or not to set a table fill style to specific parts of a table object. The value of this Variant parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpTableFillStyleAll (1870)	Indicates that you want to fill the entire table with a specific fill style.
\$LwpTableFillStyleEveryothercol (1872)	Indicates that you want to fill every other column with a specific fill style.
\$LwpTableFillStyleEveryotherrow (1871)	Indicates that you want to fill every other row with a specific fill style.
\$LwpTableFillStyleMixed (1873)	Indicates that you want to use more than one fill style for the table object.
\$LwpTableFillStyleNone (1874)	Indicates that you do not want to set a fill style for the table object.

#### *BackgroundFill*

Allows you to set one of the values below as the style for the background of a fill in a table object. Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LtsFillBarLeftDiag (1056964681)  
\$LtsFillBarRightDiag (1056964688)  
\$LtsFillBasket (1056964710)  
\$LtsFillBigCheck (1056964717)  
\$LtsFillBottomTopGrad (1056964729)  
\$LtsFillBrick (1056964712)  
\$LtsFillBubbles (1056964724)  
\$LtsFillChevron (1056964708)  
\$LtsFillCicles (1056964719)  
\$LtsFillClumpedNarrowDiagHatch (1056964696)  
\$LtsFillClumpedZs (1056964723)  
\$LtsFillDarkNarrowDiagHatch (1056964693)  
\$LtsFillDiagBasket (1056964711)  
\$LtsFillDiagBrick (1056964713)  
\$LtsFillDiagHatch (1056964695)  
\$LtsFillDiamonds (1056964725)  
\$LtsFillDottedDarkHatch (1056964853)  
\$LtsFillDottedZigzag (1056964726)  
\$LtsFillDoubleLeftDiag (1056964684)  
\$LtsFillDoubleRightDiag (1056964690)  
\$LtsFillGray1 (1056964669)  
\$LtsFillGray10 (1056964678)

\$LtsFillGray2 (1056964670)  
\$LtsFillGray3 (1056964671)  
\$LtsFillGray4 (1056964672)  
\$LtsFillGray5 (1056964673)  
\$LtsFillGray6 (1056964674)  
\$LtsFillGray7 (1056964675)  
\$LtsFillGray8 (1056964676)  
\$LtsFillGray9 (1056964677)  
\$LtsFillHoriz (1056964699)  
\$LtsFillHorizBar (1056964698)  
\$LtsFillHorizCheckerboard (1056964716)  
\$LtsFillIrregularDiagScales (1056964721)  
\$LtsFillLeftDiag (1056964682)  
\$LtsFillLeftNarrowDiagHatch (1056964694)  
\$LtsFillLeftRightGrad (1056964728)  
\$LtsFillNarrowDoubleLeftDiag (1056964685)  
\$LtsFillNarrowDoubleRightDiag (1056964854)  
\$LtsFillNarrowHoriz (1056964697)  
\$LtsFillNarrowVert (1056964701)  
\$LtsFillNeToSwDiagStripGrad (1056964738)  
\$LtsFillNeToSwGrad (1056964730)  
\$LtsFillNone (1056964667)  
\$LtsFillNwToSeDiagStripGrad (1056964739)  
\$LtsFillNwToSeGrad (1056964731)  
\$LtsFillRandomBar (1056964680)  
\$LtsFillRandomSquare (1056964679)  
\$LtsFillRegularCheck (1056964718)  
\$LtsFillRegularHatch (1056964706)  
\$LtsFillRightDiag (1056964689)  
\$LtsFillRtLeftGrad (1056964744)  
\$LtsFillRunningDash (1056964714)  
\$LtsFillScalesDown (1056964722)  
\$LtsFillScalesUp (1056964720)  
\$LtsFillSolid (1056964668)  
\$LtsFillSteel (1056964709)  
\$LtsFillTinyHatch (1056964705)  
\$LtsFillTopBottomGrad (1056964745)  
\$LtsFillTripleLeftDiag (1056964686)  
\$LtsFillTripleRightDiag (1056964691)  
\$LtsFillVert (1056964703)  
\$LtsFillVertBar (1056964702)  
\$LtsFillVertCheckerboard (1056964715)  
\$LtsFillWideHatch (1056964707)  
\$LtsFillWideHoriz (1056964700)  
\$LtsFillWideLeftDiag (1056964687)  
\$LtsFillWideRightDiag (1056964692)  
\$LtsFillWideVert (1056964704)

\$LwpFillIndian3 (273)  
\$LwpFillPattern (2000)  
\$LwpFillPeachpie (274)

### *BackgroundColor*

Data type is Long which specifies the background color of the selected table cells.

Colors are usually represented by a combination of three separate components. The three components include a red value which can range from 0-255, a green value which can range from 0-255, and a blue value which can range from 0-255. You can combine different amounts of these three component colors to produce any other color. For example, in order to produce yellow, you would set a color object's red value to 255, green value to 255, and blue value to 0. The combination of all three of the component colors would display as yellow.

Colors can also be represented by a single numeric value. This allows you to specify any available color in only one method parameter, as opposed to three separate parameters. In order to calculate the value that represents a specific color, use this formula:

$$(\text{RedValue} * 65536\&) + (\text{GreenValue} * 256\&) + \text{BlueValue}$$

The '&' suffix appended to the constant values above ensures that the results of the expressions are always Long values. If you do not append the '&', the result of the expression may cause an overflow error.

The table below shows some examples of this formula, the result of the formula and the color represented by the value.

<b>Color</b>	<b>Formula</b>	<b>Result</b>
White	$(255 * 65536\&) + (255 * 256\&) + 255$	16777215
25% Gray	$(192 * 65536\&) + (192 * 256\&) + 192$	12632256
Red	$(255 * 65536\&) + (0 * 256\&) + 0$	16711680
Yellow	$(255 * 65536\&) + (255 * 256\&) + 0$	16776960
Neon Green	$(0 * 65536\&) + (255 * 256\&) + 0$	65280
Turquoise	$(0 * 65536\&) + (255 * 256\&) + 255$	65535
Blue	$(0 * 65536\&) + (0 * 256\&) + 255$	255
Hot Pink	$(255 * 65536\&) + (0 * 256\&) + 255$	16711935
Black	$(0 * 65536\&) + (0 * 256\&) + 0$	0
50% Gray	$(128 * 65536\&) + (128 * 256\&) + 128$	8421504
Scarlet	$(128 * 65536\&) + (0 * 256\&) + 0$	8388608
Olive	$(128 * 65536\&) + (128 * 256\&) + 0$	8421376
Dark Green	$(0 * 65536\&) + (128 * 256\&) + 0$	32768
Aztec Blue	$(0 * 65536\&) + (128 * 256\&) + 128$	32896
Dark Grape	$(0 * 65536\&) + (0 * 256\&) + 97$	97
Plum Red	$(128 * 65536\&) + (0 * 256\&) + 128$	8388736

### *PatternColor*

Specifies the pattern color of the selected table cells. This parameter is a Long value which represents the addition of a red, a green, and a blue color component value. For information on how to calculate the value for this parameter, refer to the BackgroundColor parameter definition.

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Use this method to specify a fill type, a pattern, a background color, and a pattern color for a selected range of table cells.

Use this property to simultaneously specify whether you want to fill the entire table or a specific row, cell, or column in a table object, the background fill style, the background color, and the pattern color. When you specify the background fill style, the background color and the pattern color, Word Pro applies your settings to the part of the table you selected to fill.



## Word Pro: SetPopupAlignment method

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETPOPUPALIGNMENT\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is called when responding to the StatusBarButtonFillPopupList event. This method lets you align the contents in the popup list in a status bar button.

### Syntax

[objectreference].SetPopupAlignment(AlignType)

### Parameters

#### *AlignType*

Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default constant.

\$LwpButtonAlignAlignCenter (89) A value that specifies center alignment of the button's contents.

\$LwpButtonAlignAlignLeft (87) A value that specifies left alignment of the button's contents.

\$LwpButtonAlignAlignRight (88) A value that specifies right alignment of the button's contents.

### Return value

Integer. Always returns True.

### Usage

This method lets you align the contents in the button's popup list.



**Word Pro: SetPopupIndex method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETPOPUPINDEX\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is called when responding to the StatusBarButtonFillPopupList event. This method lets you select which item in the status bar button's popup list will be highlighted by definition.

**Syntax**

[objectreference].SetPopupIndex(Index)

**Parameters**

*Index*

Data type is Integer.

**Return value**

True if item was selected; False if no item was selected.

**Usage**

This method lets you indicate which item in the popup list should be highlighted and selected by default. The index is the number corresponding to the entry you want to select in the list (zero-based). For example, 0 is the top item, 1 is the next item, 2 is the next, and so on.

If you specify an index greater than the number of items, this method returns False and nothing is selected.

## Word Pro: SetPopupWidthType method

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETPOPUPWIDTHTYPE\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is called when responding to the StatusBarButtonFillPopupList event.

### Syntax

[objectreference].SetPopupWidthType(WidthType)

### Parameters

#### *WidthType*

Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default constant.

\$LwpButtonSizeToButton (90) A value that specifies that the width of the popup list will be the width of the button.

\$LwpButtonSizeToSpecified (92) A value that specifies a user-defined width for the popup list, using the SetPopupWidth method to specify the width.

\$LwpButtonSizeToText (91) A value that specifies that the width of the popup list will conform to the size of the widest text in the list. For example, the value will be calculated to the size of the longest text string, or, if the text is shorter than the button's width, the value will be calculated to the width of the button.

### Return value

Integer

### Usage

Allows you to specify how the width of the popup list will be calculated.

**Word Pro: SetPopupWidth method**

{button ,AL('H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETPOPUPWIDTH\_METHOD\_EXSCRIPT',1)} [See example](#)

This method is called when responding to the StatusBarButtonFillPopupList event. This method lets you set the width of the popup list in the status bar button.

**Syntax**

[objectreference].SetPopupWidth(Width)

**Parameters**

*Width*

Data type is Integer. Set to width in Twips. There are 1440 Twips per inch.

**Return value**

Integer

**Usage**

You can use this method in conjunction with the StatusBarButtonFillPopupList event in the StatusBar class to set the width of the popup list. The width type parameter of the popup list must be \$LwpButtonSizeToSpecified to have any effect.

## **Word Pro: SetPowerFieldValue method**

{button ,AL('H\_MACRO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETPOWERFIELDVALUE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SetPowerFieldValue(PowerFieldName, PowerFieldValue) Bool

### **Parameters**

*PowerFieldName*

Data type is String.

*PowerFieldValue*

Data type is String.

### **Return value**

### **Usage**

## Word Pro: SetRGB method

{button ,AL('H\_COLOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETRGB\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the RGB (red, green, and blue) values of a specific object.

### Syntax

[objectreference].SetRGB(r, b, g)

### Parameters

*r*

The red component of a color. Data type is Integer.

*b*

The blue component of a color. Data type is Integer.

*g*

The green component of a color. Data type is Integer.

### Return value

Integer

### Usage

You can set the RGB value to pass the red, green, and blue color triplet to an object. You can also set the RGB value independently because each one of the RGB values is a property of Color.

## **Word Pro: SetStorage method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETSTORAGE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SetStorage(plStorage, FileType)

### **Parameters**

*plStorage*

Data type is Long.

*FileType*

Data type is String.

### **Return value**

### **Usage**

## Word Pro: SetStyle method

{button ,AL(^H\_CELLCONTAINER\_CLASS;H\_CLICKHERE\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SPERTABLECONTAINER\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS';0)} [See list of classes](#)

{button ,AL(^H\_SETSTYLE\_METHOD\_EXSCRIPT',1)} [See example](#)

Redefines the specified style to match the object from which the method is called. If the specified style does not exist, Word Pro creates it. The type of style defined or created by Word Pro depends on the type of object from which you call this method.

### Syntax

[Objectreference].SetStyle(StyleType, Style[, Exceptions])

### Parameters

#### StyleType

Indicates the type of object for which you want to set the style. Data type is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). There is no default value for this parameter, but you can use "\$LwpstyleTypeDefault" to have Word Pro assign the default style type for the type of object from which you call this method.

<u>Value</u>	<u>Effect</u>
\$LwpstyleTypeCell (1834)	Assigns a cell style.
\$LwpstyleTypeCharacter (1830)	Assigns a character style.
\$LwpstyleTypeDefault (1828)	Assigns the default style for the type of object from which you call this method.
\$LwpstyleTypeFrame (1832)	Assigns a frame style.
\$LwpstyleTypePage (1831)	Assigns a page style.
\$LwpstyleTypeParagraph (1829)	Assigns a paragraph style.
\$LwpstyleTypeTable (1833)	Assigns a table style.

#### Style

A String expression that identifies the style you want to assign. If the named style does not exist, Word Pro creates a new style with that name.

#### Exceptions

Allows you to specify which elements of a style will not get set by this method. There are two main types of style exceptions: Layout and Text. Furthermore, the default exception depends on the object from which you call this method.

<u>Object Class</u>	<u>Exception Type</u>	<u>Default Value</u>
PageContainer SuperPageContainer SubPageContainer	Layout Style Exceptions	LayoutstylePlacement
FrameContainer NoteContainer CellContainer RubyContainer SuperTableContainer DropCapContainer	Layout Style Exceptions	LayoutstyleSizeAndPlacement
ClickHere Text TextMarker	Text Style Exceptions	Textstyle None
WpApplication	N/A	0

### Layout Style Exceptions

LwpLayStyOverBackground (&H10)  
LwpLayStyOverBorders (&H8)  
LwpLayStyOverChildren (&H10000)  
LwpLayStyOverColumns (&H400)  
LwpLayStyOverContents (&H20000)  
LwpLayStyOverJoins (&H20)  
LwpLayStyOverLeaders (&H2000)  
LwpLayStyOverMargins (&H4)  
LwpLayStyOverMisc (&H8000)  
LwpLayStyOverNumerics (&H200)  
LwpLayStyOverOrientation (&H4000)  
LwpLayStyOverPlacement (&H2)  
LwpLayStyOverRotation (&H1000)  
LwpLayStyOverScaling (&H800)  
LwpLayStyOverScript (&H100)  
LwpLayStyOverShadow (&H40)  
LwpLayStyOverSize (&H1)  
LwpLayStyOverSizeAndPlacement (&H3)  
LwpLayStyOverTabs (&H80)

#### **Text Style Exceptions**

LwpTextStyleOverridesNone (&H0)  
LwpTextStyleBoverridesUllet (&H200)  
LwpTextStyleloverridesNdent (&H20)  
LwpTextStyleOverridesAlignment (&H10)  
LwpTextStyleOverridesAmikake (&H4000)  
LwpTextStyleOverridesAttributes (&H4)  
LwpTextStyleOverridesBorders (&H80)  
LwpTextStyleOverridesBreaks (&H100)  
LwpTextStyleOverridesBullet (&H200)  
LwpTextStyleOverridesCharborder (&H2000)  
LwpTextStyleOverridesFace (&H1)  
LwpTextStyleOverridesFont (&H8)  
LwpTextStyleOverridesIndent (&H20)  
LwpTextStyleOverridesKinsoku (&H1000)  
LwpTextStyleOverridesNumbering (&H400)  
LwpTextStyleOverridesSize (&H2)  
LwpTextStyleOverridesSpacing (&H40)  
LwpTextStyleOverridesTabs (&H800)  
LwpTextStyleSoverridesPacing (&H40)

The WPApplication object does not allow for the use of these exception values. Default when called from the WPApplication object is 0. Data type is Long.

#### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

#### **Usage**



## Word Pro: SetTableFill method

{button ,AL('H\_TABLEFILL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETTABLEFILL\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to simultaneously set the following: the specific part of the table you want to fill, the background fill style, the background color, and the pattern color.

### Syntax

[objectreference].SetTableFill(TableFillStyle, BackgroundFill, BackgroundColor, PatternColor)

### Parameters

#### *TableFillStyle*

Allows you to set or not to set a table fill style to specific parts of a table object. The value of this Variant parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpTableFillStyleAll (1870)	Indicates that you want to fill the entire table with a specific fill style.
\$LwpTableFillStyleEveryothercol (1872)	Indicates that you want to fill every other column with a specific fill style.
\$LwpTableFillStyleEveryotherrow (1871)	Indicates that you want to fill every other row with a specific fill style.
\$LwpTableFillStyleMixed (1873)	Indicates that you want to use more than one fill style for the table object.
\$LwpTableFillStyleNone (1874)	Indicates that you do not want to set a fill style for the table object.

#### *BackgroundFill*

Allows you to set one of the values below as the style for the background of a fill in a table object. The value of this Variant parameter must be one of the strings below or its code equivalent.

\$LtsFillBarLeftDiag (1056964681)

\$LtsFillBarRightDiag (1056964688)

\$LtsFillBasket (1056964710)

\$LtsFillBigCheck (1056964717)

\$LtsFillBottomTopGrad (1056964729)

\$LtsFillBrick (1056964712)

\$LtsFillBubbles (1056964724)

\$LtsFillChevron (1056964708)

\$LtsFillCicles (1056964719)

\$LtsFillClumpedNarrowDiagHatch (1056964696)

\$LtsFillClumpedZs (1056964723)

\$LtsFillDarkNarrowDiagHatch (1056964693)

\$LtsFillDiagBasket (1056964711)

\$LtsFillDiagBrick (1056964713)

\$LtsFillDiagHatch (1056964695)

\$LtsFillDiamonds (1056964725)

\$LtsFillDottedDarkHatch (1056964853)

\$LtsFillDottedZigzag (1056964726)

\$LtsFillDoubleLeftDiag (1056964684)

\$LtsFillDoubleRightDiag (1056964690)

\$LtsFillGray1 (1056964669)

\$LtsFillGray10 (1056964678)

\$LtsFillGray2 (1056964670)  
\$LtsFillGray3 (1056964671)  
\$LtsFillGray4 (1056964672)  
\$LtsFillGray5 (1056964673)  
\$LtsFillGray6 (1056964674)  
\$LtsFillGray7 (1056964675)  
\$LtsFillGray8 (1056964676)  
\$LtsFillGray9 (1056964677)  
\$LtsFillHoriz (1056964699)  
\$LtsFillHorizBar (1056964698)  
\$LtsFillHorizCheckerboard (1056964716)  
\$LtsFillIrregularDiagScales (1056964721)  
\$LtsFillLeftDiag (1056964682)  
\$LtsFillLeftNarrowDiagHatch (1056964694)  
\$LtsFillLeftRightGrad (1056964728)  
\$LtsFillNarrowDoubleLeftDiag (1056964685)  
\$LtsFillNarrowDoubleRightDiag (1056964854)  
\$LtsFillNarrowHoriz (1056964697)  
\$LtsFillNarrowVert (1056964701)  
\$LtsFillNeToSwDiagStripGrad (1056964738)  
\$LtsFillNeToSwGrad (1056964730)  
\$LtsFillNone (1056964667)  
\$LtsFillNwToSeDiagStripGrad (1056964739)  
\$LtsFillNwToSeGrad (1056964731)  
\$LtsFillRandomBar (1056964680)  
\$LtsFillRandomSquare (1056964679)  
\$LtsFillRegularCheck (1056964718)  
\$LtsFillRegularHatch (1056964706)  
\$LtsFillRightDiag (1056964689)  
\$LtsFillRtLeftGrad (1056964744)  
\$LtsFillRunningDash (1056964714)  
\$LtsFillScalesDown (1056964722)  
\$LtsFillScalesUp (1056964720)  
\$LtsFillSolid (1056964668)  
\$LtsFillSteel (1056964709)  
\$LtsFillTinyHatch (1056964705)  
\$LtsFillTopBottomGrad (1056964745)  
\$LtsFillTripleLeftDiag (1056964686)  
\$LtsFillTripleRightDiag (1056964691)  
\$LtsFillVert (1056964703)  
\$LtsFillVertBar (1056964702)  
\$LtsFillVertCheckerboard (1056964715)  
\$LtsFillWideHatch (1056964707)  
\$LtsFillWideHoriz (1056964700)  
\$LtsFillWideLeftDiag (1056964687)  
\$LtsFillWideRightDiag (1056964692)  
\$LtsFillWideVert (1056964704)

\$LwpFillIndian3 (273)  
\$LwpFillPattern (2000)  
\$LwpFillPeachpie (274)

#### *BackgroundColor*

Allows you to set the background color of a fill in a table object. The value of this Variant parameter must be one of the strings below or its code equivalent.

\$LwpColorOverrideBlack (2017)  
\$LwpColorOverrideBlue (2024)  
\$LwpColorOverrideDarkGray (2021)  
\$LwpColorOverrideGray (2019)  
\$LwpColorOverrideGreen (2023)  
\$LwpColorOverrideInvalid (2026)  
\$LwpColorOverrideLightGray (2020)  
\$LwpColorOverrideRed (2022)  
\$LwpColorOverrideReserved (2025)  
\$LwpColorOverrideRgb (2016)  
\$LwpColorOverrideTransparent (2027)  
\$LwpColorOverrideWhite (2018)

#### *PatternColor*

Allows you to set the pattern color of a fill in a table object. The value of this Variant parameter must be one of the strings below or its code equivalent.

\$LwpColorOverrideBlack (2017)  
\$LwpColorOverrideBlue (2024)  
\$LwpColorOverrideDarkGray (2021)  
\$LwpColorOverrideGray (2019)  
\$LwpColorOverrideGreen (2023)  
\$LwpColorOverrideInvalid (2026)  
\$LwpColorOverrideLightGray (2020)  
\$LwpColorOverrideRed (2022)  
\$LwpColorOverrideReserved (2025)  
\$LwpColorOverrideRgb (2016)  
\$LwpColorOverrideTransparent (2027)  
\$LwpColorOverrideWhite (2018)

#### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

#### **Usage**

Use this property to simultaneously specify whether you want to fill the entire table or a specific row, cell, or column in a table object, the background fill style, the background color, and the pattern color. When you specify the background fill style, the background color and the pattern color, Word Pro applies your settings to the part of the table you selected to fill.

This method provides a quick and easy way to set multiple table fill properties. If you would like to set the background or pattern to a color that is not available in this method, you must access the appropriate table fill objects and set those properties individually.

## Word Pro: SetTOCProperties method

{button ,AL('H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETTOCPROPERTIES\_METHOD\_EXSCRIPT',1)} [See example](#)

### Syntax

[objectreference].SetTOCProperties(TOCScope, Index, NewValue)

### Parameters

#### *TOCScope*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpTOCScopeDestPageStyle (1853)

\$LwpTOCScopeDestStyle (1854)

\$LwpTOCScopeLevelNumber (1851)

\$LwpTOCScopeRtAlgnPgNum (1850)

\$LwpTOCScopeSearchStyle (1852)

\$LwpTOCScopeTypeOfLeader (1846)

\$LwpTOCScopeUseLeader (1847)

\$LwpTOCScopeUsePageNumber (1849)

\$LwpTOCScopeUseText (1848)

#### *Index*

Data type is Integer.

#### *NewValue*

Data type is Variant.

### Return value

### Usage

## **Word Pro: SetUpEnvelopeMerge method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SETUPENVELOPEMERGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the options for an envelope merge and opens the Merge bar.

### **Syntax**

[objectreference].SetUpEnvelopeMerge(envWidth, envHeight, envBin, barcode, returnAddress)

### **Parameters**

*envWidth*

A numeric expression of type Long which specifies the width of the envelopes you will be merging.

*envHeight*

A numeric expression of type Long which specifies the height of the envelopes you will be merging.

*envBin*

A String expression which specifies the name of the printer bin which contains the envelopes. Available values for this parameter are displayed in your printer's Properties dialog box in the "Paper source" option.

*barcode*

An integer expression which tells Word Pro whether or not you want to include a bar code on the envelope. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. This is an optional parameter. The default value is 0 (False).

*returnAddress*

An integer expression which tells Word Pro whether or not you want to include a return address on the envelope. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. This is an optional parameter. The default value is 0 (False).

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: Shade method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHADE\_METHOD\_EXSCRIPT',1)} [See example](#)

Selects text in the current context.

### Syntax

[objectreference].Shade(LocationType,Unit, N)

### Parameters

#### *LocationType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpLocationTypeDocument (573)

\$LwpLocationTypeLine (571)

\$LwpLocationTypeParagraph (572)

\$LwpLocationTypeSelection (568)

\$LwpLocationTypeSentence (570)

\$LwpLocationTypeStream (574)

\$LwpLocationTypeWord (569)

#### *Unit*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpNavigateDirectionDown (1515)

\$LwpNavigateDirectionLeft (1516)

\$LwpNavigateDirectionRight (1517)

\$LwpNavigateDirectionUp (1514)

#### *Numberof Units*

Data type is Integer.

### Return value

### Usage

## **Word Pro: ShowCaretAndSelection method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWCARETANDSELECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the current selection

### **Syntax**

[objectreference].ShowCaretAndSelection(ForceCaretToShow, ShowCaret, ShowSelection)

### **Parameters**

forces the caret to the screen if True default is False

True

True

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: ShowContainers method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASSES;H_TABLEONLYCONT_CLASS',0)} See list of classes
```

```
{button ,AL('H_SHOWCONTAINERS_METHOD_EXSCRIPT',1)} See example
```

Selects a specific container in a document.

### Syntax

```
[objectreference].ShowContainers()
```

### Parameters

### Return value

### Usage



## **Word Pro: ShowCursor method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWCURSOR\_METHOD\_EXSCRIPT',1)} [See example](#)

Scrolls the active document window so the insertion point is visible on the screen.

### **Syntax**

[objectreference].ShowCursor()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: ShowIconBars method**

{button ,AL('H\_ICONBARMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWICONBARS\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the set of SmartIcons in its default location on the screen.

**Syntax**

[objectreference].ShowIconBars()

**Parameters****Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: ShowScrollBars method**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWSCROLLBARS\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the scroll bars in the document window.

### **Syntax**

[objectreference].ShowScrollBars(ScrollVert,ScrollHorz)

### **Parameters**

*ScrollVert*

Data type is Boolean.

*ScrollHorz*

Data type is Boolean.

### **Return value**

### **Usage**

**Word Pro: ShowStatusBar method**

{button ,AL('H\_STATUSBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOWSTATUSBAR\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays the status bar.

**Syntax**

[objectreference].ShowStatusBar()

**Parameters****Return value**

Integer

**Usage**

**Word Pro: Show method**

{button ,AL('H\_DOCWINDOW\_CLASS;H\_ICONBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SHOW\_METHOD\_EXSCRIPT',1)} [See example](#)

Displays an object, such as an icon bar set or a document.

**Syntax**

[objectreference].Show()

**Parameters**

Data type is Integer. The legal values for this parameter are -1 or 0 but you may use the LotusScript constants of True (-1) and False (0).

**Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: SimulateButtonClick method**

{button ,AL(^H\_STATUSBARBUTTON\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SIMULATEBUTTONCLICK\_METHOD\_EXSCRIPT',1)} [See example](#)

Simulates a user click on the button. The StatusBarButtonClicked event will be emitted.

**Syntax**

[objectreference].SimulateButtonClick()

**Parameters****Return value**

Integer. Always returns True.

**Usage**

Use this method to simulate a user click on the button. This method forces a StatusBarButtonClicked event.

## Word Pro: Skip method

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SKIP\_METHOD\_EXSCRIPT',1)} [See example](#)

### Syntax

[objectreference].Skip(SkipType,SkipMeaning)

### Parameters

#### *SkipType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpSkipTypeFormatCheck (1767)

\$LwpSkipTypeRevision (1765)

\$LwpSkipTypeSpell (1766)

#### *SkipMeaning*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

\$LwpSkipMeanRevSkipBackwordWord (1763)

\$LwpSkipMeanRevSkipForwardWord (1762)

\$LwpSkipMeanSpellMarkWord (1764)

### Return value

### Usage

### **Word Pro: SmallCaps method**

{button ,AL('H\_FONT\_CLASS;H\_FONTMETRICS\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SMALLCAPS\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the Small Caps attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on and on if it is off. This is the same as choosing Text - Attributes - Other and then choosing "Small Caps" from the Attributes box.

### **Syntax**

[objectreference].SmallCaps()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**



**Word Pro: SmartSumColumn method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_SMARTSUMCOLUMN\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a SmartSum formula in the currently active table cell.

**Syntax**

[objectreference].SmartSumColumn()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

The SmartSum formula adds the values of cells above and in the same column as the SmartSum cell and displays the total value in the SmartSum cell. If the column contains a cell with a text value, the SmartSum formula only totals the cells between the SmartSum cell and the cell with the text.

## **Word Pro: SmartSumRow method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_SMARTSUMROW\_METHOD\_EXSCRIPT',1)} [See example](#)

Inserts a SmartSum formula in the currently active table cell.

### **Syntax**

[objectreference].SmartSumRow()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

The SmartSum formula adds the values of cells to the left of and in the same row as the SmartSum cell and displays the total value in the SmartSum cell. If the row contains a cell with a text value, the SmartSum formula only totals the cells between the SmartSum cell and the cell with the text.

```
'Example: RevisionAcceptLayoutChange method
' This example accepts all revision markup changes made to the current
' layout object.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Layout.RevisionAcceptLayoutChange True, "Mark Osborne"
```

```
'Example: RevisionCancelLayoutChange method
' This example cancels all revision markup changes made to the current
' layout object.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Layout.RevisionCancelLayoutChange True, ""
```

'Example: RevisionCancel method

'This example script has not yet been created.

'Example: RevisionDisplay property

'This example script has not yet been created.

'Example: RevisionMarkMode property  
'This example script has not yet been created.

'Example: RevisionMark property

'This example script has not yet been created.



'Example: RevisionType property

'This example script has not yet been created.

'Example: RevMarkCharacter property  
'This example script has not yet been created.

'Example: RevMarkPosition property  
'This example script has not yet been created.

'Example: RevMarkType property

'This example script has not yet been created.

'Example: RightAlign property

'This example script has not yet been created.

'Example: RightBorder property

'This example script has not yet been created.

'Example: RightExternalMargin property

'This example script has not yet been created.

'Example: RightIntArea property

'This example script has not yet been created.



'Example: RightMouseDown property

'This example script has not yet been created.

'Example: RightMousePropId property

'This example script has not yet been created.

'Example: RightMousePropText property  
'This example script has not yet been created.

'Example: RightPage property

'This example script has not yet been created.

'Example: Right property

'This example script has not yet been created.

'Example: RotationAngle property

'This example script has not yet been created.

'Example: RowLayouts property

'This example script has not yet been created.

'Example: Rows property

'This example script has not yet been created.



'Example: RubyLayouts property

'This example script has not yet been created.

'Example: RunAutoNewMacro method

'This example script has not yet been created.

'Example: RunAutoOpenMacro method

'This example script has not yet been created.

'Example: RunMacroOnDocEvents property

'This example script has not yet been created.

'Example: RunMacroOnLoad property

'This example script has not yet been created.

'Example: RunOnCloseDoc property

'This example script has not yet been created.

'Example: RunOnNewDoc property

'This example script has not yet been created.

'Example: RunOnOpenDoc property

'This example script has not yet been created.



```
'Example: RunScript method  
' This example runs the 'TestFunction' module in the script file 'EXAMPLE.LWP'  
' RUNTIME DEPENDENCIES: You must have a script file named 'EXAMPLE.LWP' which  
' contains a function named 'TestFunction'. The script file should be  
' located in the WordPro preferences Script directory.
```

```
Dim ScriptModule as String  
Dim ScriptFunction as String
```

```
ScriptModule = "EXAMPLE.LWP"  
ScriptFunction = "TestFunction"
```

```
.RunScript(ScriptModule, ScriptFunction)
```

'Example: Run method

'This example script has not yet been created.

'Example: SaveAs event

'This example script has not yet been created.

```
'Example: SaveAs method
' This example saves the current document in ASCII format.  The file is saved
' in Word Pro's document directory using the file name given returned by the
' input box.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Dim FileName as String
Dim FilePath as String
Dim FilterType as String
FilterType = "Text - ASCII (DOS)"
FilePath = .ApplicationWindow.UserInterfacePrefs.DocPath
FileName = InputBox("Enter a filename to save as:", "Example Script", "")
If FileName <> "" Then
    .SaveAs FileName, FilePath , FilterType, False, True, False
End If
```

'Example: SavedAs event

'This example script has not yet been created.

'Example: SaveData method

'This example script has not yet been created.

'Example: SaveDivision method

'This example script has not yet been created.

'Example: Saved event

'This example script has not yet been created.



'Example: Saved property

'This example script has not yet been created.

'Example: SaveEnvelopeMaster method  
'This example script has not yet been created.

'Example: SaveGlossary method

' This example stores a file name in the variable GlossFileName, hides the  
' open documents, opens the default Word Pro glossary file, creates and saves  
' a glossary file named "GLOSTST.GLS" in the User Setup glossary directory,  
' then closes the glossary files and resets the default values user interface  
' preferences.

' RUNTIME DEPENDENCIES: You must have create file rights in the specified  
' glossary directory for this script to work.

Dim GlossFileName As String

GlossFileName = "GLOSTST.GLS"

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = False

.GlossaryOpen GlossFileName, "Lotus Word Pro"

.CreateGlossary

**.SaveGlossary GlossFileName, "Lotus Word Pro", False**

.Close

.ApplicationWindow.UserInterfacePrefs.IsReplacement = False

.ApplicationWindow.UserInterfacePrefs.OpenDocsVisible = True

.ApplicationWindow.UserInterfacePrefs.OpenReadOnly = False

'Example: SaveMacro method

'This example script has not yet been created.

'Example: SaveSnapshot method

'This example script has not yet been created.

'Example: SaveSnapShot property

'This example script has not yet been created.

'Example: SaveToStorageComplete method

'This example script has not yet been created.

'Example: SaveToStorage method

'This example script has not yet been created.



'Example: SaveUserDefaults method

' This example changes the number of recent files to 5.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.UserInterfacePrefs.NumOfRecentFiles = 5

.ApplicationWindow.SaveUserDefaults

```
'Example: SaveVersion method
' This example saves the selected version as a file named TESTVER.LWP.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim FileName As String
Dim Version As Long
FileName = .ApplicationWindow.UserInterfacePrefs.DocPath & "\TESTVER.LWP"
Version = 2
.SaveVersion FileName, Version
```

'Example: Save event

'This example script has not yet been created.

'Example: Save method

' This example saves the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Save

'Example: ScaleHeight property

'This example script has not yet been created.

'Example: ScaleMode property

'This example script has not yet been created.

'Example: ScalePercentage property

'This example script has not yet been created.

'Example: ScaleWidth property

'This example script has not yet been created.



```
'Example: ScreenPositionX property
Dim CR As String*1
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager

IcnPallet = "Comment Tools"
CR = Chr(10)
Set IcnMgr = .ApplicationWindow.IconBarManager

With IcnMgr.IconBars(IcnPallet)
    MsgStr = "Height = " & .Height & CR
    MsgStr = MsgStr & "IconBarPositionState = " & .IconBarPositionState & CR
    MsgStr = MsgStr & "PositionType = " & .PositionType & CR
    MsgStr = MsgStr & "ScreenPositionX = " & .ScreenPositionX & CR
    MsgStr = MsgStr & "ScreenPositionY = " & .ScreenPositionY
    MessageBox MsgStr, 64, "Script Example - " & .Name
End With
```

```
'Example: ScreenPositionY property
Dim CR As String*1
Dim IcnPallet As String
Dim MsgStr As String
Dim IcnMgr As IconBarManager

IcnPallet = "Comment Tools"
CR = Chr(10)
Set IcnMgr = .ApplicationWindow.IconBarManager

With IcnMgr.IconBars(IcnPallet)
    MsgStr = "Height = " & .Height & CR
    MsgStr = MsgStr & "IconBarPositionState = " & .IconBarPositionState & CR
    MsgStr = MsgStr & "PositionType = " & .PositionType & CR
    MsgStr = MsgStr & "ScreenPositionX = " & .ScreenPositionX & CR
    MsgStr = MsgStr & "ScreenPositionY = " & .ScreenPositionY
    MessageBox MsgStr, 64, "Script Example - " & .Name
End With
```

'Example: ScriptName property

'This example script has not yet been created.

'Example: Script property

'This example script has not yet been created.

'Example: SearchAttributes property  
'This example script has not yet been created.

'Example: SearchLanguage property

'This example script has not yet been created.

'Example: SectionName property

'This example script has not yet been created.

'Example: Sections property

'This example script has not yet been created.



'Example: SectionTabs property

'This example script has not yet been created.

'Example: Section property

'This example script has not yet been created.

'Example: SelectAStyle method

' This example displays the popup style menu located on Word Pro's status bar.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.SelectAStyle

```
'Example: SelectCell method
' This example creates a table with 5 columns and 4 rows based on the
' Default Table' style. The current cell is then selected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.Text.InsertText "A table cell"
.SelectCell
```

'Example: SelectColumn method

' This example creates a table with 5 columns and 4 rows based on the

' Default Table style. The first column is selected.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4

**.SelectColumn**

'Example: SelectCustomIcon method

'This example script has not yet been created.

'Example: SelectDoc method

'This example script has not yet been created.

'Example: SelectedPages property

'This example script has not yet been created.



```
'Example: SelectEntireCellRange method
' This example creates a table with 5 columns and 5 rows, inserts some text
' and then selects the inserted text.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.Table.CellLayout(1,1).GotoLayout
.Type "Some text for the table cell"
.SelectEntireCellRange
```

```
'Example: SelectEntireColumn method
' This example creates a table with 5 columns and 4 rows based on the
' Default Table style and then selects the current column.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.Text.InsertText "A table cell"
.SelectEntireColumn
```

```
'Example: SelectEntirePColCellRange method
' This example creates a parallel column table, inserts some text and then
' selects the inserted text and cell.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, $LtsAlignmentHorizCenter
.Type "Some text for the cell"
.SelectEntirePColCellRange
```

'Example: SelectEntirePColColumn method

' This example creates a parallel column table with 3 columns and then selects  
' the first column.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.SelectEntirePColColumn

'Example: SelectEntirePColRow method

' This example creates a parallel column table with 3 columns and then selects  
' the first row.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.SelectEntirePColRow

'Example: SelectEntirePCol method

' This example creates a parallel column table with 3 columns and then selects  
' the first column.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateParallelColumns 3, \$LtsAlignmentHorizCenter

.SelectEntirePCol

```
'Example: SelectEntireRow method
' This example creates a table with 5 columns and 4 rows based on the
' Default Table' style and then selects the current row.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.Text.InsertText "A table cell"
.SelectEntireRow
```

```
'Example: SelectEntireTable method
' This example creates a table with 5 columns and 4 rows based on the
' Default Table' style and then selects the entire table.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.Text.InsertText "A table cell"
.SelectEntireTable
```



'Example: SelectionBorderColor1 property

'This example script has not yet been created.

'Example: SelectionBorderColor2 property  
'This example script has not yet been created.

'Example: SelectionBorderColor3 property

'This example script has not yet been created.

'Example: SelectionHidden property  
'This example script has not yet been created.

'Example: SelectionType property

'This example script has not yet been created.

```
'Example: SelectMarker method
' This example creates a temporary text marker for the inserted text.
' The marker is then selected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim NewMarker As TextMarker
Dim MarkerText As String
.Text.InsertText "Hello "
.Select $LwpSelectObjectTypeParagraph
MarkerName = .Mark($LWPMarkerTypeDefault)
Set NewMarker = .Division.Foundry.TextMarkers.Item(MarkerName)
.Deselect
.Text.SelectMarker MarkerName
```

```
'Example: SelectParagraph method
' This example inserts text into the current document and selects the
' current paragraph.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "This is some sample text."
.SelectParagraph
```

'Example: SelectPColCell method

' This example creates a parallel column table with 3 columns and then selects  
' the first cell.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
.CreateParallelColumns 3, $LtsAlignmentHorizCenter  
.SelectPColCell
```



'Example: SelectRow method

' This example creates a table with 5 columns and 4 rows based on the

' Default Table style. The first row is selected.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4

**.SelectRow**

'Example: SelectRuler method

'This example script has not yet been created.

'Example: SelectSection method

' This example selects the contents of the current section of the document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.SelectSection

```
'Example: SelectSentence method
' This example inserts some text into the current document and positions the
' cursor at the line's begininning. The entire sentence is then selected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "Some sample text."
.Text.Backward $LwpNavigateObjectTypeSentence, 1

.SelectSentence
```

'Example: SelectStandardIcon method

'This example script has not yet been created.

```
'Example: SelectTableItem method
' This example creates a table with 5 rows and 5 columns and then selects the
' entire table.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5,5
.Table.SelectTableItem($LwpTableSelectionTable)
```

```
'Example: SelectTable method
' This example creates a table with 5 columns and 4 rows based on the
' Default Table' style and then selects the table.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 5, 4
.Text.InsertText "A table cell"
.SelectTable
```

'Example: SelectTab property

'This example script has not yet been created.



'Example: SelectType property

'This example script has not yet been created.

```
'Example: SelectWord method
' This example inserts some text into the current document and positions the
' cursor at the line's begininning. The first word is then selected.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
.Text.InsertText "Some sample text."
.Text.Backward $LwpNavigateObjectTypeSentence, 1

.SelectWord
```

'Example: Select method

' This example inserts some text into the current document which is then selected.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "This is some text."

.Select \$LwpSelectObjectTypeParagraph

**Word Pro: SortLevel2 property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTLEVEL2\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies a secondary sort order in a multi-level sort based on the primary sort order established by the SortLevel1 property.

**Data Type**

[SortKey](#)

**Syntax**

[objectreference].SortLevel2 = sortlevel2value

sortlevel2value = [objectreference].SortLevel2

**Legal values**

Always contains an instance of the SortKey class.

**Usage**

Equivalent to choosing Text - Sort and selecting any one of the options in the "Field/col.," "Type," "Order," or "Word" box in the second sort level.

**Word Pro: SortLevel3 property**

{button ,AL('H\_SORTOPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SORTLEVEL3\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Specifies an additional sort order in a multi-level sort based on the secondary sort order established by the SortLevel2 property.

**Data Type**

[SortKey](#)

**Syntax**

[objectreference].SortLevel3 = sortlevel3value

sortlevel3value = [objectreference].SortLevel3

**Legal values**

Always contains an instance of the SortKey class.

**Usage**

Equivalent to choosing Text - Sort and selecting any one of the options in the "Field/col.," "Type," "Order," or "Word" box in the third sort level.

**Word Pro: SortOptions property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SORTOPTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[SortOptions](#)

**Syntax**

sortoptionsvalue = [objectreference].SortOptions

**Legal values**

Always contains an instance of the SortOptions class.

**Usage**

## Word Pro: Spacing property

{button ,AL(^H\_CLICKHERE\_CLASS;H\_FORMULA\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_TEXT\_CLASS;H\_T  
EXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPACING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### Data Type

[Spacing](#)

### Syntax

spacingvalue = [objectreference].Spacing

### Legal values

Always contains an instance of the Spacing class.

### Usage

## **Word Pro: SpellColor property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPELLCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores the color that highlights spelling and grammar errors.

### **Data Type**

[Color](#)

### **Syntax**

spellcolorvalue = [objectreference].SpellColor

### **Legal values**

Always contains an instance of the Color class.

### **Usage**

Equivalent to the "Color for unrecognized words" value in the Spell Check Options dialog box.



**Word Pro: SpellFocusedColor property**

{button ,AL(^H\_APPVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPELLFOCUSEDCOLOR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The color used to highlight spelling and grammar errors.

**Data Type**

[Color](#)

**Syntax**

spellfocusedcolorvalue = [objectreference].SpellFocusesColor

**Legal values**

Always contains an instance of the Color class.

**Usage**

This color is always the inverse of the color that is stored in the SpellColor property.

**Word Pro: StatusBarButtons property**

{button ,AL('H\_STATUSBAR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBARBUTTONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) This property provides a list of all buttons in the status bar, by name.

**Data Type**

[StatusBarButtonCollection](#)

**Syntax**

statusbarbuttonvalue = [objectreference].StatusBarButtons

**Legal values**

Always contains an instance of the StatusBarButtonCollection class.

**Usage**

If you add a new status button to the status bar in LotusScript, this property returns an assigned number, such as CUSTOM1.

**Word Pro: StatusBar property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STATUSBAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the StatusBar class.

**Data Type**

[StatusBar](#)

**Syntax**

statusbarvalue = [objectreference].StatusBar

**Legal values**

Always contains an instance of the StatusBar class.

**Usage**

Use this property to access the StatusBar object when a document is not open.

**Word Pro: StylePaths property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STYLEPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores multiple paths (drive and directories) for SmartMaster files.

**Data Type**

[StringCollection](#)

**Syntax**

stylepathsvalue = [objectreference].StylePaths

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

Equivalent to the "SmartMaster" value on the Locations panel of the Word Pro Preferences dialog box. "SmartMaster" can contain multiple paths. You can use this property to read these multiple paths, including the default or first path that is also stored in the StylePath property on UserInterfacePrefs.

## Word Pro: Style property

```
{button ,AL('H_BORDER_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUP  
LAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLA  
SS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAM  
ELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAY  
OUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLA  
YOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADIN  
GLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_STYLE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates the specific layout object from which the current layout object inherited its style settings.

## Data Type

[Layout](#)

## Syntax

stylevalue = [objectreference].Style

## Legal values

Always contains an instance of the Layout class.

## Usage

You can use this property to access the default property values that were assigned to a layout object when it was created. For example, you could create a frame based on the Default Frame style and then change the number of columns within that frame. To access the current number of columns for the frame, you could use the following statement:

```
[framelayoutobject].numcols
```

To check the number of columns a frame was assigned when it was initially created, you could use the following statement:

```
[framelayoutobject].style.numcols
```

## **Word Pro: SuperTableContainer property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SUPERTABLECONTAINER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the SuperTableContainer class. This is a current context property which only contains an object when the focus of Word Pro includes a table. If there is no table in the focus, this property is empty.

### **Data Type**

[SuperTableContainer](#)

### **Syntax**

supertablecontainervalue = [objectreference].SuperTableContainer

### **Legal values**

An instance of the SuperTableContainer class.

### **Usage**

When the focus includes a table, this property contains the SuperTableContainer object which groups together the objects that comprise the SuperTable object in the focus. You can use this property to access the Layout or other objects related to that super table.

## **Word Pro: SuperTableLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUPERTABLELAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the SuperTableLayoutCollection class. This object provides access to SuperTableLayout objects.

### **Data Type**

[SuperTableLayoutCollection](#)

### **Syntax**

supertablelayoutsvalue = [objectreference].SuperTableLayouts

### **Legal values**

Always contains an instance of the SuperTableLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the SuperTableLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the SuperTableLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the SuperTableLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the SuperTableLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: SuperTables property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUPERTABLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the SuperTableCollection class. This object provides access to SuperTable objects.

### **Data Type**

[SuperTableCollection](#)

### **Syntax**

supertablesvalue = [objectreference].SuperTables

### **Legal values**

Always contains an instance of the SuperTableCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the SuperTable objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the SuperTable objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the SuperTable objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the SuperTable objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.



## **Word Pro: TableContainer property**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLECONTAINER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the ParallelColsContainer class or the TableOnlyCont class. This is a current context property which only contains an object when the focus of Word Pro includes a page with parallel columns or a table. If neither parallel columns nor a table are in the focus, this property is empty.

### **Data Type**

[TableContainer](#)

### **Syntax**

tablecontainervalue = [objectreference].TableContainer

### **Legal values**

An instance of the ParallelColsContainer class or the TableOnlyCont class.

### **Usage**

When the focus includes a page with parallel columns, this property contains the ParallelColsContainer object which groups together the objects that comprise the parallel columns in the focus. You can use this property to access the Layout object related to those parallel columns.

When the focus includes a table, this property contains the TableOnlyCont object which groups together the objects that comprise the table in the focus. You can use this property to access the TableLayout or other objects related to that table.

**Word Pro: TableExports property**

{button ,AL(^H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TABLEEXPORTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

tableexportsvalue = [objectreference].TableExports

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

**Word Pro: TableFill property**

{button ,AL(^H\_BASERABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TABLEFILL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the background that displays in a cell within a table object.

**Data Type**

[TableFill](#)

**Syntax**

tablefillvalue = [objectreference].TableFill

**Legal values**

Always contains an instance of the TableFill class.

**Usage**

## **Word Pro: TableHeadingLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLEHEADINGLAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TableHeadingLayoutCollection class. This object provides access to TableHeadingLayout objects.

### **Data Type**

[TableHeadingLayoutCollection](#)

### **Syntax**

tableheadinglayoutsvalue = [objectreference].TableHeadingLayouts

### **Legal values**

Always contains an instance of the TableHeadingLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the TableHeadingLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the TableHeadingLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the TableHeadingLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the TableHeadingLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## **Word Pro: TableHeadings property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLEHEADINGS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TableHeadingCollection class. This object provides access to TableHeading objects.

### **Data Type**

[TableHeadingCollection](#)

### **Syntax**

tableheadingsvalue = [objectreference].TableHeadings

### **Legal values**

Always contains an instance of the TableHeadingCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the TableHeading objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the TableHeading objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the TableHeading objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the TableHeading objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: TableImports property**

{button ,AL('H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLEIMPORTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

tableimportsvalue = [objectreference].TableImports

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

## **Word Pro: TableLayouts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLELAYOUTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TableLayoutCollection class. This object provides access to TableLayout objects.

### **Data Type**

[TableLayoutCollection](#)

### **Syntax**

tablelayoutvalue = [objectreference].TableLayouts

### **Legal values**

Always contains an instance of the TableLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the TableLayout objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the TableLayout objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the TableLayout objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the TableLayout objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

### **Word Pro: TableLine property**

{button ,AL(^H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMNS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TABLELINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains an instance of the TableLine class, which allows you to modify border line styles around tables.

### **Data Type**

[TableLine](#)

### **Syntax**

tablelinevalue = [objectreference].TableLine

### **Legal values**

Always contains an instance of the TableLine class.

### **Usage**



## **Word Pro: TableMarkers property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLEMARKERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TableMarkerCollection class. This object provides access to TableMarker objects.

### **Data Type**

[TableMarkerCollection](#)

### **Syntax**

tablemarkersvalue = [objectreference].TableMarkers

### **Legal values**

Always contains an instance of the TableMarkerCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the TableMarker objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the TableMarker objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the TableMarker objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the TableMarker objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: TableOnlyContainer property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TABLEONLYCONTAINER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the TableOnlyCont class. This is a current context property which only contains an object when the focus of Word Pro includes a table. If there is no table in the focus, this property is empty.

### **Data Type**

[TableOnlyCont](#)

### **Syntax**

tableonlycontainervalue = [objectreference].TableOnlyContainer

### **Legal values**

An instance of the TableOnlyCont class.

### **Usage**

When the focus includes a table, this property contains the TableOnlyCont object which groups together the objects that comprise the table in the focus. You can use this property to access the TableLayout or other objects related to that table.

## **Word Pro: TableStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLESTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TableLayoutCollection class. This object provides access to TableLayout objects which are used as table styles.

### **Data Type**

[TableLayoutCollection](#)

### **Syntax**

tablestylesvalue = [objectreference].TableStyles

### **Legal values**

Always contains an instance of the TableLayoutCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the TableLayout objects which are used as table styles and contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the TableLayout objects which are used as table styles and contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the TableLayout objects which are used as table styles and placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the TableLayout objects which are used as table styles and contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: Tables property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TableOnlyCollection class. This object provides access to Table objects.

### **Data Type**

[TableOnlyCollection](#)

### **Syntax**

tablesvalue = [objectreference].Tables

### **Legal values**

Always contains an instance of the TableOnlyCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Table objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the Table objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the Table objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the Table objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

**Word Pro: Table property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The Table object which is uppermost in the focus when this property is called.

**Data Type**

[Table](#)

**Syntax**

tablevalue = [objectreference].Table

**Legal values**

Always contains an instance of the Table class.

**Usage**

### Word Pro: TabRack property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_CLICKHERE\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTEELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABRACK\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the TabRack object for a layout object.

### Data Type

[TabRack](#)

### Syntax

tabrackvalue = [objectreference].TabRack

### Legal values

Always contains an instance of the TabRack class.

### Usage

## **Word Pro: TempFindAndReplace property**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TEMPFINDANDREPLACE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[FindAndReplace](#)

### **Syntax**

tempfindandreplacevalue = [objectreference].TempFindAndReplace

### **Legal values**

Always contains an instance of the FindAndReplace class.

### **Usage**

## Word Pro: TempFoundry property

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEMPFOUNDRY\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A Foundry object used by Word Pro to temporarily store objects during drag and drop operations. You can use the collections in this property's Foundry object to temporarily store objects while your script is running. You must clear the TempFoundry contents after each use.

### Data Type

[Foundry](#)

### Syntax

tempfoundryvalue = [objectreference].TempFoundry

### Legal values

Always contains an instance of the Foundry class.

### Usage

TempFoundry is a property in WordPro. TempFoundry contains another Foundry object. Word Pro uses TempFoundry to temporarily store objects which are part of a drag and drop or other internal operation. You can use TempFoundry in much the same way.

Like its counterpart, AppFoundry, TempFoundry contains a Foundry object. You can use the collection objects on this Foundry object as a staging area for any Word Pro LotusScript objects you create and manipulate. For example, when you want to move an object or objects from one document to another, you can store those objects temporarily in the TempFoundry collection objects. The TempFoundry property is always available, regardless of which document is active, so you always have access to the contents of its collections. This makes it an ideal place for temporarily storing items which you want to use or move.

**Note** You must clear TempFoundry after each use. Any objects left in any of TempFoundry's collections can reappear during drag and drop and other operations, and result in unpredictable behavior.



**Word Pro: TextandTableExports property**

{button ,AL(^H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TEXTANDTABLEEXPORTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

textandtableexportsvalue = [objectreference].TextandTableExports

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

**Word Pro: TextandTableImports property**

{button ,AL(^H\_FILTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TEXTANDTABLEIMPORTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[StringCollection](#)

**Syntax**

textandtableimportsvalue = [objectreference].TextandTableImports

**Legal values**

Always contains an instance of the StringCollection class.

**Usage**

## Word Pro: TextAttributes property

{button ,AL('H\_CHARACTERSTYLE\_CLASS;H\_EDITOR\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_REVISIONDISPLAY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTATTRIBUTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Comprised of other attributes that are not normally associated as part of the font, such as highlight, hidden, or protected. This property only applies to text characteristics of specific deleted text.

### Data Type

[Attributes](#)

### Syntax

textattributesvalue = [objectreference].TextAttributes

[objectreference].TextAttributes = textattributesvalue

### Legal values

Always contains an instance of the Attributes class.

### Usage

In a Word Pro document, the font describes the typeface, point size, color, and attributes (such as bold, italic, and so on) of any inserted text. However, the TextAttributes property describes other text characteristics, such as whether the text is highlighted, hidden, or protected, and applies only to deleted text. For example, when an editor uses the markup option that hides deleted text, the TextAttributes property, as part of the Editor class, captures all the characteristics of the deleted text. That editor can then use the TextAttributes property to determine the text characteristics of specific deleted text.

## **Word Pro: TextMarkers property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTMARKERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TextMarkerCollection class. This object provides access to TextMarker objects.

### **Data Type**

[TextMarkerCollection](#)

### **Syntax**

textmarkersvalue = [objectreference].TextMarkers

### **Legal values**

Always contains an instance of the TextMarkerCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the TextMarker objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the TextMarker objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the TextMarker objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the TextMarker objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## **Word Pro: TextStyles property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTSTYLES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TextStyleCollection class. This object provides access to both CharacterStyle and ParagraphStyle objects.

### **Data Type**

[TextStyleCollection](#)

### **Syntax**

textstylesvalue = [objectreference].TextStyles

### **Legal values**

Always contains an instance of the TextStyleCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the CharacterStyle and ParagraphStyle objects contained in that Division object.

When accessed through the AppFoundry property on the WPApplication object, this collection object provides access to all the CharacterStyle and ParagraphStyle objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPApplication object, this collection object provides access to all the CharacterStyle and ParagraphStyle objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPApplication object, this collection object provides access to all the CharacterStyle and ParagraphStyle objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see [Overview: Word Pro LotusScript Collection](#)

[ClassesH\\_WORD\\_PRO\\_LOTUSSCRIPT\\_COLLECTION\\_CLASSES\\_OVER](#).

## **Word Pro: Texts property**

{button ,AL('H\_FOUNDRY\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the TextCollection class. This object provides access to Text objects.

### **Data Type**

[TextCollection](#)

### **Syntax**

textsvalue = [objectreference].Texts

### **Legal values**

Always contains an instance of the TextCollection class.

### **Usage**

When accessed through the Foundry property on a Division object, the collection object in this property provides access to all the Text objects contained in that Division object.

When accessed through the AppFoundry property on the WPAApplication object, this collection object provides access to all the Text objects contained in the Word Pro Clipboard.

When accessed through the TempFoundry property on the WPAApplication object, this collection object provides access to all the Text objects placed in TempFoundry by WordPro or a script.

When accessed through the Foundry property on the WPAApplication object, this collection object provides access to all the Text objects contained in the currently active Division object.

This property is not used in the Foundry property on the TextDocument object.

For more information about collection classes, see Overview: Word Pro LotusScript Collection

ClassesH\_WORD\_PRO\_LOTUSSCRIPT\_COLLECTION\_CLASSES\_OVER.

## Word Pro: Text property

{button ,AL('H\_BULLET\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_OLEOBJECT\_CLASS;H\_SILVERBULLET\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An object created from the Text class.

## Data Type

Text

## Syntax

textvalue = [objectreference].Text

## Legal values

Always contains an instance of the Text class.

## Usage

The Text object you retrieve when you call this property is determined by the object from which you call the property.

*CurrentApplication.Text*

When you call this property from the WPAApplication object, the property returns the Text object which is uppermost in the focus when this property is called.

**Note** The remainder of this topic is not yet complete.

*<Bulletobject>.Text*

When you call this property from a Bullet object, the property returns the Text object which

*<GraphicOleObjectobject>.Text*

When you call this property from a GraphicOleObject object, the property returns the Text object which

*<Graphicobject>.Text*

When you call this property from a Graphic object, the property returns the Text object which

*<NoteLayoutobject>.Text*

When you call this property from a NoteLayout object, the property returns the Text object which

*<OleObjectobject>.Text*

When you call this property from a OleObject object, the property returns the Text object which

*<SilverBulletonject>.Text*

When you call this property from a SilverBullet object, the property returns the Text object which

**Word Pro: TopBorder property**

{button ,AL(^H\_BORDERLINES\_CLASS;H\_GUTTER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TOPBORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Border](#)

**Syntax**

topbordervalue = [objectreference].TopBorder

**Legal values**

Always contains an instance of the Border class.

**Usage**



## **Word Pro: UndoLevels property**

{button ,AL(^H\_REVISIONDISPLAY\_CLASS;H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UNDOLEVELS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[StringCollection](#)

### **Syntax**

undolevelsvalue = [objectreference].UndoLevels

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

**Word Pro: Units property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UNITS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A collection of all the units of measurement available in Word Pro.

**Data Type**

[StringCollection](#)

**Syntax**

unitsvalue = [objectreference].Units

**Legal values**

Always contains an instance of the StringCollection class with the following members: Inches (in), Centimeters (cm), Points (pts), and Picas (pi).

**Usage**

This property is used to populate the "Measure in" box on the General panel of the Word Pro Preferences dialog box.

## **Word Pro: UserDictionaryFiles property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERDICTIONARYFILES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Stores multiple user dictionary names.

### **Data Type**

[StringCollection](#)

### **Syntax**

userdictionaryfilesvalue = [objectreference].UserDictionaryFiles

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "Default user dictionary" value on the Default files panel of the Word Pro Preferences dialog box. In the Word Pro interface, "Default user dictionary" can contain multiple file names. You can use this property to read these multiple file names, including the default file which is stored in the UserDictFiles property.

## **Word Pro: UserDictionaryPaths property**

{button ,AL(^H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_USERDICTIONARYPATHS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains multiple paths (drive and directory) for the Word Pro user dictionary.

### **Data Type**

[StringCollection](#)

### **Syntax**

userdictionarypathsvalue = [objectreference].UserDictionaryPaths

### **Legal values**

Always contains an instance of the StringCollection class.

### **Usage**

Equivalent to the "User dictionaries" option on the Locations panel of the Word Pro Preferences dialog box. This value can contain multiple paths. You can use this property to read these multiple paths, including the default or first path. The default path is stored in the UserDictionaryPath property.

**Word Pro: UserInterfacePrefs property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERINTERFACEPREFS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) An instance of the UserInterfacePrefs class.

**Data Type**

[UserInterfacePrefs](#)

**Syntax**

userinterfaceprefsvalue = [objectreference].UserInterfacePrefs

**Legal values**

Always contains an instance of the UserInterfacePrefs class.

**Usage**

Use this property to set Word Pro user preferences when no document is open.

## Word Pro: UseWhen property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_USEWHEN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Allows you to determine when a specific layout object should be used.

## Data Type

[UseWhen](#)

## Syntax

usewhenvalue = [objectreference].UseWhen

## Legal values

Always contains an instance of the UseWhen class.

## Usage

**Word Pro: VersionManager property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_VERSIONMANAGER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[VersionManager](#)

**Syntax**

versionmanagervalue = [objectreference].VersionManager

**Legal values**

Always contains an instance of the VersionManager class.

**Usage**

**Word Pro: Versions property**

{button ,AL('H\_VERSIONMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERSIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[VersionCollection](#)

**Syntax**

versionsvalue = [objectreference].Versions

**Legal values**

Always contains an instance of the VersionCollection class.

**Usage**



**Word Pro: VertRuler property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERTRULER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Contains a ruler object that indicates tab settings, indents, margins, and columns.

**Data Type**

[Ruler](#)

**Syntax**

verrulervalue = [objectreference].VertRuler

**Legal values**

Always contains an instance of the Ruler class.

**Usage**

Use this property to display the vertical ruler when no document is open.

**Word Pro: WinViewPrefs property**

{button ,AL('H\_DOCWINDOW\_CLASS;H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WINVIEWPREFS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The pointer to the view preferences options in the WinViewPrefs class.

**Data Type**

[WinViewPrefs](#)

**Syntax**

winviewprefsvalue = [objectreference].WinViewPrefs

**Legal values**

Always contains an instance of the WinViewPrefs class.

**Usage**

## Word Pro: WPDataSets property

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_CLICK  
HERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H  
_DROPCAPLAYOOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYO  
UT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HE  
ADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_  
CLASS;H_PARAGRAPHSTYLE_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLA  
YOUT_CLASS;H_RUBYLAYOOUT_CLASS;H_RUBYMAREK_CLASS;H_SUPERTABLEGROUPLAYOOUT_CLASS  
;H_SUPERTABLELAYOOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOOUT_CLASS;H_TALE  
MARKER_CLASS;H_TEXT_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTA  
BLELAYOOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL(^H_WPDATASETS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) This property contains an instance of the WPDataSetCollection class.

## Data Type

[WPDataSetCollection](#)

## Syntax

wpdatasetsvalue = [objectreference].WPDataSets

## Legal values

Always contains an instance of the WPDataSetCollection class.

## Usage

This WPDataSetCollection object gives you access to WPDataSet objects. For more information, see the WPDataSetCollection and WPDataSet classes.

**Word Pro: Zero property**

{button ,AL(^H\_NUMERICFORMAT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_ZERO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[NumericFormatSubset](#)

**Syntax**

zerovalue = [objectreference].Zero

**Legal values**

Always contains an instance of the NumericFormatSubset class.

**Usage**

## **Word Pro: SortParagraphs method**

{button ,AL(^H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS;H\_WPAPPLICATION\_CLASS',0)}

[See list of classes](#)

{button ,AL(^H\_SORTPARAGRAPHS\_METHOD\_EXSCRIPT',1)} [See example](#)

Sorts the currently active text object or selection, based on the settings in the SortOptions object.

### **Syntax**

[objectreference].SortParagraph()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

This method behaves the same way as the Sort function found on the Word Pro Text menu.

If the focus is in a table cell or table selection when you call this method, Word Pro sorts the entire table by the contents of the first column.

## **Word Pro: SpellAddToUserDict method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPELLADDTouserDict\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds the selected word to the user dictionary.

### **Syntax**

[objectreference].SpellAddToUserDict()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SpellClearSkippedWords method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_SPELLCLEARSKIPPEDWORDS\_METHOD\_EXSCRIPT',1)} [See example](#)

Removes the Skip flag from words you have instructed Word Pro to skip.

### **Syntax**

[objectreference].SpellClearSkippedWords()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SpellMarkSkippedWords method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLMARKSKIPPEDWORDS\_METHOD\_EXSCRIPT',1)} [See example](#)

Marks the selected word as skipped.

**Note** This method also removes the flags for misspelled or double words.

### **Syntax**

[objectreference].SpellMarkSkippedWords()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**



## **Word Pro: SpellSkipAll method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_SPELLSKIPALL\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds the selected word to Spell Check's skip all list.

### **Syntax**

[objectreference].SpellSkipAll()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SpellWord method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPELLWORD\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SpellWord()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: SplitParagraph method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SPLITPARAGRAPH\_METHOD\_EXSCRIPT',1)} [See example](#)

Splits the current paragraph into two separate paragraphs.

### **Syntax**

[objectreference].SplitParagraph([PropagateAttributes])

### **Parameters**

#### *PropagateAttributes*

Allows you to carry over any local paragraph attributes from the existing paragraph to the new paragraph. The data type for this parameter is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0) instead of the integer values. This is an optional parameter. The default value is False which passes on only the paragraph style attributes.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: Split method

{button ,AL(`H\_BASSETABLE\_CLASS;H\_FOOTNOTETABLE\_CLASS;H\_GLOSSARY\_CLASS;H\_PARALLELCOLUMN  
NS\_CLASS;H\_TABLE\_CLASS;H\_TABLEHEADING\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_SPLIT\_METHOD\_EXSCRIPT',1)} [See example](#)

Splits a cell into two or more rows or columns.

### Syntax

[objectreference].Split(SplitType, [NumRows,] [NumCols])

### Parameters

#### *SplitType*

Data type is Variant. The value of this parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpTableSplitTypeCell (1904)	Equivalent to choosing Table - Split cell. Splits the current cell into the number of cells specified by the NumRows and NumCols parameter values.
\$LwpTableSplitTypeTable (1905)	Equivalent to choosing Table - Split Entire Table. Values in the NumRows and NumCols parameters have no effect.

#### *NumRows*

Optional Integer parameter. Determines how many rows into which a cell will be split.

#### *NumCols*

Optional Integer parameter. Determines how many columns into which a cell will be split.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

## **Word Pro: SRReplace method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SRREPLACE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].SRReplace([IsTemporary])

### **Parameters**

*IsTemporary*

Data type is Boolean. Optional parameter. Default value is False.

### **Return value**

### **Usage**

## **Word Pro: StartEditMergeData method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_STARTEDITMERGEDATA\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens the data file for the currently active merge document. The data file remains hidden from the user but available for editing from the merge document. Equivalent to choosing Text - Merge and clicking Edit Data File.

### **Syntax**

[objectreference].StartEditMergeData()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: StartEnvelopeDiv method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STARTENVELOPEDIV\_METHOD\_EXSCRIPT',1)} [See example](#)

Creates an envelope division in the currently active document.

### **Syntax**

[objectreference].StartEnvelopeDiv()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

This method performs the following tasks in the course of creating the envelope division:

- Creates frames for the return address and delivery address.
- Checks for selected text in the active document.
- If text is selected, Word Pro copies that text into the delivery address frame.
- If no text is selected, Word Pro looks at the beginning of the document for 3 to 6 lines which have fewer than 60 characters each.
- If Word Pro finds lines which meet these criteria, Word Pro assumes those lines are the delivery address and copies them into the delivery address frame.

**Word Pro: StartFieldInsert method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STARTFIELDINSERT\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens the Merge bar so the user can insert fields for merging. This is the same Merge bar which appears during step two of the automated merge process.

**Syntax**

[objectreference].StartFieldInsert()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**



## Word Pro: Start method

```
{button ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_DROPCAPCONTAINER_CLASS;H_FR  
AMECONTAINER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCO  
LSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLA  
SS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLAS  
S;H_TABLEONLYCONT_CLASS;H_USEWHEN_CLASS',0)} See list of classes
```

```
{button ,AL('H_START_METHOD_EXSCRIPT',1)} See example
```

In any container, this method moves the insertion point from its current position to the beginning of the document.

## Syntax

[objectreference].Start(OfWhat)

## Parameters

### *OfWhat*

Data type is Variant. The value of this parameter must be \$LwpDocumentObjectTypeDocument or its code equivalent (216).

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: StoreInternetFile method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STOREINTERNETFILE\_METHOD\_EXSCRIPT',1)} [See example](#)

Uses FTP to transfer a file from your local machine to an Internet site. You can transfer any file including Word Pro documents and HTML files.

### Syntax

[objectreference].StoreInternetFile()

### Parameters

#### *LocalFile*

A String expression specifying the name and path of the file which you want to send to an Internet site.

#### *URL*

A String expression specifying URL for the FTP server that will receive the file. This string value must include the name of the directory in which you want to store the file.

#### *UserID*

A String expression representing the name of the user who has an account with the FTP server.

#### *Password*

A String expression representing the password for the user named in UserID.

#### *Passive*

Set this value to True when you want to initiate the file transfer. Set it to False to allow the server to respond to your request when it is ready. Some FTP servers do not support this feature. The value of this parameter is usually False. If you are retrieving a WWW document, the value of this parameter should be a null string (""). Data type is Integer but the value is always 0 (False) or -1 (True). You can use the LotusScript constants of True and False.

#### *Proxy*

A String expression specifying the DNS (for example, *screen.companyname.com*) or IP address (for example, *123.456.78.912*). Do not include "http:\\" in front of the the proxy value.

#### *ProxyPort*

An Integer which specifies the port number for the Proxy server. The value of this parameter is usually 8080 for the WWW and 21 for FTP, but you should check with your Internet service provider for your settings.

### Return value

A String representing the name of the file you sent to the FTP site.

### Usage

This method will not work unless your machine is configured for Internet access. A standard Internet access configuration includes a WINSOCK compliant DLL.

## **Word Pro: StrikeThru method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_STRIKETHRU\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the strikethrough attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on and on if it is off. This is the same as choosing Text - Attributes - Other and then choosing "Strikethrough" from the Attributes box.

### **Syntax**

[objectreference].StrikeThru()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: SubScript method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUBSCRIPT\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the subscript attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on and on if it is off. This is the same as choosing Text - Attributes - Other and then choosing "Subscript" from the Attributes box.

### **Syntax**

[objectreference].SubScript()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: Sum method

{button ,AL('H\_TABLE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUM\_METHOD\_EXSCRIPT',1)} [See example](#)

Adds the contents of a range of cells within a table.

### Syntax

[objectreference].Sum(TableSumScope)

### Parameters

*TableSumScope*

Adds the contents in a range of columns or rows. The value of this Variant parameter must be one of the strings below or its code equivalent.

<u>Value</u>	<u>Effect</u>
\$LwpTableSumScopeColumn (1907)	Adds the contents of a range of columns within a table.
\$LwpTableSumScopeRow (1906)	Adds the contents of a range of rows within a table.

### Return value

This method returns an Integer value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

Equivalent to inserting a SmartSum formula.

## **Word Pro: SuperScript method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUPERSCRIPT\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the Superscript attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on and on if it is off. This is the same as choosing Text - Attributes - Other and then choosing "Superscript" from the Attributes box.

### **Syntax**

[objectreference].SuperScript()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

**Word Pro: TeamMail method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_TEAMMAIL\_METHOD\_EXSCRIPT',1)} [See example](#)

Opens and displays the TeamMail dialog box. Equivalent to choosing File - TeamMail.

**Syntax**

[objectreference].TeamMail()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

## **Word Pro: TerminateFormatCheck method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXTMARKER\_CLASS;H\_TEXT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TERMINATEFORMATCHECK\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].TerminateFormatCheck()

### **Parameters**

### **Return value**

### **Usage**



## **Word Pro: TextNumber method**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTNUMBER\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].TextNumber(NumberObjectType, StartingNumber)

### **Parameters**

*NumberObjectType*

Data type is Variant. The value of this parameter must be \$LwpNumObjTypeMulticompareParatag or its code equivalent (1530).

*StartingNumber*

Data type is Integer.

### **Return value**

### **Usage**

## **Word Pro: TheoreticalScaledSize method**

{button ,AL('H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_THEORETICALSCALED\_SIZE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].TheoreticalScaledSize(ScaleMode, Percentage, Width, Height, MaintainAspectRatio)

### **Parameters**

*ScaleMode*

Data type is Integer.

*Percentage*

Data type is Long.

*Width*

Data type is Long.

*Height*

Data type is Long.

*MaintainAspectRatio*

Data type is Integer.

### **Return value**

Long

### **Usage**

**Word Pro: TileWindowHorz method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_TILEWINDOWHORZ\_METHOD\_EXSCRIPT',1)} [See example](#)

Resizes and arranges all the active document windows so they appear side by side in the Word Pro application window. Equivalent to choosing Window - Tile Left-Right.

**Syntax**

[objectreference].TileWindowHorz()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

If more than two windows are open, Word Pro may arrange the windows in rows so the first windows are tiled left to right in the top row, and the remaining windows are tiled in more rows beneath.

**Word Pro: TileWindowVert method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TILEWINDOWVERT\_METHOD\_EXSCRIPT',1)} [See example](#)

Resizes and arranges all the active document windows so they are one above the other in the Word Pro application window. This is the same as choosing Window - Tile Top-Bottom.

**Syntax**

[objectreference].TileWindowVert()

**Parameters**

None

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**

If more than two windows are open, Word Pro may arrange the windows in columns so the first windows are tiled top to bottom on the left and the remaining windows are tiled in columns to the right.

## Word Pro: Tile method

```
{button ,AL('H_APPLICATIONWINDOW_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLAYOUT_CLASS;H_CO  
LUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPPLAYOUT_CLASS;H_ENDNOTE  
LAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CL  
ASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS  
;H_NOTELAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW_LAYOUT_CLAS  
S;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_T  
ABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list  
of classes
```

```
{button ,AL('H_TILE_METHOD_EXSCRIPT',1)} See example
```

Tiles the document windows within the Word Pro application window.

## Syntax

[objectreference].Tile()

## Parameters

None

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

This method is equivalent to choosing Window - Tile Left-Right in the Word Pro interface.

## Word Pro: TimedSave method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TIMEDSAVE\_METHOD\_EXSCRIPT',1)} [See example](#)

Performs a Timed Save of all the open documents which have been saved at least once.

### Syntax

[objectreference].TimedSave()

### Parameters

None

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

This Timed Save is the procedure which Word Pro performs automatically if you choose File - User Setup - Word Pro Preferences and select the option, "Automatically time save every". This method will perform a Timed Save regardless of whether a document's automatic save option is selected.

In a Timed Save, Word Pro creates a .~TS file for the document. The .~TS file is a copy of the document, located in the same folder as the original document.

Word Pro updates the .~TS file each time it autosaves the document. When you save a document, Word Pro saves the .~TS file to the original document and deletes the .~TS file. When you close a document without saving it, Word Pro deletes the .~TS file without saving it to the original document.

If you exit Word Pro abnormally, the .~TS files are not deleted. The next time you start Word Pro or open the original document, Word Pro prompts you about opening the .~TS file. If you do not open the .~TS file, Word Pro deletes it.

## **Word Pro: ToggleIconBar method**

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TOGGLEICONBAR\_METHOD\_EXSCRIPT',1)} [See example](#)

Shows or hides the currently active set of SmartIcons. Equivalent to choosing View - Show/Hide and choosing SmartIcons.

### **Syntax**

[objectreference].ToggleIconBar()

### **Parameters**

None

### **Return value**

None

### **Usage**

## Word Pro: Type method

{button ,AL(^H\_CHARACTERSTYLE\_CLASS;H\_FOOTNOTE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_POWERFIELD\_CLASS;H\_SPACING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TYPE\_METHOD\_EXSCRIPT',1)} [See example](#)

Performs the specified keystrokes in a document. Available keystrokes include insertion point movement and function keys as outlined below in Parameters.

### Syntax

[objectreference].Type(Keystroke)

### Parameters

#### *Keystroke*

A String expression which represents the characters you want Word Pro to type in the document. To type a double quote mark, you must use two double quotes so that LotusScript can distinguish between the double quote you want to type and those which surround the rest of the string.

For example, this statement:

```
.Type("a double quote "" in a document")
```

would type this in your document:

```
a double quote " in a document
```

You can also include an insertion point movement or function key. To type a key, surround its name with square braces. The following key names can be used:

[Home] - Home key

[End] - End key

[PgUp] - Page Up key

[PgDn] - Page Down key

[Ins] - Insert key

[Del] - Delete key

[Backspace] - Backspace key

[Enter] - Enter or Return key

[Tab] - Tab key

[ESC] - Escape key

[Up] - Up Arrow key

[Down] - Down Arrow key

[Left] - Left Arrow key

[Right] - Right Arrow key

[F1] - [F12] - Function keys F1 through F12

You can also add the standard modifiers (CTRL, SHIFT and ALT) to these keys by appending "CTRL", "SHIFT" or "ALT", in any combination, to the front of the key name. For example, "CTRLDown" is the same as holding down the CTRL key while pressing the Down Arrow key and "CTRLSHIFTDown" is the same as holding down the CTRL and SHIFT keys while pressing the Down Arrow key.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage



## **Word Pro: Underline method**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UNDERLINE\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the underline attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on and on if it is off. This is the same as choosing Text - Attributes - Other and then choosing "Underline" from the Attributes box.

### **Syntax**

[objectreference].Underline()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: UndoRedo method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UNDOREDO\_METHOD\_EXSCRIPT',1)} [See example](#)

Allows you to reverse (undo) or repeat (redo) actions in Word Pro. Equivalent to choosing Edit - Undo/Redo Special.

### **Syntax**

[objectreference].UndoRedo()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: Undo method**

{button ,AL('H\_GRAPHIC\_CLASS;H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UNDO\_METHOD\_EXSCRIPT',1)} [See example](#)

Reverses (undoes) the previous editing function. Equivalent to choosing Edit - Undo.

### **Syntax**

When called from WPAApplication or a Graphic object:

[Objectreference].Undo()

When called from a TextDocument object:

[Objectreference].Undo(Count)

### **Parameters**

#### *Count*

Specifies the number of recent edits to undo. This is an optional parameter only available when this method is called from a TextDocument object. The data type is Integer and the default value is 1.

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: UnregisterWPDataSet method

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_CLICK  
HERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H  
_DROPCAPLAYOOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYO  
UT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HE  
ADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_  
CLASS;H_PARAGRAPHSTYLE_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLA  
YOUT_CLASS;H_RUBYLYAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS  
;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLE  
MARKER_CLASS;H_TEXT_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTA  
BLELAYOUT_CLASS;H_WPAPPLICATION_CLASS',0)} See list of classes
```

```
{button ,AL(^H_UNREGISTERWPDATASET_METHOD_EXSCRIPT',1)} See example
```

Deletes a WPDataSet from the object from which you call this method.

### Syntax

```
[objectreference].UnregisterWPDataSet(GroupName)
```

### Parameters

*GroupName*

A string expression representing the name given to the WPDataSet.

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

WPDataSet objects are useful tools which store data with a document. When you close a document that has one or more data sets attached to it, Word Pro saves the data set(s) with the document. Any time the document is open, you have access to the data sets created for that document.

When you register or unregister a WPDataSet on a Text object, that WPDataSet is assigned to the currently active paragraph.

This method deletes a WPDataSet object. The deleted data set cannot be restored once it has been unregistered.

## **Word Pro: UpdateFootersText method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEFOOTERSTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates all the footers in a document to match the footer content in the currently active division.

### **Syntax**

[objectreference].UpdateFootersText()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: UpdateHeadersText method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEHEADERSTEXT\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates all the headers in a document to match the header content in the currently active division.

### **Syntax**

[objectreference].UpdateHeadersText()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: UpdateIndexSection method**

{button ,AL(^H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEINDEXSECTION\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates the specified index section to reflect any changes made to the associated division or document.

### **Syntax**

[objectreference].UpdateIndexSection(DivisionName, SectionName)

### **Parameters**

*DivisionName*

Data type is String.

*SectionName*

Data type is String.

### **Return value**

### **Usage**

## **Word Pro: UpdateLink method**

{button ,AL('H\_GRAPHIC\_CLASS;H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATELINK\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].UpdateLink(ShowErrorMessage)

### **Parameters**

*ShowErrorMessage*

Data type is Integer. The legal values for this parameter are -1 and 0 but you may use the LotusScript constants True (-1) and False (0).

### **Return value**

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: UpdateOle method**

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATEOLE\_METHOD\_EXSCRIPT',1)} [See example](#)

When the Word Pro document is OLE embedded in another container, this method updates the Word Pro document which is embedded.

### **Syntax**

[objectreference].UpdateOle()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## **Word Pro: UpdatePageSizeChange method**

{button ,AL(`H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_UPDATEPAGESIZECHANGE\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates any changes to the size of the paper used to print the document.

### **Syntax**

[objectreference].UpdatePageSizeChange()

### **Parameters**

### **Return value**

### **Usage**

## **Word Pro: UpdatePowerFields method**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATEPOWERFIELDS\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates any changes to power fields in a division.

### **Syntax**

[objectreference].UpdatePowerFields(Reset)

### **Parameters**

*Reset*

Data type is Boolean.

### **Return value**

### **Usage**

## **Word Pro: UpdatePrinterChanges method**

{button ,AL(^H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEPRINTERCHANGES\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates any changes to power fields in a document, power fields used to create a table of contents, and power fields used to create indexes. Prints the results at the location of the power field.

### **Syntax**

[objectreference].UpdatePrinterChanges()

### **Parameters**

### **Return value**

### **Usage**

## Word Pro: UpdateSelectedFields method

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATESELECTEDFIELDS\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates selected DocInfo fields in a document.

### Syntax

[objectreference].UpdateSelectedFields([FieldUpdateSelect])

### Parameters

*FieldUpdateSelect*

Data type is Variant. Default value is \$LwpFieldUpdateWordsPagesFilesize. The value of this parameter must be one of the strings below or its equivalent (in parentheses):

<u>Value</u>	<u>Effect</u>
\$LwpFieldUpdateChars (257)	Changes and displays within a DocInfo field the current number of characters in a document.
\$LwpFieldUpdatePages (256)	Changes and displays within a DocInfo field the current number of pages in a document.
\$LwpFieldUpdateSize (258)	Changes and displays within a DocInfo field the current kilobyte size of a document.
\$LwpFieldUpdateWords (255)	Changes and displays within a DocInfo field the current number of words in a document.
\$LwpFieldUpdateWordsPagesFilesize (254)	Changes and displays within a DocInfo field the current number of pages, the number of words, and the file size of a document.

### Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### Usage

## **Word Pro: UpdateTOC method**

{button ,AL(`H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(`H\_UPDATETOC\_METHOD\_EXSCRIPT',1)} [See example](#)

Updates any changes to a table of contents. Equivalent to choosing Create - Other Document Part, choosing Table of Contents, and clicking Update to display the Update Table of Contents dialog box.

### **Syntax**

[objectreference].UpdateTOC(DivisionName, TOCName)

### **Parameters**

*DivisionName*

Data type is String.

*TOCName*

Data type is String.

### **Return value**

### **Usage**

## Word Pro: Update method

```
{button ,AL(^H_APPLICATIONWINDOW_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACTERSTYLE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DOCWINDOW_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPHSTYLE_CLASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_STATUSBAR_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_WINDOW_CLASS',0)} See list of classes
```

```
{button ,AL(^H_UPDATE_METHOD_EXSCRIPT',1)} See example
```

[CharacterStyle]

Takes the referenced character style and propagates that character style to all divisions. Creates where it doesn't exist and updates existing character styles.

[Layout]

Takes the referenced layout style and propagates that layout style to all the divisions. When this method is invoked, it creates a layout style if one doesn't exist and updates an existing layout style.

[ParagraphStyle]

[PowerField]

[Window]

[ApplicationWindow]

Forces a repaint of the application window

## Syntax

```
[objectreference].IconBarManager.Update()
```

```
[objectreference].StatusBar.Update()
```

```
[objectreference].ApplicationWindow.Update()
```

## Parameters

## Return value

The return value for this method will always be -1 or 0. When testing the return value, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

Layout - No return value.

ApplicationWindow - No return value.

## Usage

This method is not valid for IconBarManager and StatusBar.

## **Word Pro: UpperCase method**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPPERCASE\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the Upper Case attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on and on if it is off. This is the same as choosing Text - Attributes - Other and then choosing "Upper Case" from the Attributes box.

### **Syntax**

[objectreference].UpperCase()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**



**Word Pro: ValidateValue method**

{button ,AL('H\_CELLLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VALIDATEVALUE\_METHOD\_EXSCRIPT',1)} [See example](#)

Determines whether or not a table cell contains numeric or string content.

**Syntax**

[objectreference].ValidateValue()

**Parameters****Return value**

Integer. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: WMCommand method

{button ,AL('H\_TEXTDOCUMENT\_CLASS;H\_WPAPPLICATION\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WMCOMMAND\_METHOD\_EXSCRIPT',1)} [See example](#)

Issues the standard Word Pro menu command specified in the CommandID parameter. The complete list of available menu commands and their corresponding CommandID values is included below.

**Note** To use these menu commands, you must have the WPBITMSK.LSS file in the Word Pro directory.

## Syntax

[objectreference].WMCommand(CommandID)

## Parameters

### CommandID

A numeric expression of type Integer which specifies the menu command you want to issue. The values in the WPBITMSK.LSS file are expressed as hexadecimal numbers.

Command	ID	Command	ID
LwpMenuFilemenu	100	LwpMenuMIFielddoauto	226
LwpMenuMfNew	101	LwpMenuMIFieldremove	227
LwpMenuMfOpen	102	LwpMenuMIExechand	574
LwpMenuMfSave	103	LwpMenuMIIndexopts	228
LwpMenuMfSaveas	104	LwpMenuHelpmenu	550
LwpMenuMfRevert	105	LwpMenuMhIdxhelp	551
LwpMenuMfAppendtext	106	LwpMenuMhAbout	552
LwpMenuMfImport	107	LwpMenuMhMacrohelp	553
LwpMenuMfFilemanage	108	LwpMenuMhCompatible	555
LwpMenuMfDocdesc	109	LwpMenuMhUsingHelp	830
LwpMenuMfCreatedatafile	110	LwpMenuMhKeyboard	831
LwpMenuMfPrint	111	LwpMenuMhHowDol	832
LwpMenuMfChgprinter	112	LwpMenuMhDoHelp	833
LwpMenuMfExit	113	LwpMenuMhForUpgraders	834
LwpMenuMfPrintreset	114	LwpMenuMhTutorial	837
LwpMenuMfDde	115	LwpMenuMhSearch	879
LwpMenuMfMergeaction	116	LwpMenuMsControl	601
LwpMenuMfJustprint	117	LwpMenuMsFileman	602
LwpMenuMfStandalonep	327	LwpMenuMtTblayout	626
LwpMenuMfAllfldsrename	118	LwpMenuMtInscolorw	627
LwpMenuMfPrintwrongpaper	119	LwpMenuMtDelcolorw	628
LwpMenuMfPrintcancel	120	LwpMenuMtEditfrmla	629
LwpMenuMfDummymrgsel	121	LwpMenuMtTbladd	630
LwpMenuMfDummymrgext	122	LwpMenuMtHeading	631
LwpMenuMfAsciiopts	123	LwpMenuMtTableinfobox	633
LwpMenuMfCanprint	124	LwpMenuMtSizecolorw	634
LwpMenuMfFldrename	125	LwpMenuMtConnect	635
LwpMenuMfPrintopt	126	LwpMenuMtSavefrmla	637
LwpMenuMfSaveasnw	127	LwpMenuMtQuickaddrow	643
LwpMenuMfImportnw	128	LwpMenuMtQuickaddcol	644
LwpMenuMfDocdescnw	129	LwpMenuMtDelcolumn	645
LwpMenuMfMasternw	130	LwpMenuMtDelrow	646

LwpMenuMfNwimport	131	LwpMenuMtInsertrow	647
LwpMenuMfMergeviewprint	132	LwpMenuMtInsertcol	648
LwpMenuMfLastopen1	133	LwpMenuMtInsertaccel	649
LwpMenuMfLastopen2	134	LwpMenuMtDeleteaccel	650
LwpMenuMfLastopen3	135	LwpMenuMtDeltable	651
LwpMenuMfLastopen4	136	LwpMenuMtSelectcolumn	652
LwpMenuMfLastopen5	137	LwpMenuMtSelectrow	653
LwpMenuMfOpennostyle	138	LwpMenuMtSelecttable	654
LwpMenuMfMergelabels	139	LwpMenuMtChgiconsize	655
LwpMenuMfOtherflds	140	LwpMenuMtDisconnect	656
LwpMenuMfImportpicture	141	LwpMenuMtSplitcells	657
LwpMenuMfExport	142	LwpMenuMtSplittable	658
LwpMenuMfAttribnw	143	LwpMenuMtPcolinscolrow	659
LwpMenuMfCanmergeprint	146	LwpMenuMtPcolinsertrow	660
LwpMenuMfImporttext	152	LwpMenuMtPcolinsertcol	661
LwpMenuMfMergeprintopts	565	LwpMenuMtPcoldeltable	662
LwpMenuMfClose	144	LwpMenuMtPcoldelcolumn	663
LwpMenuMfCloseall	731	LwpMenuMtPcoldelrow	664
LwpMenuMfCloseallreplacelast	732	LwpMenuMtPcolconnect	665
LwpMenuMfPassword	145	LwpMenuMtPcoldisconnect	666
LwpMenuMfPassword2	290	LwpMenuMtPcolsplitcells	667
LwpMenuMfMasterdocument	736	LwpMenuMtPcolselectcolumn	668
LwpMenuMfClosefile	147	LwpMenuMtPcolselectrow	669
LwpMenuMfLock	148	LwpMenuMtPcolselecttable	670
LwpMenuMfNextmdiwindow	727	LwpMenuMtPcoldelcolrow	671
LwpMenuMfMail	229	LwpMenuMtPcolheading	672
LwpMenuMfMailNew	737	LwpMenuMtCellinfoebox	673
LwpMenuMfMailread	230	LwpMenuMtShowtablelineinfoebox	674
LwpMenuMfQuickopen	231	LwpMenuMtShowtablesizeinfoebox	675
LwpMenuMfFilesep	235	LwpMenuMtCreatetablewithgrid	676
LwpMenuMfSavecopyas	292	LwpMenuMtShowpcollineinfoebox	677
LwpMenuMfUpdateobject	293	LwpMenuMtShowpcolsizeinfoebox	678
LwpMenuMfEditorgreet	733	LwpMenuMtSplitpcol	679
LwpMenuMfNewdivision	734	LwpMenuMtPcolselectentiretable	680
LwpMenuMfOpendivision	735	LwpMenuMtSelectentiretable	681
LwpMenuMfSaContinueRoute	738	LwpMenuMtSelectentirecellrange	682
LwpMenuMfSaRoute	739	LwpMenuMtSelectentirepcolcellrange	683
LwpMenuMfSeldatafile	780	LwpMenuMtTabletopalign	684
LwpMenuMfMrgdelimit	780	LwpMenuMtTablecenteralign	685
LwpMenuMfMrgletter	781	LwpMenuMtTablebottomalign	686
LwpMenuMfMrgenvelope	786	LwpMenuMtConnectrow	687
LwpMenuMfMrggotorecord	782	LwpMenuMtPcolconnectrow	688
LwpMenuMfCreatedesc	783	LwpMenuMtGotonextblock	689
LwpMenuMfMrgsort	784	LwpMenuMtCellinfoeboxmouse	690
LwpMenuMfCreatemerge	785	LwpMenuMtTableinfoeboxmouse	691

LwpMenuMfEditdatafile	787	LwpMenuMxRunrexx	699
LwpMenuMfUsecurasmrgdoc	788	LwpMenuMxRunontime	700
LwpMenuMfOpenandeditdatafile	789	LwpMenuMxHourglass	720
LwpMenuMfOpenpreview	790	LwpMenuMxMacpause	721
LwpMenuMfNewpreview	791	LwpMenuMxMacresume	722
LwpMenuMxMergemacro	792	LwpMenuMxTimedelay	723
LwpMenuMfMergeprintall	793	LwpMenuMxTestforcancel	724
LwpMenuMfMergeopen	794	LwpMenuMxMacmessages	725
LwpMenuMIsRunscript	795	LwpMenuMxPreanswermsgbox	726
LwpMenuMIsCreatescript	863	LwpMenuMxMaccancel	728
LwpMenuMIsCreatedialog	864	LwpMenuMxMaccontinue	729
LwpMenuMIsCompile	869	LwpMenuMxMacrovariables	730
LwpMenuMIsAutoapp	873	LwpMenuMxRunautonew	798
LwpMenuMIsIdeStepIn	960	LwpMenuMxRunautoopen	799
LwpMenuMIsIdeStepOver	961	LwpMenuOutlinemenu	750
LwpMenuMIsIdeStepExit	962	LwpMenuMolPromote	751
LwpMenuMIsIdeContinue	963	LwpMenuMolDemote	752
LwpMenuMIsIdeStop	964	LwpMenuMolMoveup	753
LwpMenuMIsIdeSetBp	965	LwpMenuMolMovedown	754
LwpMenuMIsIdeDisBp	966	LwpMenuMolExpand	755
LwpMenuMIsIdeClearAll	967	LwpMenuMolContract	756
LwpMenuMIsIdeDisAll	968	LwpMenuMolShowlevels	757
LwpMenuMIsIdeBreakpoints	969	LwpMenuMolExpandsingle	758
LwpMenuMIsIdeBrowser	970	LwpMenuMolContractsingle	759
LwpMenuMIsIdeOutput	971	LwpMenuMolUseoutlinestyle	760
LwpMenuMIsIdeVariables	972	LwpMenuMolSetshowlevel1	761
LwpMenuMIsIdeCheckScript	973	LwpMenuMolSetshowlevel2	762
LwpMenuMIsIdeNewSub	974	LwpMenuMolSetshowlevel3	763
LwpMenuMIsIdeNewFun	975	LwpMenuMolSetshowlevel4	764
LwpMenuMIsIdeDebug	976	LwpMenuMolSetshowlevel5	765
LwpMenuMIsIdeRedo	977	LwpMenuMolSetshowlevel6	766
LwpMenuMIsShowide	978	LwpMenuMolSetshowlevel7	767
LwpMenuMIsIdeImport	979	LwpMenuMolSetshowlevel8	768
LwpMenuMIsIdeExport	980	LwpMenuMolSetshowlevel9	769
LwpMenuMIsIdeScriptPref	981	LwpMenuMolSetshowlevel10	770
LwpMenuMIsIdeCloseWin	982	LwpMenuMolOutstylesequence	749
LwpMenuMIsIdeToggle	983	LwpMenuMtTabkey	800
LwpMenuMIsIdePrevSub	984	LwpMenuMeDefpathsnw	804
LwpMenuMIsIdeNextSub	985	LwpMenuMvHidepicts	805
LwpMenuMIsIdeTogglebp	986	LwpMenuMvHidemarks	806
LwpMenuMIsIdeCodeWindow	987	LwpMenuMvHidetrack	807
LwpMenuMIsIdeCheckAll	988	LwpMenuMvColguide	808
LwpMenuMIsIdeSaveLso	989	LwpMenuMvShowcleanscreen	840
LwpMenuMIsIdeShowicon	992	LwpMenuMvFixediconbar	842
LwpMenuMIsIdeRuncursub	993	LwpMenuMvViewlevel	843

LwpMenuMlsideHelpLs	994	LwpMenuMtbChange	810
LwpMenuMlsideHelpWp	995	LwpMenuMtbTabbox	811
LwpMenuMlsideHelpSe	996	LwpMenuMvVertruler	814
LwpMenuMlsideHelpAbout	997	LwpMenuMvShownotes	816
LwpMenuMlsideInsertLso	998	LwpMenuMvShowtabsandreturns	818
LwpMenuMlRenunotes	149	LwpMenuMvOut2lay	820
LwpMenuEditmenu	150	LwpMenuMfImportdraw	821
LwpMenuMeUndo	151	LwpMenuMlMacroendrecord	822
LwpMenuMeCut	153	LwpMenuMvLay2out	823
LwpMenuMeCopy	154	LwpMenuMvSetpreferences	824
LwpMenuMePaste	155	LwpMenuMrCreatelastframe	825
LwpMenuMeUndoredo	156	LwpMenuMlQuickendrecord	826
LwpMenuMePastelinks	157	LwpMenuMrUngroupframes	827
LwpMenuMeInsvariable	158	LwpMenuMlRevmarkingoff	828
LwpMenuMeFootnotes	160	LwpMenuMlFieldhide	829
LwpMenuMeNotes	161	LwpMenuMcExtracton	838
LwpMenuMeNotes20	282	LwpMenuMdivDivisionproperties	844
LwpMenuMeDefaults	162	LwpMenuMdivNewdivision	845
LwpMenuMeTypeover	163	LwpMenuMdivCombinedivisions	846
LwpMenuMeNbDefaults	164	LwpMenuMdivDeletedivision	847
LwpMenuMeInsertdate	165	LwpMenuMdivExpanddivision	848
LwpMenuMeAddmergevar	166	LwpMenuMdivGroupdivision	854
LwpMenuMeAdddocdescvar	167	LwpMenuMsecSectionproperties	849
LwpMenuMxMdiswitch	168	LwpMenuMsecNewsection	850
LwpMenuMeSeclevaction	169	LwpMenuMsecCombinesections	851
LwpMenuMeDefpaths	170	LwpMenuMsecSelectsection	852
LwpMenuMeFootopts	171	LwpMenuMsecExpandsection	853
LwpMenuMeRedo	172	LwpMenuMtabSettabs	855
LwpMenuMeDocoptions	173	LwpMenuMtabCleartabs	856
LwpMenuMeDefpath	174	LwpMenuMtabLefttab	857
LwpMenuMeDefsty	175	LwpMenuMtabRighttab	858
LwpMenuMeDefback	176	LwpMenuMtabCentertab	859
LwpMenuMeFootnoteinsert	177	LwpMenuMtabNumerictab	862
LwpMenuMeInsertnote	178	LwpMenuMtabDefaulttabs	865
LwpMenuMeDelete	179	LwpMenuMtabRemoveruler	866
LwpMenuMeIndexentry	180	LwpMenuMtabHideruler	867
LwpMenuMeAddnote	181	LwpMenuMtabQuickright	990
LwpMenuMeMergevar	182	LwpMenuMtabQuickcenter	991
LwpMenuMeDescvar	183	LwpMenuMrSetmarginsinfobox	870
LwpMenuMeFormatcheckbox	184	LwpMenuMfToggleinfobox	871
LwpMenuMePageup	185	LwpMenuMfCreateversion	872
LwpMenuMePagedown	186	LwpMenuMfCreateversionDivision	905
LwpMenuMeRevinsertion	187	LwpMenuMvViewversion	880
LwpMenuMeProtected	188	LwpMenuMafidHelp	900
LwpMenuMeNohyphen	189	LwpMenuMafidNotfound	901

LwpMenuMeGetglosname	198	LwpMenuMfMailNextstop	902
LwpMenuMeMacpath	286	LwpMenuMfMailEditroute	903
LwpMenuMeTocentry	835	LwpMenuMergeScriptNotify	910
LwpMenuMeBullet	836	LwpMenuFirstpopupid	925
LwpMenuMeMergedatafile	562	LwpMenuRenderformat	860
LwpMenuMeMergeextdesc	564	LwpMenuRenderallformats	861
LwpMenuMeMergevarsdatafile	566	LwpMenuMcCommenttoolsOn	889
LwpMenuMePastespecial	588	LwpMenuMnDeletenote	890
LwpMenuMeInsertobject	589	LwpMenuMnClosenote	891
LwpMenuMeChangelink	590	LwpMenuMnNextnote	892
LwpMenuMeUseworkpath	591	LwpMenuMnPrevnote	893
LwpMenuMeSelectword	595	LwpMenuMcCommenttools	894
LwpMenuMeSelectsentence	596	LwpMenuMcHighnote	895
LwpMenuMeSelectparagraph	597	LwpMenuMnCloseall	896
LwpMenuMeSelectfile	598	LwpMenuMnOpenall	897
LwpMenuMeRevisionmode	599	LwpMenuMnDeleteall	898
LwpMenuMIDoccontrol	600	LwpMenuMnShowinitials	899
LwpMenuMfSharedoccontrol	602	LwpMenuMnNumbernotes	900
LwpMenuMIDoccontrolEditorpanel	603	LwpMenuMfQuickprint	946
LwpMenuMIDoccontrolExternal	604	LwpMenuMdDebugger	1000
LwpMenuMvViewversionExternal	605	LwpMenuMdDebugDoc	1001
LwpMenuMfDocdescExternal	606	LwpMenuMdDebugPres	1002
LwpMenuMfMrglabel	607	LwpMenuMdDebugDivinfo	1003
LwpMenuMfMrglettersetup	608	LwpMenuMdDebugCont	1004
LwpMenuMfMrgenvelopesetup	609	LwpMenuMdDebugCol	1005
LwpMenuMfMrgviewprint	612	LwpMenuMdDebugLine	1006
LwpMenuNwobjects	190	LwpMenuMdDebugVobj	1007
LwpMenuMoOpennw	191	LwpMenuMdDebugHeap	1008
LwpMenuMoCreatenw	192	LwpMenuMdDebugMain	1009
LwpMenuMoListnw	193	LwpMenuMdDebugProf	1010
LwpMenuMoLinksnw	194	LwpMenuMdDebugVcnt	1011
LwpMenuMoAttribnw	195	LwpMenuMdDebugDivision	1012
LwpMenuViewmenu	200	LwpMenuMdDebugLay	1013
LwpMenuMvFit	201	LwpMenuMdDebugVermgr	1014
LwpMenuMvWorking	202	LwpMenuMdDebugVerctrl	1015
LwpMenuMv75	796	LwpMenuMdDebugDoclay	1016
LwpMenuMv150	797	LwpMenuMdDebugPara	1018
LwpMenuMv100	203	LwpMenuMdDebugInfo	1019
LwpMenuMv200	204	LwpMenuMdDebugAtom	1020
LwpMenuMvSwitchiconstate	205	LwpMenuMdDebugWrdarg	1021
LwpMenuMvPagesort	771	LwpMenuMdDebugDwrarg	1022
LwpMenuMvZoomoptions	772	LwpMenuMdDebugStrarg	1023
LwpMenuMvComment	774	LwpMenuMdDebugSline	1024
LwpMenuMvShowheadfoot	775	LwpMenuMdDebugPhint	1025
LwpMenuMvCustom	773	LwpMenuMdDebugChint	1026

LwpMenuMvFacingpages	776	LwpMenuMdDebugMsgtrace	1027
LwpMenuMvMultiplepages	777	LwpMenuMdDebugFoundry	1028
LwpMenuMvPagemargin	778	LwpMenuMdDebugCounts	1029
LwpMenuMvPagewidth	779	LwpMenuMdDebugPresinfo	1030
LwpMenuMvSpecialview	803	LwpMenuMdDebugMeter	1031
LwpMenuMvClearspecialview	881	LwpMenuMdDebugStack	1032
LwpMenuMvSynchronizeviews	868	LwpMenuMdDebugPtrholder	1033
LwpMenuMvClearhighlight	874	LwpMenuMdDebugMemory	1034
LwpMenuMvPreferences	206	LwpMenuMdDebugMemorydump	1035
LwpMenuMvTabrack	207	LwpMenuMdDebugContent	1036
LwpMenuMvFunctionbar	208	LwpMenuMdDebugSilver	1037
LwpMenuMvChgsidebar	209	LwpMenuMdDebugVernode	1038
LwpMenuMvLayout	210	LwpMenuMdDebugCheckpoint	1039
LwpMenuMvDraft	211	LwpMenuMdDebugSocket	1040
LwpMenuMvFacing	212	LwpMenuMdDebugSfb	1041
LwpMenuMvFacecancel	214	LwpMenuMdDebugDemandload	1042
LwpMenuMvShowpicts	215	LwpMenuMdDebugNumberhint	1043
LwpMenuMvShowmarks	216	LwpMenuMdDebugDialog	1044
LwpMenuMvPreferdefaults	217	LwpMenuMdDebugBack	1045
LwpMenuMvOutline	218	LwpMenuMdDebugHeadtailholder	1046
LwpMenuMvFloatingicon	219	LwpMenuMdDebugNotifypersistent	1047
LwpMenuMvLefticon	220	LwpMenuMdDebugObjectholder	1048
LwpMenuMvRighticon	221	LwpMenuMdDebugEnumobjhead	1049
LwpMenuMvTopicon	222	LwpMenuMdDebugNotify	1050
LwpMenuMvBottomicon	223	LwpMenuMdDebugNamedobjectholder	1051
LwpMenuMvSmarticons	232	LwpMenuMdDebugSection	1052
LwpMenuMvFullscreen	233	LwpMenuMdDebugFontmgr	1053
LwpMenuMvFullscreenoptions	234	LwpMenuMdDebugSanity	1054
LwpMenuMvEditicon	236	LwpMenuMdDebugAtomWatch	1055
LwpMenuMfPrintenvelope	237	LwpMenuMdDebugAtomCount	1056
LwpMenuMvNexticonset	239	LwpMenuMdDebugPieceCount	1057
LwpMenuMvSaveicon	240	LwpMenuMdDebugObjectSizes	1058
LwpMenuMvCreateicon	241	LwpMenuMdDebugTextstyle	1059
LwpMenuMvFullscreenstartup	242	LwpMenuMdDebugEnd	1099
LwpMenuMvSectiontabs	244	LwpMenuMkPageup	1100
LwpMenuCharmenu	250	LwpMenuMkPagedown	1101
LwpMenuMcTextinfo	283	LwpMenuMcDooledragdrop	1102
LwpMenuMcTextinfo	610	LwpMenuMsMplMargins	1300
LwpMenuMcDrawinfo	611	LwpMenuMsMplPage	1301
LwpMenuMcFont	251	LwpMenuMsMplFooter	1302
LwpMenuMcNormal	253	LwpMenuMsMplHeader	1303
LwpMenuMcBold	254	LwpMenuMsMplLines	1304
LwpMenuMcItalic	255	LwpMenuMrMfiType	1400
LwpMenuMcUnderline	256	LwpMenuMrMfiSizepos	1401
LwpMenuMcWordunder	257	LwpMenuMrMfiLines	1402

LwpMenuMcLeft	259	LwpMenuMrMfIColumnstabs	1403
LwpMenuMcCenter	260	LwpMenuMISetmacrofile	1500
LwpMenuMcRight	261	LwpMenuMICompilemacro	1501
LwpMenuMcJustify	262	LwpMenuMIMacrorun	1502
LwpMenuMcAlignment	263	LwpMenuMIMacroappend	1503
LwpMenuMcCaps	264	LwpMenuMIPrototype	1504
LwpMenuMcIndentall	247	LwpMenuMeSmartcorrect	1505
LwpMenuMcIndentfirst	248	LwpMenuMrChart	1506
LwpMenuMcIndentrest	249	LwpMenuMrCharttodraw	1507
LwpMenuMcExtract	246	LwpMenuMmCascades	1600
LwpMenuMcMacrofastformat	238	LwpMenuSubmenuInsert	1600
LwpMenuMcFontrevert	243	LwpMenuMmFields	1601
LwpMenuMcEnargetext	252	LwpMenuSubmenuMarktextas	1602
LwpMenuMcReducetext	245	LwpMenuSubmenuAlignment	1603
LwpMenuMcIndent	258	LwpMenuMmCaps	1604
LwpMenuMcOutdent	284	LwpMenuMmMacros	1607
LwpMenuMcBulletpanel	285	LwpMenuSubmenuInsertsmartsum	1609
LwpMenuMcNextfacename	950	LwpMenuMMailmemo	1610
LwpMenuMcNextalignment	951	LwpMenuMPrinttofile	1611
LwpMenuMcNextattribute	952	LwpMenuSubmenuUsersetup	1612
LwpMenuMcNextbullet	953	LwpMenuSubmenuScripts	1613
LwpMenuMcNextnumbering	954	LwpMenuMmShowoptions	1614
LwpMenuMcNextstyle	955	LwpMenuSubmenuQuickdemo	1615
LwpMenuMcDoubleunder	270	LwpMenuSubmenuOtherhelpstuff	1616
LwpMenuMcSuperscript	271	LwpMenuMmOtherframe	1617
LwpMenuMcSubscript	272	LwpMenuMmOthertable	1618
LwpMenuMcAllcaps	273	LwpMenuSubmenuOthertextstuff	1619
LwpMenuMfEnvelopeprint	274	LwpMenuSubmenuPriority	1620
LwpMenuMcAlllower	287	LwpMenuMmOtherstuff	1621
LwpMenuMcInitcaps	288	LwpMenuSubmenuSelect	1622
LwpMenuMcSmallcaps	289	LwpMenuSubmenuOtherdocpart	1623
LwpMenuMcJustfont	278	LwpMenuSubmenuAttributes	1624
LwpMenuMcJustpointsize	279	LwpMenuSubmenuOutline	1625
LwpMenuMcStrikethru	280	LwpMenuSubmenuCollapse	1626
LwpMenuMcFastformat	291	LwpMenuSubmenuEdit	1627
LwpMenuMcHighlight	294	LwpMenuSubmenuDebug	1628
LwpMenuStylemenu	300	LwpMenuSubmenuDocprops	1629
LwpMenuMsCreate	301	LwpMenuSubmenuZoomto	1630
LwpMenuMsStylemanage	303	LwpMenuSubmenuShowhide	1631
LwpMenuMsChgstyle	304	LwpMenuSubmenuOtherviewstuff	1632
LwpMenuMsSavestyle	305	LwpMenuSubmenuNamedstyles	1633
LwpMenuMsSelectstyle	306	LwpMenuSubmenuDelete	1634
LwpMenuMsOutlinestyle	320	LwpMenuSubmenuFile	1635
LwpMenuMsStylemanageinit	321	LwpMenuSubmenuView	1636
LwpMenuMsStylemanageaction	322	LwpMenuSubmenuTeammail	1637



LwpMenuMsChangestylename	323	LwpMenuSubmenuEditmodemenu	1638
LwpMenuMsDefinestyle	324	LwpMenuSubmenuDebugmodemenu	1639
LwpMenuMsModcreatestyle	325	LwpMenuSubmenuHelp	1640
LwpMenuMsStylemanagefinish	326	LwpMenuCreatemenu	1650
LwpMenuMsRedefinestyle	336	LwpMenuParallelstylemenu	1658
LwpMenuMsMargins	328	LwpMenuEnvAddress	1661
LwpMenuMsPagesettings	329	LwpMenuEnvSave	1665
LwpMenuMsApply	330	LwpMenuEnvPrint	1666
LwpMenuMsStylemru1	331	LwpMenuMfCreatetabdivision	1670
LwpMenuMsStylemru2	332	LwpMenuMfCreatedivision	1667
LwpMenuMsStylemru3	333	LwpMenuMeTocassist	1668
LwpMenuMsStylemru4	334	LwpMenuMpNewsection	1669
LwpMenuMrManuatable	335	LwpMenuMtInsertclickhere	1671
LwpMenuMrFrameanchorinfolbox	391	LwpMenuMcTextlinepanel	1672
LwpMenuMrFramelineinfolbox	392	LwpMenuMvOtherviewstuff	1673
LwpMenuMrFramebackinfolbox	393	LwpMenuMoApplystyle	1674
LwpMenuMrFramegfxinfolbox	394	LwpMenuMsFramecreate	1675
LwpMenuMrFrametopalign	395	LwpMenuMsFrameredefinestyle	1676
LwpMenuMrFramecenteralign	396	LwpMenuMsTablecreate	1677
LwpMenuMrFramebottomalign	397	LwpMenuMsTableedefinestyle	1678
LwpMenuMrDeleteframe	399	LwpMenuMsTablecellcreate	1679
LwpMenuPagemenu	350	LwpMenuMsTablecelledefinestyle	1680
LwpMenuMpCreatelayout	351	LwpMenuMvShowmisspelled	1681
LwpMenuMpModifylayout	352	LwpMenuMtDelrowacc	1682
LwpMenuMpTabs	355	LwpMenuMtCtrlretacc	1683
LwpMenuMpPagemenu	356	LwpMenuMsQuickedefinestyle	1684
LwpMenuMpPagebreak	358	LwpMenuMtInsertrowacc	1685
LwpMenuMpLinenummer	359	LwpMenuMrOleconvert	1686
LwpMenuMpHeadfoot	360	LwpMenuMrOleconvertandactivate	1687
LwpMenuMpModlayoutinit	361	LwpMenuMrFrameleft	1688
LwpMenuMpModlaypagesettings	362	LwpMenuMrFramecenter	1689
LwpMenuMpModlayrightfacepage	363	LwpMenuMrFrameright	1690
LwpMenuMpModlayrightheadfacepage	364	LwpMenuMrFramehorzspan	1691
LwpMenuMpModlayrightfooterfacepage	365	LwpMenuMrFramevertspan	1692
LwpMenuMpModlayrightlines	366	LwpMenuMfNewanytab	1693
LwpMenuMpModlayleftfacepage	367	LwpMenuMfOdmaopen	1694
LwpMenuMpModlayleftheaderfacepage	368	LwpMenuMfOdmainsert	1695
LwpMenuMpModlayleftfooterfacepage	369	LwpMenuMfOdmaconvert	1696
LwpMenuMpModlayleftlines	370	LwpMenuMfOdmaasaveas	1697
LwpMenuMpDellayout	373	LwpMenuMfOdmaexternalize	1698
LwpMenuMpRevertlayout	374	LwpMenuMfOdmaexport	1699
LwpMenuMpInslayout	375	LwpMenuMrOleverb	1700
LwpMenuMpModlayoutfinish	376	LwpMenuMrOleverbmax	1799
LwpMenuMpHeaderfootergoto	377	LwpMenuMrOledivtabverb	1800
LwpMenuMpMacinslayout	378	LwpMenuMrOledivtabverbmax	1899

LwpMenuMpParallel	379	LwpMenuMtSelectcell	1900
LwpMenuMpPcolinfox	380	LwpMenuMtPcolselectcell	1901
LwpMenuMpPageproperties	381	LwpMenuMpColumnbreak	1902
LwpMenuMpCreate	382	LwpMenuMeViewtablegrid	1903
LwpMenuMpNewpagestyle	383	LwpMenuMeViewtableheading	1904
LwpMenuMpHeaderproperties	384	LwpMenuMeViewpcolguides	1905
LwpMenuMpFooterproperties	385	LwpMenuMeViewmarginguides	1906
LwpMenuMpPagepropertiesmouse	386	LwpMenuMeViewpagegauge	1907
LwpMenuMpFooterpropertiesmouse	387	LwpMenuMeViewshowallmarks	1908
LwpMenuMpHeaderpropertiesmouse	388	LwpMenuMeViewhideallmarks	1909
LwpMenuMpPcolinfoxmouse	389	LwpMenuMtTableleft	1910
LwpMenuFramemenu	400	LwpMenuMtTablecenter	1911
LwpMenuMrFrametext	402	LwpMenuMtTableright	1912
LwpMenuMrScale	403	LwpMenuMtTablehorzspan	1913
LwpMenuMrGroupframes	404	LwpMenuMtGotoinfox	1914
LwpMenuMrFrameinfox	405	LwpMenuMhTour	1915
LwpMenuMrBringtofront	407	LwpMenuMfCreateoledivision	1917
LwpMenuMrSendtoback	408	LwpMenuMfToggleclickhereprompts	1918
LwpMenuMrAdd	409	LwpMenuMcToggleskipbullet	1919
LwpMenuMrSendbackone	410	LwpMenuMISkipmisspelledword	1920
LwpMenuMrDrawingicon	411	LwpMenuMISkipallmisspelledword	1921
LwpMenuMrChartingicon	412	LwpMenuMIAddtouserdictionary	1922
LwpMenuMrCreateframe	413	LwpMenuMITogglebubblehelp	1923
LwpMenuMrAddframewithdlg	414	LwpMenuMfPlainnew	1924
LwpMenuMrFramedefaults	415	LwpMenuMvToggleshowanchors	1925
LwpMenuMrFramelines	416	LwpMenuMvNextnewestversion	1926
LwpMenuMrFrametype	417	LwpMenuMvMostrecentversion	1927
LwpMenuMrFrameborders	418	LwpMenuMvOldestversion	1928
LwpMenuMrFramecolumns	419	LwpMenuMcJustifyall	1929
LwpMenuMrModframereflow	420	LwpMenuMcClosegraphicole	1930
LwpMenuMrModframeinit	421	LwpMenuMfFramerevert	1931
LwpMenuMrBringfrontone	390	LwpMenuMtCellrevert	1932
LwpMenuMrSelectframe	398	LwpMenuMfOpentabdivision	1933
LwpMenuMrEquationsicon	423	LwpMenuMfNewtabdivision	1934
LwpMenuMrImageprocessing	424	LwpMenuMtInsertclickherelink	1935
LwpMenuMrFrameinfoxmouse	620	LwpMenuMtEditclickherelink	1936
LwpMenuMrFramelineinfoxmouse	621	LwpMenuMsecDeletesection	1949
LwpMenuMrFramebackinfoxmouse	622	LwpMenuMfExternalfileexit	1950
LwpMenuMrFramegfxinfoxmouse	623	LwpMenuMfExternalfilesavedivision	1951
LwpMenuMrFrameanchorinfoxmouse	624	LwpMenuMfFtpopen	1952
LwpMenuMrExpert	911	LwpMenuMfFtpinsert	1953
LwpMenuMrExpertmax	924	LwpMenuMfFtpconvert	1954
LwpMenuMrWppack	349	LwpMenuMfFtpsaveas	1955
LwpMenuMrAfid	425	LwpMenuMfFtpexternalize	1956
LwpMenuMrAfidmax	449	LwpMenuSubmenuInternet	1957

LwpMenuToolmenu	450	LwpMenuMfFtpoptions	1958
LwpMenuMISearch	451	LwpMenuMfOdmashowattributes	1959
LwpMenuMIGoto	452	LwpMenuMrRuby	1960
LwpMenuMISpellopts	453	LwpMenuMfFtpopendivision	1961
LwpMenuMIUserdict	454	LwpMenuMfFtpopenurl	1962
LwpMenuMIThesaurus	455	LwpMenuIbShowfrommenu	1963
LwpMenuMISort	456	LwpMenuMoHideinfobox	1964
LwpMenuMITables	457	LwpMenuMfWelcome	1965
LwpMenuMIGotolast	458	LwpMenuMsQuickredefineframestyle	1969
LwpMenuMISpellopts2	459	LwpMenuMsQuickredefinetablestyle	1970
LwpMenuMIRevmarkoptions	461	LwpMenuMsQuickredefinecellstyle	1971
LwpMenuMIToc	462	LwpMenuMsQuickredefinestyle	1972
LwpMenuMIAssignmacro	463	LwpMenuMeInsertsymbol	1973
LwpMenuMIMacros	464	LwpMenuMeWordcount	1974
LwpMenuMIMacroaccel	422	LwpMenuMeDropcaps	1975
LwpMenuMIGeneratetoc	465	LwpMenuMfOdmimportpicture	1976
LwpMenuMIIndexnextfile	281	LwpMenuMrOleobjectinfobox	1977
LwpMenuMIBookmark	466	LwpMenuMcInternettoolson	1978
LwpMenuMIGlossary	467	LwpMenuMcInternettools	1979
LwpMenuMIGlosset	468	LwpMenuMhLotushomepage	1980
LwpMenuMIMacpause	469	LwpMenuMhLotussupport	1981
LwpMenuMIGenerateindex	470	LwpMenuMhLotusftpsite	1982
LwpMenuMISpelladddict	471	LwpMenuMrLinkframes	1983
LwpMenuMISpellskipall	472	LwpMenuMrUnlinkframes	1984
LwpMenuMISpellreplace	474	LwpMenuMfMrgwelcome	1985
LwpMenuMISpell	475	LwpMenuMfHtmloptions	1986
LwpMenuMISpellinit	476	LwpMenuMcUpdateindex	1987
LwpMenuMISpellcheck	477	LwpMenuMcUpdatetoc	1988
LwpMenuMISpellcancel	478	LwpMenuMvDesignmode	1989
LwpMenuMISpellskip	479	LwpMenuMfNosopen	1990
LwpMenuMIReplace	480	LwpMenuMfNossaveas	1991
LwpMenuMIChglang	481	LwpMenuMfMailSendmessage	1992
LwpMenuMIRenumseq	482	LwpMenuMcInsertdefnumber	1993
LwpMenuMISRattrs	485	LwpMenuMcInsertdefbullet	1994
LwpMenuMIRepopts	486	LwpMenuMeTogglesmartselect	1995
LwpMenuMISropts	487	LwpMenuMcCreateobjectviewer	1996
LwpMenuMISearchonly	488	LwpMenuMcCreateobjectcomment	1997
LwpMenuMIMacplay	489	LwpMenuMcCreateobjectdraw	1998
LwpMenuMIMacassign	490	LwpMenuMcCreateobjectschedule	1999
LwpMenuMIMacdde	491	LwpMenuMcCreateobjectspreadsheet	2000
LwpMenuMIMactype	492	LwpMenuMcCreateobjectchart	2001
LwpMenuMICall	493	LwpMenuMhInternetsearch	2002
LwpMenuMISammy2	494	LwpMenuMhLotussmartsuite	2003
LwpMenuMITocopts	495	LwpMenuMcClickherekeyword	2004
LwpMenuMISrfindmatch	496	LwpMenuMfPublishweb	2005

LwpMenuMISrreplacethenfind	497	LwpMenuMICrossreference	2006
LwpMenuMISrcancel	498	LwpMenuMtabHidevertruler	2007
LwpMenuMISrinit	499	LwpMenuMpUpdateheaders	2008
LwpMenuMIDoccompare	500	LwpMenuMpUpdatefooters	2009
LwpMenuMIMacroedit	501	LwpMenuMtEditclickhere	2010
LwpMenuMIMacroplay	502	LwpMenuMfShowwebauthor	2011
LwpMenuMIMacrorecord	503	LwpMenuMfHidewebauthor	2012
LwpMenuMIMacrooptions	504	LwpMenuMcWebauthorbegin	2013
LwpMenuMIQuickrec	505	LwpMenuMcWebcreatelink	2013
LwpMenuMIQuickplay	506	LwpMenuMcWebcreatehr	2014
LwpMenuMIGrammar	507	LwpMenuMcWebcreatewallpaper	2015
LwpMenuMIGrammaropts	508	LwpMenuMcWebcreateformcontrol	2016
LwpMenuMIGrammarstart	509	LwpMenuMcWebauthorend	2024
LwpMenuMIGrammarinit	510	LwpMenuMrRubyAbove	3000
LwpMenuMIModcreaterule	511	LwpMenuMrRubyBelow	3001
LwpMenuMIGrammarnext	512	LwpMenuMrRubyDelete	3002
LwpMenuMIGrammarcancel	513	LwpMenuMxMinmacromenu	9000
LwpMenuMIGrammarsugtext	514	LwpMenuMxMaxmacromenu	9099
LwpMenuMIGrammaroptionstext	515	LwpMenuMrAfidmenu	10000
LwpMenuMIGrammarchange	516	LwpMenuMrAfidmenumax	16000
LwpMenuMIGrammarstats	517	LwpMenuMaNotesflow	16001
LwpMenuMIIconpath	518	LwpMenuMaNotesflowmax	16200
LwpMenuMICyclekeysetup	519	LwpMenuFcsTextmenu	0
LwpMenuMISmartfill	520	LwpMenuFcsFramemenu	1
LwpMenuMINewspell	521	LwpMenuFcsFrametextmenu	2
LwpMenuMIRevisionbar	522	LwpMenuFcsTabletextmenu	3
LwpMenuMIToa	523	LwpMenuFcsFramegraphicmenu	4
LwpMenuWindowmenu	4	LwpMenuFcsPowerfieldmenu	5
LwpMenuMwNewwindow	525	LwpMenuFcsParallelcoltextmenu	6
LwpMenuMwTilewindow	526	LwpMenuFcsDivisionmenu	7
LwpMenuMwCascadewindow	527	LwpMenuFcsSectionmenu	8
LwpMenuMwStartoffiles	528	LwpMenuFcsTablemenu	9
LwpMenuMwTilewindowhorz	582	LwpMenuFcsParallelcolmenu	10
LwpMenuMwSplitvertwindow	593	LwpMenuFcsTablegraphicmenu	11
LwpMenuMwSplithorzwindow	594	LwpMenuFcsParallelcolgraphicmenu	12
LwpMenuMwEndoffiles	548	LwpMenuFcsHeadermenu	13
LwpMenuMwFilesseparator	549	LwpMenuFcsFootermenu	14
LwpMenuMIMacresume	554	LwpMenuFcsRulermenu	15
LwpMenuMIReviewrevs	557	LwpMenuFcsNotemenu	16
LwpMenuMIRevacceptall	875	LwpMenuFcsOutlinemenu	17
LwpMenuMIRevcancelall	876	LwpMenuFcsOledivisionmenu	18
LwpMenuMIRevaccept	877	LwpMenuFcsFramechartmenu	19
LwpMenuMIRevcancel	878	LwpMenuFcsTablechartmenu	20
LwpMenuMIField	568	LwpMenuFcsParallelcolchartmenu	21
LwpMenuMIFieldnext	569	LwpMenuFcsTablecornermenu	22

LwpMenuMIFieldprev	570	LwpMenuFcsTablerowmenu	23
LwpMenuMIFieldeval	571	LwpMenuFcsTablecolumnmenu	24
LwpMenuMIUpdatedate	575	LwpMenuFcsVertrulermenu	25
LwpMenuMIUpdatealldates	576	LwpMenuFcsRubymenu	26
LwpMenuMIFieldadd	572	LwpMenuFcsFootnotetextmenu	40
LwpMenuMIFieldupdate	573	LwpMenuFcsIndextextmenu	41
LwpMenuMIFieldtog	266	LwpMenuFcsToctextmenu	42
LwpMenuMIFieldupall	267	LwpMenuFcsTablecornermenutoc	43
LwpMenuMIFieldlock	268	LwpMenuFcsTabletextmenutoc	44
LwpMenuMIFieldsave	269	LwpMenuFcsFrametextmenudropcap	45
LwpMenuMIFieldauto	224	LwpMenuFcsFramemenudropcap	46
LwpMenuMIFieldshowrt	225	LwpMenuFcsTextdatemenu	47

### Return value

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### Usage

### **Word Pro: WordUnderline method**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_WORDUNDERLINE\_METHOD\_EXSCRIPT',1)} [See example](#)

Sets the WordUnderline attribute for selected text, or all following text if no text is selected. It acts as a toggle, turning the attribute off if it is on, and on if it is off. Equivalent to choosing Text - Attributes - Other, and then choosing "Word Underline" from the Attributes box.

### **Syntax**

[objectreference].WordUnderline()

### **Parameters**

None

### **Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

### **Usage**

## Word Pro: WriteProfileString method

{button ,AL('H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WRITEPROFILESTRING\_METHOD\_EXSCRIPT',1)} [See example](#)

Writes a new profile string in the specified INI file or INI entry. You can write a profile string in any of the standard Word Pro INI files or you can specify another INI file.

### Syntax

[objectreference].WriteProfileString(Section, Key, NewString[, IniFileType][, WhichIniLocation][, IniName])

### Parameters

#### *Section*

A String expression which specifies a name of a section in the INI. Word Pro will search only the section you name in this parameter. If the named section does not match a section in the specified INI, this method fails. If you use an empty string ("") Word Pro assumes you are writing to the LWPUSER.INI file (IniFileType parameter = "\$LwpIniUserPrefs") and looks for the "WordProUser" section. Most INIs have more than one section. The section name you provide in this parameter must match the section name in the INI exactly.

#### *Key*

A String expression which specifies the key name in the section you are writing.

#### *NewString*

A String expression which Word Pro writes as the value for the key.

#### *IniFileType*

Specifies the INI in which you want to write the new profile string. You can choose one of the standard Word Pro INI files or choose "\$LwpIniCustomFile" to search another INI file. The data type for this parameter is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). You do not have to provide a value for this parameter. The default value for this parameter is "\$LwpIniUserPrefs".

\$LwpIniUserPrefs (2101)

The default value for this parameter. This is the INI file used to store Word Pro's user preference information (lwpuser.ini).

\$LwpIniConfigPrefs (2102)

The INI file used to store Word Pro's configuration preference information.

\$LwpIniEnvelopeAndMerge (2105)

The INI file used to store Word Pro's envelope and merge information.

\$LwpIniLanguages (2107)

The INI file used to store some of Word Pro's language information.

\$LwpIniSharedLotusInfo (2103)

The INI file used to store shared information between Word Pro and other Lotus products.

\$LwpIniSmartcorrect (2106)

The INI file used to store Word Pro's SmartCorrect information.

\$LwpIniSmartfill (2104)

The INI file used to store Word Pro's SmartFill lists.

\$LwpIniCustomfile (2100)

Allows you to write a profile string in an INI file which is not one of the standard Word Pro INI files. If you use this value, you must use the IniName parameter to specify the name of the INI file (Windows 3.1 or OS/2) or INI entry (Windows 95) in which the profile string is located.

#### *WhichIniLocation*

Tells Word Pro whether to look on the network or the local machine for the specified INI file. The data type for this parameter is Variant which allows the value of this parameter to be one of the constants listed below or its numeric equivalent (in parentheses). You do not have to provide a value for this parameter. The default value for this parameter is "\$LwpUserIniLocation".

\$LwpNetworkIniLocation (2171)

Searches directory for network INI files.

\$LwpUserIniLocation (2172)

Searches directory for user INI files.

**Note** For Windows 95, in the registry, the user location is HKEY\_CURRENT\_USER or HKEY\_USERS. The network

location is HKEY\_LOCAL\_MACHINE. Within either of these locations, the path below this would be: Software\Lotus\WordPro\96.0.

*IniName*

An optional String expression that identifies the INI to which you want to write. Use this parameter only if you used "\$LwpIniCustomFile" as the value of the IniFileType parameter. This INI must be stored in the same directory as the standard Word Pro INIs.

**Note** If you are using Windows 3.1 or OS/2, this value is an INI file name. If you are using Windows 95, this value is an INI entry as seen in the Windows Registry application (REGEDIT.EXE).

**Return value**

This method returns a value of -1 (True) or 0 (False) indicating that the method succeeded or failed respectively.

**Usage**



## **Word Pro: Write method**

{button ,AL(^H\_BAG\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_WRITE\_METHOD\_EXSCRIPT',1)} [See example](#)

### **Syntax**

[objectreference].Write(Data, Length)

### **Parameters**

*Data*

Data type is String.

*Length*

Data type is Long.

### **Return value**

### **Usage**

'Example: SplitParagraph method

' This examples inserts two paragraphs of text into the current document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.InsertText "This is paragraph 1."

.Text.SplitParagraph

.Text.InsertText "This is paragraph 2."

'Example: SplitPercentage property  
'This example script has not yet been created.

'Example: Split method

' This example creates a table with one row and one column. The table cell

' is then split into 3 rows and three columns.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateTable False, "Default Table", 1,1

.Table.Split \$LwpTableSplitTypeCell, 3,3

'Example: SRReplace method

'This example script has not yet been created.

'Example: StartColumns property

'This example script has not yet been created.

```
'Example: StartEditMergeData method
' This example adds some records to merge data file for the current document.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
' The document must have a merge data file associated with it.

'Get the data file name for the current document
DataFile = .ActiveDocument.MergeOptions.DataFileName

'Opens the current data file invisibly for editing
.StartEditMergeData

'Add some records to the data file
.MergeAddDataRecord "John Doe~100 Main St.~ Atlanta~ GA~30319|"
.MergeAddDataRecord "Jane Doe~100 Main St.~ Atlanta~ GA~30319|"

'Save the data file
.Application.Documents.Item(DataFile).Save

'Close merge
.Merge $LwpMergeActionClose
```

```
'Example: StartEnvelopeDiv method
' This example moves the insertion point to the beginning of the current
' document and inserts sample address information. A new envelope division is
' created and an address prompt is inserted in the return address frame.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Text.MoveToStart $LwpLocationTypeDocument
.Type "FirstName LastName[Enter]"
.Type "Address[Enter]"
.Type "City, State Zip"

.StartEnvelopeDiv
.GoToLayout "Envelope:Envelope Return"
.Type "Return Address Goes Here[Enter]"
```



```
'Example: StartFieldInsert method
' This example creates a data file for the current document. Two records are
' added and the Merge bar is opened so to insert fields for merging.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.CreateDataFile "~|", "Name~Address~City~State~Zip|", False, "C:\mergedat.lwp"

'Add some records to the virtual datafile
.MergeAddDataRecord "Jane Doe~100 Main St.~ Atlanta~ GA~30319|"
.MergeAddDataRecord "John Doe~100 Main St.~ Atlanta~ GA~30319|"

.StartFieldInsert
```

'Example: StartingColOfSelection property  
'This example script has not yet been created.

'Example: StartingColStringOfSelection property  
'This example script has not yet been created.

'Example: StartingNumber property

'This example script has not yet been created.

'Example: StartingRowOfSelection property  
'This example script has not yet been created.

'Example: StartRow property

'This example script has not yet been created.

'Example: Start method

'This example script has not yet been created.

'Example: Start property

'This example script has not yet been created.



'Example: StateID property

'This example script has not yet been created.

'Example: StatusBarButtonClicked event  
'This example script has not yet been created.

'Example: StatusBarButtonFillPopupList event  
'This example script has not yet been created.

'Example: StatusBarButtonItemSelected event  
'This example script has not yet been created.

'Example: StatusBarButtonOverrideGraphic event  
'This example script has not yet been created.

'Example: StatusBarButtonOverrideTextAndGraphic event  
'This example script has not yet been created.

'Example: StatusBarButtonOverrideText event  
'This example script has not yet been created.

'Example: StatusBarButtons property  
'This example script has not yet been created.



'Example: StatusBarVisible property  
'This example script has not yet been created.

'Example: StatusBar property

'This example script has not yet been created.

'Example: StatusSpellReplaceAll property

'This example script has not yet been created.

'Example: StoreInternetFile method

'This example script has not yet been created.

'Example: StrikeThru method

' This example toggles the strikethru attribute for the selected text.

' RUNTIME DEPENDENCIES: You must have a document open with some text selected  
' for this script to work.

.StrikeThru

'Example: StyleExceptions property  
'This example script has not yet been created.

'Example: StyleName property

'This example script has not yet been created.

'Example: StylePaths property

'This example script has not yet been created.



'Example: StylePath property

'This example script has not yet been created.

'Example: StyleSheetFullPath property

'This example script has not yet been created.

'Example: StyleSheetName property

'This example script has not yet been created.

'Example: StyleSheetPath property

'This example script has not yet been created.

'Example: Style property

'This example script has not yet been created.

'Example: SubScript method

' This example toggles the subscript attribute for the selected text.

' RUNTIME DEPENDENCIES: You must have a document open with some text selected

.Subscript

'Example: Subscript property

'This example script has not yet been created.

'Example: Suffix property

'This example script has not yet been created.



'Example: Suggestions property

'This example script has not yet been created.

```
'Example: Sum method
' This example creates a table with 5 rows and 1 column. Rows 1 through 4
' are consecutively numbered and summed in row 5.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Dim RowCounter As Integer
.CreateTable False, "Default Table", 5,5
For RowCounter = 0 To MyTable.NumRows - 2
    .Type Format$(RowCounter)
    .Text.MoveDown 1
Next

.Table.Sum $LwpTableSumScopeColumn
```

'Example: SuperScript method

' This example toggles the superscript attribute for the selected text.

' RUNTIME DEPENDENCIES: You must have a document open with some text selected

.Superscript

'Example: Superscript property

'This example script has not yet been created.

'Example: SuperTableContainer property  
'This example script has not yet been created.

'Example: SuperTableLayouts property  
'This example script has not yet been created.

'Example: SuperTables property

'This example script has not yet been created.

'Example: SuppressHeaders property  
'This example script has not yet been created.



'Example: TabExits property

'This example script has not yet been created.

'Example: TableContainer property

'This example script has not yet been created.

'Example: TableExports property

'This example script has not yet been created.

'Example: TableFill property

'This example script has not yet been created.

'Example: TableHeadingLayouts property

'This example script has not yet been created.

'Example: TableHeadings property

'This example script has not yet been created.

'Example: TableImports property

'This example script has not yet been created.

'Example: TableLayouts property

'This example script has not yet been created.



'Example: TableLine property

'This example script has not yet been created.

'Example: TableMarkers property

'This example script has not yet been created.

'Example: TableOnlyContainer property  
'This example script has not yet been created.

'Example: TableStyleName property

'This example script has not yet been created.

'Example: TableStyles property

'This example script has not yet been created.

'Example: Tables property

'This example script has not yet been created.

'Example: Table property

'This example script has not yet been created.

'Example: TabOrder property

'This example script has not yet been created.



'Example: TabRack property

'This example script has not yet been created.

'Example: TabRelativeTo property

'This example script has not yet been created.

'Example: TabSpacing property

'This example script has not yet been created.

'Example: TabSymbolChar property

'This example script has not yet been created.

'Example: TabType property

'This example script has not yet been created.

'Example: TabWidth property

'This example script has not yet been created.

'Example: TeamMail method

' This example displays the Team Mail dialog box.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.TeamMail

'Example: TempFindAndReplace property

'This example script has not yet been created.



'Example: TempFoundry property

'This example script has not yet been created.

'Example: TerminateFormatCheck method  
'This example script has not yet been created.

'Example: TextandTableExports property  
'This example script has not yet been created.

'Example: TextandTableImports property  
'This example script has not yet been created.

'Example: TextAttributes property

'This example script has not yet been created.

'Example: TextMarkers property

'This example script has not yet been created.

'Example: TextNumber method

'This example script has not yet been created.

'Example: TextOrientation property  
'This example script has not yet been created.



'Example: TextOrient property

'This example script has not yet been created.

'Example: TextStyleName property

'This example script has not yet been created.

'Example: TextStyles property

'This example script has not yet been created.

'Example: Texts property

'This example script has not yet been created.

'Example: TextTightness property

'This example script has not yet been created.

'Example: TextViewAttributes property

'This example script has not yet been created.

'Example: Text property

'This example script has not yet been created.

'Example: TheoreticalScaledSize method  
'This example script has not yet been created.



'Example: TheMaxNumSynonymsReturned property  
'This example script has not yet been created.

```
'Example: TileWindowHorz method
' This example creates two new documents based on the 'DEFAULT.MWP'
' SmartMaster.
' The script then prompts you to tile the new windows horizontally.

.NewDocument , , "DEFAULT.MWP", ,
.NewDocument , , "DEFAULT.MWP", ,
MessageBox "Click OK to tile the new windows horizontally.", MB_OK, "Example Script"

.TileWindowHorz
```

```
'Example: TileWindowVert method
' This example creates two new documents based on the 'DEFAULT.MWP'
' SmartMaster.
' The script then prompts you to tile the new windows vertically.

.NewDocument , , "DEFAULT.MWP", ,
.NewDocument , , "DEFAULT.MWP", ,
MessageBox "Click OK to tile the new windows vertically.", MB_OK, "Example Script"

.TileWindowVert
```

```
'Example: TileWindow property
' This example creates two new documents based on the 'DEFAULT.MWP'
' SmartMaster.
' The script then prompts you to tile the new windows.

.NewDocument , , "DEFAULT.MWP", ,
.NewDocument , , "DEFAULT.MWP", ,
MessageBox "Click OK to tile the new windows.", MB_OK, "Example Script"

.TileWindow
```

'Example: Tile method

' This example tiles any open document windows.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.ApplicationWindow.Tile

'Example: TimedSaveFileExtension property  
'This example script has not yet been created.

'Example: TimedSave method

' This example performs a Timed Save of all the open documents which have been  
' saved at least once.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.TimedSave

'Example: Time property

'This example script has not yet been created.



'Example: TitleBarDocNumber property

'This example script has not yet been created.

'Example: TitleBarVisible property  
'This example script has not yet been created.

'Example: Title property

'This example script has not yet been created.

**Word Pro: Subscript property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUBSCRIPT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

subscriptvalue = [objectreference].Subscript

[objectreference].Subscript = subscriptvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: Suffix property**

{button ,AL('H\_NUMERICFORMATSUBSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUFFIX\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

suffixvalue = [objectreference].Suffix

[objectreference].Suffix = suffixvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: Suggestions property

{button ,AL('H\_EDITOR\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUGGESTIONS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The suggestions that appear for an editor when a document is opened.

### Data Type

The data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its hexadecimal equivalent (in parentheses). You can combine these constants when you want Word Pro to combine the features listed below. Use the OR operator to combine constants.

### Syntax

suggestionsvalue = [objectreference].Suggestions

### Legal values

<u>Value</u>	<u>Effect</u>
LwpEditSuggEditingInNewVersion (&H4)	Suggests that an editor create a new version to work in. This is only an option when edits are in the current or new version.
LwpEditSuggNoSuggestions (&H0)	No suggestions display.
LwpEditSuggOnlyMarkupEditsAllowed (&H1)	Suggests that edits appear as markups.
LwpEditSuggRevAndCommentIconbar (&H2)	Suggests that the editor use the Review & Comment Tools icon bar to insert comments.

### Usage

Use one of the values above to determine which suggestion displays when the editor opens the document.

## **Word Pro: Superscript property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUPERSCRIPT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

superscriptvalue = [objectreference].Superscript

[objectreference].Superscript = superscriptvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: SuppressHeaders property**

{button ,AL('H\_DIVISIONINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_SUPPRESSHEADERS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

suppressheadersvalue = [objectreference].SuppressHeaders

[objectreference].SuppressHeaders = suppressheadersvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



### **Word Pro: TabExits property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABEXITS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether pressing the Tab key while in a particular ClickHere object will exit the ClickHere or insert a normal tab mark.

### **Data Type**

[Integer](#)

### **Syntax**

tabexitsvalue = [objectreference].TabExits

[objectreference].TabExits = tabexitsvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: TableStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABLESTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[String](#)

### **Syntax**

tablestylevalue = [objectreference].TableStyleName

[objectreference].TableStyleName = tablestylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: TabOrder property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABORDER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The tab order for a ClickHere object.

### **Data Type**

Long

### **Syntax**

tabordervalue = [objectreference].TabOrder

[objectreference].TabOrder = tabordervalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: TabRelativeTo property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABRELATIVETO\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The tab position on the ruler.

### **Data Type**

Variant (enumerated)

### **Syntax**

tabrelativetovalue = [objectreference].TabRelativeTo

[objectreference].TabRelativeTo = tabrelativetovalue

### **Legal values**

\$LwpTabRelativeLeft = 1860

\$LwpTabRelativeRight = 1861

\$LwpTabRelativeCenter = 1862

### **Usage**

Equivalent to the "Tab position on ruler" field in the Set Tabs on Ruler dialog box. Although this box contains four choices, the legal values for this property only encompass three of those choices. The fourth choice, "Evenly spaced every," cannot be set as a value for this property

## **Word Pro: TabSpacing property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TABSPACING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

tabspacingvalue = [objectreference].TabSpacing

[objectreference].TabSpacing = tabspacingvalue

### **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

**Word Pro: TabSymbolChar property**

{button ,AL('H\_CHARACTERSET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABSYMBOLCHAR\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

tabsymbolcharvalue = [objectreference].TabSymbolChar

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: TabType property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The type of tab to set on the ruler.

**Data Type**

Variant (Enumerated)

**Syntax**

tabtypevalue = [objectreference].TabType

[objectreference].TabType = tabtypevalue

**Legal values**

\$LwpTabTypeCenter (1864)

\$LwpTabTypeLeft (1863)

\$LwpTabTypeNumeric (1866)

\$LwpTabTypeRight (1865)

**Usage**

Equivalent to the "Tab type" field in the Set Tabs on Ruler dialog box.

## **Word Pro: TabWidth property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TABWIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

tabwidthvalue = [objectreference].TabWidth

[objectreference].TabWidth = tabwidthvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



## Word Pro: TextOrientation property

{button ,AL('H\_BASECONTAINER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_FR  
AMECONTAINER\_CLASS;H\_NOTECONTAINER\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PARALLELCO  
LSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_SUBPAGECONTAINER\_CLA  
SS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_TABLECONTAINER\_CLAS  
S;H\_TABLEONLYCONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTORIENTATION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Returns the orientation of text in a container.

## Data Type

Data type for this property is [Integer](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

textorientationvalue = [objectreference].TextOrientation

## Legal values

The legal values for this property are listed below:

<u>Value</u>	<u>Effect</u>
0	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientLefttorightToptobottom
1	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientToptobottomRighttoleft
2	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientRighttoleftBottomtotop
3	Indicates that the TextOrient property of the Layout object is set to \$LwpTextOrientBottomtotopLefttoright

## Usage

## Word Pro: TextOrient property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_TEXTORIENT_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) The orientation of text in a layout.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

textorientvalue = [objectreference].TextOrient

[objectreference].TextOrient = textorientvalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpTextOrientBottomtotopLefttoright (1913)	Setting this value starts text in a layout object from the bottom to the top and then flows the text from left to right.
\$LwpTextOrientBottomtotopRighttoleft (1917)	Setting this value starts text in a layout object from the top to the bottom and then flows the text from right to left.
\$LwpTextOrientLefttorightBottomtotop (1916)	Setting this value starts text in a layout object from the left to the right and then flows the text from bottom to top.
\$LwpTextOrientLefttorightToptobottom (1910)	Setting this value starts text in a layout object from the left to the right and then flows the text from top to bottom.
\$LwpTextOrientRighttoleftBottomtotop (1912)	Setting this value starts text in a layout object from the right to the left and then flows the text from bottom to top.
\$LwpTextOrientRighttoleftToptobottom (1914)	Setting this value starts text in a layout object from the right to the left and then flows the text from top to bottom.
\$LwpTextOrientToptobottomLefttoright (1915)	Setting this value starts text in a layout object from the top to the bottom and then flows the text from left to right.
\$LwpTextOrientToptobottomRighttoleft (1911)	Setting this value starts text in a layout object from the top to the bottom and then flows the text from right to left.

## Usage

Equivalent to the "Text direction" setting, located on the Misc panel of the InfoBox for certain layout objects.

## **Word Pro: TextStyleName property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTSTYLENAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

textstylevalue = [objectreference].TextStyleName

[objectreference].TextStyleName = textstylevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: TextTightness property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTTIGHTNESS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

texttightnessvalue = [objectreference].TextTightness

[objectreference].TextTightness = texttightnessvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: TextViewAttributes property**

{button ,AL('H\_CLICKHERE\_CLASS;H\_TEXT\_CLASS;H\_TEXTMARKER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TEXTVIEWATTRIBUTES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates which of three special attributes has been set for viewing the current text stream.

**Data Type**

Integer (Enumerated Bitmask)

**Syntax**

textviewattributesvalue = [objectreference].TextViewAttributes

[objectreference].TextViewAttributes = textviewattributesvalue

**Legal values**

LwpTextViewCaretVisible (&H1) Sets the cursor blinking at the current text stream.

LwpTextViewHiddenStoryptr (&H4) Keeps the window from scrolling when the cursor moves off the screen.

LwpTextViewSelectionVisible (&H2) The shading is not visible.

**Usage**

**Word Pro: ThesMaxNumSynonymsReturned property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_THESMAXNUMSYNONYMSRETURNED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The maximum number of synonyms returned by the Thesaurus.

**Data Type**

[Integer](#)

**Syntax**

thesmaxnumsynonymsreturnedvalue = [objectreference].ThesMaxNumSynonymsReturned

[objectreference].ThesMaxNumSynonymsReturned = thesmaxnumsynonymsreturnedvalue

**Legal values**

Default is 0.

**Usage**

The Thesaurus checks the LOTUS.INI file for the setting for maximum number of synonyms returned. If there is no setting in the LOTUS.INI file, Word Pro automatically sets the number to 25.

## **Word Pro: TileWindow property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TILEWINDOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

TileType

### **Syntax**

tilewindowvalue = [objectreference].TileWindow

[objectreference].TileWindow = tilewindowvalue

### **Legal values**

\$LwpTileTypeCascade (1946)

\$LwpTileTypeHorz (1948)

\$LwpTileTypeVert (1947)

### **Usage**

**Word Pro: TimedSaveFileExtension property**

{button ,AL(^H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TIMEDSAVEFILEEXTENSION\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default file extension for time-saved files.

**Data Type**

[String](#)

**Syntax**

timedsavefileextensionvalue = [objectreference].TimedSaveFileExtension

[objectreference].TimedSaveFileExtension = timedsavefileextensionvalue

**Legal values**

Valid file extension characters. Default is .~TS.

**Usage**

Use this property to change the default file extension for time-saved files. You can set this property to Empty.



**Word Pro: Time property**

{button ,AL('H\_NOTELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TIME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)The time that a specific note layout was created.

**Data Type**

Long

**Syntax**

timevalue = [objectreference].Time

[objectreference].Time = timevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

The value that this property returns represents the number of seconds that have elapsed since midnight on January 1, 1970.

**Word Pro: TitleBarDocNumber property**

{button ,AL('H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TITLEBARDOCNUMBER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

titlebardocnumbervalue = [objectreference].TitleBarDocNumber

[objectreference].TitleBarDocNumber = titlebardocnumbervalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: TitleBarVisible property**

{button ,AL('H\_APPLICATIONWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TITLEBARVISIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A flag that tells you whether or not the TitleBar is turned on.

**Data Type**

[Integer](#)

**Syntax**

titlebarvisiblevalue = [objectreference].TitleBarVisible

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default value is True (-1).

**Usage**

If you are in Clean Screen mode, the value is False (0).

**Word Pro: Title property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TITLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

String

**Syntax**

titlevalue = [objectreference].Title

[objectreference].Title = titlevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: TOCNumEntries property**

{button ,AL('H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TOCNUMENTRIES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

tocnumentriesvalue = [objectreference].TOCNumEntries

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: TOCRange property**

{button ,AL('H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TOCRANGE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

tochangevalue = [objectreference].TOCRange

[objectreference].TOCRange = tochangevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## **Word Pro: TOCSource property**

{button ,AL('H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TOCSOURCE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

GenerateFrom

### **Syntax**

tocsourcevalue = [objectreference].TOCSource

[objectreference].TOCSource = tocsourcevalue

### **Legal values**

\$LwpGenerateFromCurrentdivision (362)

\$LwpGenerateFromCurrentleveldivision (361)

\$LwpGenerateFromCurrentsection (363)

\$LwpGenerateFromEntiredocument (360)

\$LwpGenerateFromMarker (364)

### **Usage**

## Word Pro: TopExternalMargin property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_TOEXTERNALMARGIN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to set the amount of margin space that is present above a layout object.

### Data Type

Long

### Syntax

topexternalmarginvalue = [objectreference].TopExternalMargin

[objectreference].TopExternalMargin = topexternalmarginvalue

### Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### Usage

This property can't be set individually for FrameLayout objects within Word Pro. It is combined with all external margin values in the "Padding around border" setting, located on the Size & Margins panel of the InfoBox.



## Word Pro: TopIntArea property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERBLEGROUPLAYOUT_CLASS;H_SUPERBLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERBLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_TOPINTAREA_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Not implemented.

## Data Type

Long

## Syntax

topintareavalue = [objectreference].TopIntArea

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

## Word Pro: TopLeftCellRowId property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_TOPLEFTCELLROWID_PROPERTY_EXSCRIPT',1)} See example
```

(Read-only) Indicates the row ID of the top left cell when cells are connected in a layout object. In an unconnected cell, this property returns the current cell's row ID.

## Data Type

Integer

## Syntax

topleftcellrowidvalue = [objectreference].TopLeftCellRowId

## Legal values

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

## Usage

The row ID is a zero based value, which means that the first row in a table has a row ID value of zero.

## **Word Pro: Top property**

{button ,AL(^H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TOP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[StatusBar]

The top position of the status bar on the workspace.

[ApplicationWindow]

The top position of the application window on the desktop.

## **Data Type**

Long

## **Syntax**

topvalue = [objectreference].Top

[objectreference].Top = topvalue

## **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## **Usage**

This property is not valid for IconBarManager.

[StatusBar]

Use this property to obtain the top position of the status bar on the workspace.

[ApplicationWindow]

Use this property to manipulate the top position of the application window on the desktop.

**Word Pro: TotalEditingTime property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TOTALEDITINGTIME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read only) The total number of seconds that a document has been open for editing.

**Data Type**

Long

**Syntax**

totaleditingtimevalue = [objectreference].TotalEditingTime

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

### **Word Pro: TrailingText property**

{button ,AL(^H\_ENDNOTEDIVISIONGROUPNUM\_CLASS;H\_ENDNOTEDIVISIONNUM\_CLASS;H\_ENDNOTEDOCNUM\_CLASS;H\_FOOTNOTENUMBERING\_CLASS;H\_FOOTNOTENUMOPT\_CLASS';0)} [See list of classes](#)

{button ,AL(^H\_TRAILINGTEXT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Adds a string of characters after an endnote reference.

### **Data Type**

[String](#)

### **Syntax**

trailingtextvalue = [objectreference].TrailingText

[objectreference].TrailingText = trailingtextvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

Equivalent to choosing Create - Footnote/Endnote, clicking Options, selecting "Enclosed by text before," and inserting text in the "text after" box located on the Numbering panel.

## **Word Pro: TypeAbove property**

{button ,AL('H\_PARAGRAPHBORDER\_CLASS;H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TYPEABOVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

ParaBorderWidth

### **Syntax**

typeabovevalue = [objectreference].TypeAbove

[objectreference].TypeAbove = typeabovevalue

### **Legal values**

\$LwpParaBorderWidthMargin (2052)

\$LwpParaBorderWidthNone (1624)

\$LwpParaBorderWidthOther (2053)

\$LwpParaBorderWidthText (2051)

### **Usage**

## **Word Pro: TypeBelow property**

{button ,AL('H\_PARAGRAPHBORDER\_CLASS;H\_SPACING\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TYPEBELOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

ParaBorderWidth

### **Syntax**

typebelowvalue = [objectreference].TypeBelow

[objectreference].TypeBelow = typebelowvalue

### **Legal values**

\$LwpParaBorderWidthMargin (2052)

\$LwpParaBorderWidthNone (1624)

\$LwpParaBorderWidthOther (2053)

\$LwpParaBorderWidthText (2051)

### **Usage**

## **Word Pro: TypeRight property**

{button ,AL('H\_PARAGRAPHBORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TYPERIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Variant (Enumerated)

ParaBorderWidth

### **Syntax**

typerightvalue = [objectreference].TypeRight

[objectreference].TypeRight = typerightvalue

### **Legal values**

\$LwpParaBorderWidthMargin (2052)

\$LwpParaBorderWidthNone (1624)

\$LwpParaBorderWidthOther (2053)

\$LwpParaBorderWidthText (2051)

### **Usage**



## **Word Pro: Type property**

{button ,AL(^H\_CHARACTERSTYLE\_CLASS;H\_FOOTNOTE\_CLASS;H\_PARAGRAPHSTYLE\_CLASS;H\_POWERFIELD\_CLASS;H\_SPACING\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_TYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) Indicates what type of style this object is. For CharacterStyle objects, the default is always "\$LwpCharStyleTypeCharacter."

### **Data Type**

Variant (Enumerated)

CharStyleType

### **Syntax**

typevalue = [objectreference].Type

### **Legal values**

\$LwpCharStyleTypeCharacter (140)

### **Usage**

## Word Pro: Typos property

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_TYPOS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### Data Type

Long (Enumerated Bitmask)

TypoChoices

### Syntax

typosvalue = [objectreference].Typos

[objectreference].Typos = typosvalue

### Legal values

LwpTypoChoicesListbox1 (&H1)

LwpTypoChoicesListbox2 (&H2)

LwpTypoChoicesListbox3 (&H4)

LwpTypoChoicesListbox4 (&H8)

LwpTypoChoicesListboxall (&HF)

### Usage

**Word Pro: Underline property**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UNDERLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

underlinevalue = [objectreference].Underline

[objectreference].Underline = underlinevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: UndoEnable property**

{button ,AL('H\_SCRIPT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UNDOENABLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

undoenablevalue = [objectreference].UndoEnable

[objectreference].UndoEnable = undoenablevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: UpdateFields property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPDATEFIELDS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Updates any power fields before the document prints.

**Data Type**

Integer

**Syntax**

updatefieldsvalue = [objectreference].UpdateFields

[objectreference].UpdateFields = updatefieldsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to choosing File - Print, clicking Options, and selecting "Fields" in the Update section in the Print Options dialog box.

**Word Pro: UpdateOnLoadImmediate property**

{button ,AL(^H\_POWERFIELD\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_UPDATEONLOADIMMEDIATE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(WriteOnly)

**Data Type**

[Integer](#)

**Syntax**

[objectreference].UpdateOnLoadImmediate = updateonloadimmediatevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: UpperCase property**

{button ,AL('H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_UPPERCASE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

uppercasevalue = [objectreference].UpperCase

[objectreference].UpperCase = uppercasevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: UseConsistentSpaceBetweenSentences property**

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USECONSISTENTSPACEBETWEENSENTENCES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

useconsistentspacebetweensentencesvalue = [objectreference].UseConsistentSpaceBetweenSentences

[objectreference].UseConsistentSpaceBetweenSentences = useconsistentspacebetweensentencesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: UseContents property**

{button ,AL('H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USECONTENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

usecontentsvalue = [objectreference].UseContents

[objectreference].UseContents = usecontentsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: UseDefaultPrinter property**

{button ,AL('H\_PRINTMANAGER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USEDEFAULTPRINTER\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

usedefaultprintvalue = [objectreference].UseDefaultPrinter

[objectreference].UseDefaultPrinter = usedefaultprintvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## **Word Pro: UsedFirstPageHeight property**

{button ,AL(^H\_DIVISION\_CLASS;H\_TEXTDOCUMENT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_USEDFIRSTPAGEHEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Long

### **Syntax**

usedfirstpageheightvalue = [objectreference].UsedFirstPageHeight

### **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

## **Word Pro: UseEncrypt property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USEENCRYPT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

useencryptvalue = [objectreference].UseEncrypt

[objectreference].UseEncrypt = useencryptvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

**Word Pro: UseExcludeRect property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USEEXCLUDERECT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

useexcluderectvalue = [objectreference].UseExcludeRect

[objectreference].UseExcludeRect = useexcluderectvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: UseFindStyle property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USEFINDSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to find a paragraph style when using Find & Replace.

**Data Type**

[Integer](#)

**Syntax**

usefindstylevalue = [objectreference].UseFindStyle

[objectreference].UseFindStyle = usefindstylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Use this property to find a specific paragraph style in Find & Replace. Equivalent to choosing Edit - Find & Replace Text, clicking Options, clicking the Font button in the "Find options" section, and selecting a find style in the "Style" list box.

## Word Pro: UseFooter property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_USEFOOTER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to add or remove a footer in a layout object.

## Data Type

Integer

## Syntax

usefootervalue = [objectreference].UseFooter

[objectreference].UseFooter = usefootervalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

When you assign a value of True to this property, the footer object inserted in the layout object has a height equal to the bottom margin of the layout object and a width equal to the total width of the layout object.

**Word Pro: UseGreeting property**

{button ,AL('H\_DOCCONTROL\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USEGREETING\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether or not a greeting displays when a file is opened.

**Data Type**

[Integer](#)

**Syntax**

usegreetingvalue = [objectreference].UseGreeting

[objectreference].UseGreeting = usegreetingvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

Equivalent to choosing File - TeamSecurity and selecting "Display Greeting with this text" box in the Editing Rights panel.



## Word Pro: UseHeader property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_USEHEADER_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Allows you to add or remove a header in a layout object.

## Data Type

Integer

## Syntax

useheadervalue = [objectreference].UseHeader

[objectreference].UseHeader = useheadervalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

When you assign a value of True to this property, the header object inserted in the layout object has a height equal to the top margin of the layout object and a width equal to the total width of the layout object.

**Word Pro: UseNextStyle property**

{button ,AL('H\_BREAKS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USENEXTSTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

usenextstylevalue = [objectreference].UseNextStyle

[objectreference].UseNextStyle = usenextstylevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

## Word Pro: UsePrinterSettings property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_USEPRINTERSETTINGS_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Instructs the layout object to retrieve printer setting information, such as orientation, margins, printer bin, scale, and so on.

## Data Type

Integer

## Syntax

useprintersettingsvalue = [objectreference].UsePrinterSettings

[objectreference].UsePrinterSettings = useprintersettingsvalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: UserClassNameApp property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERCLASSNAMEAPP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

userclassnameappvalue = [objectreference].UserClassNameApp

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: UserClassNameFull property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERCLASSNAMEFULL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

userclassnamefullvalue = [objectreference].UserClassNameFull

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: UserClassNameShort property**

{button ,AL('H\_OLEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERCLASSNAMESHORT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[String](#)

**Syntax**

userclassnameshortvalue = [objectreference].UserClassNameShort

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: UserDictFiles property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERDICTFILES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The default user dictionary file(s).

### **Data Type**

[String](#)

### **Syntax**

userdictfilesvalue = [objectreference].UserDictFiles

### **Legal values**

A valid user dictionary file with the file extension .UDC.

### **Usage**

Equivalent to the "Default user dictionary" value on the Default files panel of the Word Pro Preferences dialog box. In the Word Pro interface, "Default user dictionary" can contain multiple file names. You can use this property to read multiple file names entered by the user.

## **Word Pro: UserDictionaryPath property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERDICTIONARYPATH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) The default path (drive and directory) for the Word Pro user dictionary.

### **Data Type**

String

### **Syntax**

userdictionarypathvalue = [objectreference].UserDictionaryPath

[objectreference].UserDictionaryPath = userdictionarypathvalue

### **Legal values**

A valid path including drive, directory, and file name.

### **Usage**

Equivalent to the "User dictionaries" option on the Locations panel of the Word Pro Preferences dialog box. In the Word Pro interface, this value can contain multiple paths. You can use this property to clear all paths before setting the default or first user dictionary path, or you can use the property, UserDictionaryPaths, to read multiple paths entered by the user.



## Word Pro: UserDictStates property

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERDICTSTATES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) The state (on or off) of each user dictionary.

### Data Type

String

### Syntax

userdictstatesvalue = [objectreference].UserDictStates

### Legal values

Although the legal value for this property is a String, the string contains a series of Integers (0s and 1s) that indicate if a dictionary is turned on or off. False (0) means the dictionary is not being used. True (1) means the dictionary is being used.

The series of integers is separated by commas and corresponds to the string of user dictionary files found in the UserDictFiles property.

For example, UserDictFiles may be set to "d:\wordpro\userdic.udc; c:\lotus\components\wordpro\userdic.udc;s:\lotus\userdic.udc." Then the UserDictStates property might contain the string "0,1,0," indicating that the only user dictionary enabled is the second one, or c:\lotus\components\wordpro\userdic.udc.

### Usage

Equivalent to the "User Dictionary(s) to use" option in the Spell Check Options dialog box. The user can select multiple user dictionaries in this box.

**Word Pro: UseRelative property**

{button ,AL('H\_INDENT\_CLASS;H\_RELATIVEINIDENT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERELATIVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Integer

**Syntax**

userrelativevalue = [objectreference].UseRelative

[objectreference].UseRelative = userrelativevalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: UseReplaceStyle property**

{button ,AL('H\_FINDANDREPLACE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USEREPLACESTYLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Enables the user to replace a paragraph style in Find & Replace.

**Data Type**

[Integer](#)

**Syntax**

usereplacestylevalue = [objectreference].UseReplaceStyle

[objectreference].UseReplaceStyle = usereplacestylevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

Use this property to replace a paragraph style in Find & Replace. Equivalent to choosing Edit - Find & Replace Text, clicking Options, clicking the Font button in the "Replace options" section, and choosing a paragraph style in the "Style" list box.

## **Word Pro: UserInitials property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERINITIALS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

userinitialsvalue = [objectreference].UserInitials

[objectreference].UserInitials = userinitialsvalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: UserLinkName property**

{button ,AL('H\_DDELINK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERLINKNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) This property provides the user name for a DdeLink.

**Data Type**

[String](#)

**Syntax**

userlinknamevalue = [objectreference].UserLinkName

**Legal values****Usage**

This property gives you the user name for the DdeLink (composed of the server name, the topic name, and the item name).

### **Word Pro: UserName property**

{button ,AL('H\_INDEXSECTION\_CLASS;H\_PREFERENCES\_CLASS;H\_SECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

usernamevalue = [objectreference].UserName

[objectreference].UserName = usernamevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

**Word Pro: UserPassword property**

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USERPASSWORD\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Stores the attempted password that a user enters in the "Enter Document Control Password" message box.

**Data Type**

[String](#)

**Syntax**

userpasswordvalue = [objectreference].UserPassword

[objectreference].UserPassword = userpasswordvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

This property is only available if Password(263) is chosen as the legal value for FileProtectionType property.

### **Word Pro: UseSeparatorLine property**

{button ,AL(^H\_FOOTNOTECONTSEP\_CLASS;H\_FOOTNOTESEPARATOR\_CLASS;H\_FOOTNOTESEPOPT\_CLASS;0)} [See list of classes](#)

{button ,AL(^H\_USESEPARATORLINE\_PROPERTY\_EXSCRIPT,1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

useseparatorlinevalue = [objectreference].UseSeparatorLine

[objectreference].UseSeparatorLine = useseparatorlinevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



### **Word Pro: UsesHelp property**

{button ,AL('H\_CLICKHERE\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USESHELP\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates whether a ClickHere object has bubble help.

### **Data Type**

[Integer](#)

### **Syntax**

useshelpvalue = [objectreference].UsesHelp

[objectreference].UsesHelp = useshelpvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: UsesPalette property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USESPALETTE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

usespalettevalue = [objectreference].UsesPalette

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

### **Word Pro: UseSuperscriptReferenceNum property**

{button ,AL(`H\_ENDNOTEDIVISIONGROUPNUM\_CLASS;H\_ENDNOTEDIVISIONNUM\_CLASS;H\_ENDNOTEDOC  
NUM\_CLASS;H\_FOOTNOTENUMBERING\_CLASS;H\_FOOTNOTENUMOPT\_CLASS';0)} [See list of classes](#)

{button ,AL(`H\_USESUPERSCRIPTREFERENCENUM\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Changes endnote/footnote references in the text to superscript.

### **Data Type**

Integer

### **Syntax**

usesuperscriptreferencenumvalue = [objectreference].UseSuperscriptReferenceNum

[objectreference].UseSuperscriptReferenceNum = usesuperscriptreferencenumvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

Equivalent to choosing Create - Footnote/Endnote, clicking Options, and selecting "Superscript reference number" on the Numbering panel.

## **Word Pro: UseTwoSpacesBetweenSentences property**

{button ,AL('H\_FORMATCHECKPREF\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_USETWOSPACESBETWEENSENTENCES\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

usetwospacesbetweensentencesvalue = [objectreference].UseTwoSpacesBetweenSentences

[objectreference].UseTwoSpacesBetweenSentences = usetwospacesbetweensentencesvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: Valid property**

{button ,AL('H\_BULLET\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VALID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

Variant (Enumerated)

CommandState

### **Syntax**

validvalue = [objectreference].Valid

### **Legal values**

\$LwpCommandStateOff (151)

\$LwpCommandStateOn (152)

\$LwpCommandStateStyle (153)

### **Usage**

## **Word Pro: Value property**

{button ,AL('H\_DOCINFO\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VALUE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

valuevalue = [objectreference].Value

[objectreference].Value = valuevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: VerificationResult property

{button ,AL('H\_FILEPROTECTION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERIFICATIONRESULT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Word Pro always verifies that a user has access to a document. VerificationResult stores the result of Word Pro's verification.

### Data Type

Variant (Enumerated)

Verify

### Syntax

verificationresultvalue = [objectreference].VerificationResult

[objectreference].VerificationResult = verificationresultvalue

### Legal values

\$LwpVerifyRestrictByEditorViolation (1960)

\$LwpVerifyRestrictByInvalidPassword (1956)

\$LwpVerifyRestrictInternalPurpose (1958)

\$LwpVerifyRestrictInternalPurpose2 (1959)

\$LwpVerifyRestrictToEditors (1954)

\$LwpVerifyRestrictToOriginalAuthor (1955)

\$LwpVerifySuccessNoRestrictions (1957)

### Usage

Use this property to track or change results of Word Pro's TeamSecurity verification process.

## **Word Pro: VersionID property**

{button ,AL('H\_BASEOBJECT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERSIONID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only) A numeric expression that represents the version number of the LotusScript engine in which an object was generated. This is essential when comparing the relative functionality of certain LotusScript language elements and Word Pro classes.

### **Data Type**

Long

### **Syntax**

versionidvalue = [objectreference].VersionID

### **Legal values**

You cannot set the value of this property.

### **Usage**

Use this property to determine in which version of the Word Pro object model an object was generated. By comparing the VersionID properties of two language elements, you can determine if the two language elements are likely to be compatible. This is especially useful when debugging or analyzing inherited or legacy scripts.



**Word Pro: VersionName property**

{button ,AL('H\_VERSION\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERSIONNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

versionnamevalue = [objectreference].VersionName

[objectreference].VersionName = versionnamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## **Word Pro: VersionRemarks property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERSIONREMARKS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to print version remarks in a document.

### **Data Type**

[Integer](#)

### **Syntax**

versionremarksvalue = [objectreference].VersionRemarks

[objectreference].VersionRemarks = versionremarksvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## Word Pro: VertAlign property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_VERTICALIGN_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Sets the vertical alignment of a layout object.

### Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

### Syntax

vertalignvalue = [objectreference].VertAlign

[objectreference].VertAlign = vertalignvalue

### Legal values

<u>Value</u>	<u>Effect</u>
\$LtsAlignmentTop (1056964614)	Sets the vertical alignment of a layout object to the top.
\$LtsAlignmentVertCenter (1056964615)	Sets the vertical alignment of a layout object to the center.
\$LtsAlignmentBottom (1056964616)	Sets the vertical alignment of a layout object to the bottom.

### Usage

Equivalent to the "Vertical alignment" option, located on the Misc panel of the InfoBox for certain layout objects.

## **Word Pro: VerticalSplitWindow property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VERTICALSPLITWINDOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if Word Pro will split the .MDI window vertically when the NewWindow method is called.

### **Data Type**

[Integer](#)

### **Syntax**

verticalsplitwindowvalue = [objectreference].VerticalSplitWindow

[objectreference].VerticalSplitWindow = verticalsplitwindowvalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the Split Left-Right option on the View menu.

**Word Pro: VertScrollBarVisible property**

{button ,AL(^H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_VERTSCROLLBARVISIBLE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Integer](#)

**Syntax**

vertscrollbarvisiblevalue = [objectreference].VertScrollBarVisible

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: ViewLevel property**

{button ,AL('H\_DOCWINDOW\_CLASS;H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VIEWLEVEL\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to print a document at different zoom levels.

**Data Type**

[Integer](#)

**Syntax**

viewlevelvalue = [objectreference].ViewLevel

[objectreference].ViewLevel = viewlevelvalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

## Word Pro: ViewType property

{button ,AL('H\_DOCWINDOW\_CLASS;H\_PRINTSETTINGS\_CLASS;H\_WINVIEWPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_VIEWTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to print a specific view of a document.

### Data Type

Variant (Enumerated)

PresentationType

### Syntax

viewtypevalue = [objectreference].ViewType

[objectreference].ViewType = viewtypevalue

### Legal values

[PrintSettings and DocWindow]

The value of this property must be one of the following strings or its numeric equivalent (in parentheses) when this property is found on a PrintSettings or DocWindow object.

\$LwpPresentationTypeDraft (1647)

\$LwpPresentationTypeLayout (1646)

\$LwpPresentationTypeOutline (1649)

\$LwpPresentationTypePagesort (1650)

\$LwpPresentationTypeWymiwyg (1648)

[WinViewPrefs]

The value of this property must be one of the following strings or its hexadecimal equivalent (in parentheses) when this property is found on a WinViewPrefs object.

LwpViewsStandard (&H1)

LwpViewsEnlarged (&H2)

LwpViewsFullpage (&H4)

LwpViewsCustom (&H8)

LwpViewsFacingpages (&H10)

LwpViewsPagesort (&H20)

LwpViewsMultiplepages (&H40)

LwpViewsPagemargin (&H80)

LwpViewsPagewidth (&H100)

LwpViewsReduced (&H200)

LwpViewsMoreenlarged (&H400)

### Usage

[WinViewPrefs]

The legal values for this property on a WinViewPrefs object are derived from a bitmask. This bitmask allows Word Pro to always show its true value.

## **Word Pro: Visible property**

{button ,AL(^H\_APPLICATION\_CLASS;H\_APPLICATIONWINDOW\_CLASS;H\_DOCWINDOW\_CLASS;H\_STATUSBAR\_CLASS;H\_WINDOW\_CLASS;H\_WPAPPLICATION\_CLASS^,0)} [See list of classes](#)

{button ,AL(^H\_VISIBLE\_PROPERTY\_EXSCRIPT^,1)} [See example](#)

(Read-only) Returns a boolean value indicating whether or not the object from which you called this property is visible to the user.

### **Data Type**

[Integer](#)

### **Syntax**

visiblevalue = [objectreference].Visible

### **Legal values**

The value of this property cannot be set by a script. The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

You can call this property from WPAplication, ApplicationWindow, DocWindow, or StatusBar.

Use this property to determine whether Word Pro, the application workspace, a document window, or the status bar is currently visible to the user.



## Word Pro: WasDeletedInRevMarkMode property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_WASDELETEDINREVMARKMODE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a layout object was deleted while in revision marking mode.

## Data Type

[Integer](#)

## Syntax

wasdeletedinrevmarkmodevalue = [objectreference].WasDeletedInRevMarkMode

[objectreference].WasDeletedInRevMarkMode = wasdeletedinrevmarkmodevalue

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

## Word Pro: WasInsertedInRevMarkMode property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL('H_WASINSERTEDINREVMARKMODE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Indicates whether or not a layout object was inserted while in revision marking mode.

## Data Type

[Integer](#)

## Syntax

```
wasinsertedinrevmarkmodevalue = [objectreference].WasInsertedInRevMarkMode
```

## Legal values

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

## Usage

**Word Pro: WasPasted property**

{button ,AL('H\_BOOKMARK\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WASPASTED\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) This property is not used.

**Data Type**

[Integer](#)

**Syntax**

waspastedvalue = [objectreference].WasPasted

[objectreference].WasPasted = waspastedvalue

**Legal values**

This property is not used.

**Usage**

## **Word Pro: Weight property**

{button ,AL('H\_FONTMETRICS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WEIGHT\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

### **Data Type**

[Integer](#)

### **Syntax**

weightvalue = [objectreference].Weight

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**

## Word Pro: Where property

{button ,AL('H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FINDANDREPLACE\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASSES;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WHERE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates where an anchor is connected to a parent layout object.

[FindandReplace]

Indicates where a Find & Replace text or character string is located in the document.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

wherevalue = [objectreference].Where

[objectreference].Where = wherevalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpWhereTypeLayoutUpperRight (1978)	Setting this value connects the anchor to the upper right corner of the parent layout object.
\$LwpWhereTypeLowerLeft (1981)	Setting this value connects the anchor to the lower left corner of the parent layout object.
\$LwpWhereTypeLowerRight (1983)	Setting this value connects the anchor to the lower right corner of the parent layout object.
\$LwpWhereTypeMiddle (1984)	Setting this value connects the anchor to the middle of the parent layout object.
\$LwpWhereTypeMiddleBottom (1982)	Setting this value connects the anchor to the middle of the bottom corner of the parent layout object.
\$LwpWhereTypeMiddleLeft (1979)	Setting this value connects the anchor to the middle of the left corner of the parent layout object.
\$LwpWhereTypeMiddleRight (1980)	Setting this value connects the anchor to the middle of the right corner of the parent layout object.
\$LwpWhereTypeMiddleTop (1977)	Setting this value connects the anchor to the middle of the top corner of the parent layout object.
\$LwpWhereTypeUpperLeft (1976)	Setting this value connects the anchor to the upper left corner of the parent layout object.

## Usage

[FindandReplace]

Use this property to search a specific area for Find & Replace. The values are:

- (0) Search an entire document
- (1) Search the current division
- (2) Search the current section
- (3) Search highlighted text

Equivalent to choosing Edit - Find & Replace Text, clicking Options, and choosing a selecting from the "Look in" list

box in the "Find & replace scope" section.

**Word Pro: WidowOrphan property**

{button ,AL('H\_OPTIONS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDOWORPHAN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[Integer](#)

**Syntax**

widoworphanvalue = [objectreference].WidowOrphan

[objectreference].WidowOrphan = widoworphanvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

**Word Pro: WidthAbove property**

{button ,AL('H\_CHARACTERBORDER\_CLASS;H\_PARAGRAPHBORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDTHABOVE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

widthabovevalue = [objectreference].WidthAbove

[objectreference].WidthAbove = widthabovevalue

**Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**



## **Word Pro: WidthBelow property**

{button ,AL('H\_CHARACTERBORDER\_CLASS;H\_PARAGRAPHBORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDTHBELOW\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Long

### **Syntax**

widthbelowvalue = [objectreference].WidthBelow

[objectreference].WidthBelow = widthbelowvalue

### **Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

### **Usage**

**Word Pro: WidthInTwips property**

{button ,AL('H\_BORDER\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDTHINTWIPS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

Long

**Syntax**

widthintwipsvalue = [objectreference].WidthInTwips

[objectreference].WidthInTwips = widthintwipsvalue

**Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

**Word Pro: WidthOfColumn property**

{button ,AL('H\_COLUMN\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDTHOFCOLUMN\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to set the width of a specific newspaper column.

**Data Type**

[Long](#)

**Syntax**

widthofcolumnvalue = [objectreference].WidthOfColumn

[objectreference].WidthOfColumn = widthofcolumnvalue

**Legal values**

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

**Usage**

Before you can change the width of a specific newspaper column, you must select its layout and then specify its width.

## Word Pro: Width property

{button ,AL('H\_APPLICATIONWINDOW\_CLASS;H\_BASECONTAINER\_CLASS;H\_BORDER\_CLASS;H\_CELLCONTAINER\_CLASS;H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DOCWINDOW\_CLASS;H\_DROPCAPCONTAINER\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FONT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMECONTAINER\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GRAPHIC\_CLASS;H\_GRAPHICOLEBJECT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_ICONBAR\_CLASS;H\_LAYOUT\_CLASS;H\_NOTECONTAINER\_CLASS;H\_NOTELAYOUT\_CLASS;H\_OLEOBJECT\_CLASS;H\_PAGECONTAINER\_CLASS;H\_PAGELAYOUT\_CLASS;H\_PARALLELCOLSCONTAINER\_CLASS;H\_ROWCONTAINER\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYCONTAINER\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_STATUSBAR\_CLASS;H\_SUBPAGECONTAINER\_CLASS;H\_SUPERPAGECONTAINER\_CLASS;H\_SUPERTABLECONTAINER\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLECONTAINER\_CLASS;H\_TABLEHEADINGLAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TABLEONLYCONT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS;H\_WINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WIDTH\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

[IconBar]

Used to represent the actual width of an icon bar object; the actual width of the entire object.

[Layout]

Allows you to set or return the actual width of a layout object.

[StatusBar]

Used in the StatusBar class to return the actual width of the status bar object in Twips.

[ApplicationWindow]

The actual width of the application window.

[Cell, Parallel Column, Row, SuperTable, and TableOnly Containers]

Allows you to set or return the total width of a container from the outside border edge to the outside border edge.

## Data Type

Long

## Syntax

widthvalue = [objectreference].Width

[objectreference].Width = widthvalue

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

[ApplicationWindow]

Used to manipulate the width of the application window.

[IconBar]

Used to represent the actual width of an icon bar object. Returns the distance in Twips from the left border of an icon bar object to its right border; the actual width of the entire object.

[IconBarManager]

This property is not valid for IconBarManager.

[StatusBar]

This property appears with data type Long in the Window class.

[Layout]

The width of a layout object is the distance between the left border and right border of the layout object.

**Word Pro: WindowId property**

{button ,AL('H\_DOCWINDOW\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WINDOWID\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-only)

**Data Type**

[Long](#)

**Syntax**

windowidvalue = [objectreference].WindowId

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: WindowsName property**

{button ,AL('H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WINDOWSNAME\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

**Data Type**

[String](#)

**Syntax**

windowsnamevalue = [objectreference].WindowsName

[objectreference].WindowsName = windowsnamevalue

**Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

**Usage**

**Word Pro: WithComments property**

{button ,AL('H\_PRINTSETTINGS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WITHCOMMENTS\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Allows you to print each comment note at its location in the document.

**Data Type**

[Integer](#)

**Syntax**

withcommentsvalue = [objectreference].WithComments

[objectreference].WithComments = withcommentsvalue

**Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

**Usage**

A comment note must be open in a document before you can print it. Equivalent to choosing File - Print, clicking Options, and selecting "With comments" in the Print options section in the Print Options dialog box.

## **Word Pro: WordDoubleUnderline property**

{button ,AL(^H\_FONT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_WORDDOUBLEUNDERLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

[Integer](#)

### **Syntax**

worddoubleunderlinevalue = [objectreference].WordDoubleUnderline

[objectreference].WordDoubleUnderline = worddoubleunderlinevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**



## **Word Pro: WordUnderline property**

{button ,AL(^H\_FONT\_CLASS;H\_WPAPPLICATION\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_WORDUNDERLINE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

Integer

### **Syntax**

wordunderlinevalue = [objectreference].WordUnderline

[objectreference].WordUnderline = wordunderlinevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values.

### **Usage**

## **Word Pro: WorkingType property**

{button ,AL('H\_USERINTERFACEPREFS\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_WORKINGTYPE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Indicates if "Use working type" is enabled in Word Pro Preferences.

### **Data Type**

[Integer](#)

### **Syntax**

workingtypevalue = [objectreference].WorkingType

[objectreference].WorkingType = workingtypevalue

### **Legal values**

The legal values for this property are -1 and 0. If you prefer, you can use the LotusScript constants of True (-1) and False (0) instead of the integer values. Default is False (0).

### **Usage**

Equivalent to the "Use working type" option on the Locations panel of the Word Pro Preferences dialog box. If the value for this property is True (-1), Word Pro continues to display files of the last type the user chose in the Open and Save dialog boxes. If the value is False (0), Word Pro uses the default value .LWP.

## Word Pro: WrapType property

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS';0)} See list of classes
```

```
{button ,AL(^H_WRAPTYPE_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write) Specifies how text should wrap around a layout object.

## Data Type

Data type for this property is [Variant](#) which allows the value of this property to be one of the constants listed below or its numeric equivalent (in parentheses).

## Syntax

wrapypevalue = [objectreference].WrapType

[objectreference].WrapType = wrapypevalue

## Legal values

<u>Value</u>	<u>Effect</u>
\$LwpWrapLayoutWrapRight (1999)	Setting this value wraps text to the right of a layout object.
\$LwpWrapTypeLayoutNoWrapAround (1997)	Setting this value prevents text from wrapping around a layout object. It allows text to flow behind a layout object.
\$LwpWrapTypeLayoutNoWrapBeside (1996)	Setting this value wraps text above and below a layout object.
\$LwpWrapTypeLayoutWrapAround (1995)	Setting this value wraps text around a layout object.
\$LwpWrapTypeLayoutWrapLeft (1998)	Setting this value wraps text to the left of a layout object.
\$LwpWrapTypeLayoutWrapRight (2070)	Setting this value wraps text to the right of a layout object.

## Usage

Equivalent to the "Wrap options" setting on the Placement panel of the InfoBox for frame or table layout objects.

## Word Pro: XOffset property

{button ,AL(^H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DOCWINDOW\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLEAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_XOFFSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Horizontally offsets the content of a layout without requiring a change in margins.

## Data Type

[Long](#)

## Syntax

xoffsetvalue = [objectreference].XOffset

[objectreference].XOffset = xoffsetvalue

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

This property is used for cropping graphics in a layout object.

## Word Pro: XPosition property

```
{button ,AL(^H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SHADOW_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL(^H_XPOSITION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write)

[Layout]

The X coordinate for the origin of this layout.

[Shadow]

Determines the X axis depth of a shadow object for text, frames, tables, headers, footers, and pages in a document.

## Data Type

Long

## Syntax

xpositionvalue = [objectreference].XPosition

[objectreference].XPosition = xpositionvalue

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

You can use the XPosition property to set the depth of a shadow object from left to right for text, frames, tables, headers, footers, and pages in a document.

## Word Pro: YOffset property

{button ,AL(^H\_CELLGROUPLAYOUT\_CLASS;H\_CELLLAYOUT\_CLASS;H\_COLUMNGROUPLAYOUT\_CLASS;H\_CONNECTEDLAYOUT\_CLASS;H\_DOCWINDOW\_CLASS;H\_DROPCAPLAYOUT\_CLASS;H\_ENDNOTELAYOUT\_CLASS;H\_FOOTERLAYOUT\_CLASS;H\_FOOTNOTELAYOUT\_CLASS;H\_FRAMEGROUPLAYOUT\_CLASS;H\_FRAMELAYOUT\_CLASS;H\_GROUPLAYOUT\_CLASS;H\_HEADERLAYOUT\_CLASS;H\_LAYOUT\_CLASS;H\_NOTELAYOUT\_CLASS;H\_PAGELAYOUT\_CLASS;H\_ROWGROUPLAYOUT\_CLASS;H\_ROWLAYOUT\_CLASS;H\_RUBYLAYOUT\_CLASS;H\_SUPERTABLEGROUPLAYOUT\_CLASS;H\_SUPERTABLELAYOUT\_CLASS;H\_TABLEHEADINGLEAYOUT\_CLASS;H\_TABLELAYOUT\_CLASS;H\_TOCSUPERTABLELAYOUT\_CLASS',0)} [See list of classes](#)

{button ,AL(^H\_YOFFSET\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write) Vertically offsets the content of a layout without requiring a change in margins.

## Data Type

[Long](#)

## Syntax

yoffsetvalue = [objectreference].YOffset

[objectreference].YOffset = yoffsetvalue

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

This property is used for cropping graphics in a layout object.

## Word Pro: YPosition property

```
{button ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SHADOW_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS',0)} See list of classes
```

```
{button ,AL('H_YPOSITION_PROPERTY_EXSCRIPT',1)} See example
```

(Read-write)

[Layout]

The Y coordinate for the origin of this layout.

[Shadow]

Determines the Y axis depth of a shadow object for text, frames, tables, headers, footers, and pages in a document.

## Data Type

Long

## Syntax

ypositionvalue = [objectreference].YPosition

[objectreference].YPosition = ypositionvalue

## Legal values

Data type is Long but the unit of measurement used for this property is Twips. There are 1440 Twips per inch.

## Usage

You can use the YPosition property to set the depth of a shadow object up or down for text, frames, tables, headers, footers, and pages in a document.

## **Word Pro: ZipCode property**

{button ,AL('H\_PREFERENCES\_CLASS',0)} [See list of classes](#)

{button ,AL('H\_ZIPCODE\_PROPERTY\_EXSCRIPT',1)} [See example](#)

(Read-write)

### **Data Type**

String

### **Syntax**

zipcodevalue = [objectreference].ZipCode

[objectreference].ZipCode = zipcodevalue

### **Legal values**

The legal values for this property are determined by its data type. For more information about standard LotusScript data types, choose Edit - Scripts & Macros. Choose Show Script Editor. In the Script Editor, choose Help - Lotus Script.

### **Usage**



'Example: TOCNumEntries property

'This example script has not yet been created.

'Example: TOCRange property

'This example script has not yet been created.

'Example: TOCSource property

'This example script has not yet been created.

'Example: ToggleIconBar method

' This example toggles whether or not the SmartIcon bar is displayed.

.ToggleIconBar

'Example: TopBorder property

'This example script has not yet been created.

'Example: TopExternalMargin property

'This example script has not yet been created.

'Example: TopIntArea property

'This example script has not yet been created.

'Example: TopLeftCellRowId property

'This example script has not yet been created.



'Example: Top property

'This example script has not yet been created.

'Example: TotalEditingTime property

'This example script has not yet been created.

'Example: TrailingText property

'This example script has not yet been created.

'Example: TypeAbove property

'This example script has not yet been created.

'Example: TypeBelow property

'This example script has not yet been created.

'Example: TypeRight property

'This example script has not yet been created.

'Example: Type method

' This example uses the Type method to issues keystrokes which insert two  
' lines of text into the current document and then select both lines.  
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

```
.Type("Some sample [TAB] text.")  
.Type("[ENTER]")  
.Type("More sample [TAB] text.")  
.Type("[ENTER]")  
.Type("[SHIFTUP][SHIFTUP]")
```

'Example: Type property

'This example script has not yet been created.



'Example: Typos property

'This example script has not yet been created.

'Example: Underline method

' This example first inserts sample text in the current document and selects  
' the paragraph. The script then uses the Underline method to toggle the  
' underline attribute.

' RUNTIME DEPENDENCIES: You must have a document open with selected text  
' for this script to work.

.Text.InsertText "This is some sample text."

.SelectParagraph

.Underline

'Example: Underline property

'This example script has not yet been created.

'Example: UndoEnable property

'This example script has not yet been created.

'Example: UndoLevels property

'This example script has not yet been created.

'Example: UndoRedo method

' This example types some text into the current document which is undone and  
' then redone.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "Typing this text will be be undone and then redone"

.Undo

.UndoRedo

'Example: Undo method

' This example types some text into the current document which is then undone.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

.Type "Typing this text wil be be undone"

.Undo

'Example: Units property

'This example script has not yet been created.



```
'Example: UnregisterWPDataSet method
' This example creates a dataset named 'PhoneNumbers' off of the application
' object. Two dataset items are added and then printed to the Script Editor
' Output panel. The dataset is then removed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim AppDataSet As WPDataSet
Set AppDataSet = .RegisterWPDataSet("PhoneNumbers")
AppDataSet.SetData "Mark", "555-1234"
AppDataSet.SetData "Peyton", "555-5678"
Print AppDataSet.GetData("Mark", " ")
Print AppDataSet.GetData("Peyton", " ")
```

```
.UnRegisterWPDataSet "PhoneNumbers"
```

'Example: UpdateFields property

'This example script has not yet been created.

'Example: UpdateFootersText method

'This example script has not yet been created.

'Example: UpdateHeadersText method

'This example script has not yet been created.

'Example: UpdateIndexSection method  
'This example script has not yet been created.

'Example: UpdateLink method

'This example script has not yet been created.

'Example: UpdateOle method

'This example script has not yet been created.

'Example: UpdateOnLoadImmediate property

'This example script has not yet been created.



'Example: UpdatePageSizeChange method

' This example updates any changes to the size of the paper used to print the ' document.

' RUNTIME DEPENDENCIES: You must have a document open for this script to work.

Print .ActiveDocument.PrintManager.UpdatePageSizeChange

'Example: UpdatePowerFields method

'This example script has not yet been created.

'Example: UpdatePrinterChanges method

'This example script has not yet been created.

'Example: UpdateSelectedFields method

'This example script has not yet been created.

'Example: UpdateTOC method

'This example script has not yet been created.

'Example: Update method

'This example script has not yet been created.

'Example: UpperCase method

' This example toggles the uppercase attribute of the selected text, displays  
' a message box, then toggles the uppercase again.

' RUNTIME DEPENDENCIES: You must have a document open and some text selected  
' for this script to work.

**.Uppercase**

MessageBox "Click OK undo uppercase change.",MB\_OK,"Example Script"

**.Uppercase**

'Example: UpperCase property

'This example script has not yet been created.



'Example: UseConsistentSpaceBetweenSentences property  
'This example script has not yet been created.

'Example: UseContents property

'This example script has not yet been created.

'Example: UseDefaultPrinter property

'This example script has not yet been created.

'Example: UsedFirstPageHeight property

'This example script has not yet been created.

'Example: UseEncrypt property

'This example script has not yet been created.

'Example: UseExcludeRect property

'This example script has not yet been created.

'Example: UseFindStyle property

'This example script has not yet been created.

'Example: UseFooter property

'This example script has not yet been created.



'Example: UseGreeting property

'This example script has not yet been created.

'Example: UseHeader property

'This example script has not yet been created.

'Example: UseNextStyle property

'This example script has not yet been created.

'Example: UsePrinterSettings property  
'This example script has not yet been created.

'Example: UserClassNameApp property

'This example script has not yet been created.

'Example: UserClassNameFull property

'This example script has not yet been created.

'Example: UserClassNameShort property

'This example script has not yet been created.

'Example: UserDictFiles property

'This example script has not yet been created.



'Example: UserDictionaryFiles property  
'This example script has not yet been created.

'Example: UserDictionaryPaths property  
'This example script has not yet been created.

'Example: UserDictionaryPath property  
'This example script has not yet been created.

'Example: UserDictStates property

'This example script has not yet been created.

'Example: UseRelative property

'This example script has not yet been created.

'Example: UseReplaceStyle property  
'This example script has not yet been created.

'Example: UserInitials property

'This example script has not yet been created.

'Example: UserDefaults property  
'This example script has not yet been created.



'Example: UserLinkName property

'This example script has not yet been created.

'Example: UserName pro

perty

'This example script has not yet been created.

'Example: UserPassword property

'This example script has not yet been created.

'Example: UseSeparatorLine property  
'This example script has not yet been created.

'Example: UsesHelp property

'This example script has not yet been created.

'Example: UsesPalette property

'This example script has not yet been created.

'Example: UseSuperscriptReferenceNum property  
'This example script has not yet been created.

'Example: UseTwoSpacesBetweenSentences property  
'This example script has not yet been created.



'Example: UseWhen property

'This example script has not yet been created.

'Example: ValidateValue method

'This example script has not yet been created.

'Example: Valid property

'This example script has not yet been created.

'Example: Value property

'This example script has not yet been created.

```
'Example: VerificationResult property  
'This example script has not yet been created.
```

'Example: VersionID property

'This example script has not yet been created.

'Example: VersionManager property

'This example script has not yet been created.

'Example: VersionName property

'This example script has not yet been created.



'Example: VersionRemarks property

'This example script has not yet been created.

'Example: Versions property

'This example script has not yet been created.

'Example: VertAlign property

'This example script has not yet been created.

'Example: VerticalSplitWindow property  
'This example script has not yet been created.

'Example: VertRuler property

'This example script has not yet been created.

'Example: VertScrollBarVisible property

'This example script has not yet been created.

'Example: ViewLevel property

'This example script has not yet been created.

'Example: ViewType property

'This example script has not yet been created.



```
'Example: Visible property
' Toggle the statusbar
If .ApplicationWindow.StatusBar.Visible = True Then
    .ApplicationWindow.StatusBar.HideStatusBar
Else
    .ApplicationWindow.StatusBar.ShowStatusBar
End If
```

'Example: WasDeletedInRevMarkMode property  
'This example script has not yet been created.

'Example: WasInsertedInRevMarkMode property  
'This example script has not yet been created.

'Example: WasPasted property

'This example script has not yet been created.

'Example: Weight property

'This example script has not yet been created.

'Example: Where property

'This example script has not yet been created.

'Example: WidowOrphan property

'This example script has not yet been created.

'Example: WidthAbove property

'This example script has not yet been created.



'Example: WidthBelow property

'This example script has not yet been created.

'Example: WidthInTwips property

'This example script has not yet been created.

'Example: WidthOfColumn property

'This example script has not yet been created.

'Example: Width property

'This example script has not yet been created.

'Example: WindowId property

'This example script has not yet been created.

'Example: WindowsName property

'This example script has not yet been created.

'Example: WinViewPrefs property  
'This example script has not yet been created.

'Example: WithComments property

'This example script has not yet been created.



'Example: WMCommand event

'This example script has not yet been created.

```
'Example: WmCommand method
' This example display the Word Pro Welcome dialog using the WmCommand
' function. The list of defined values for the WmCommand function are
' located in the "wpbitmsk.lss"file.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
Const LwpMenuMfWelcome = &H7AD
.WmCommand(LwpMenuMfWelcome)
```

'Example: WordDoubleUnderline property

'This example script has not yet been created.

```
'Example: WordUnderline method
' This example toggles the word underline attribute of the selected text.
' RUNTIME DEPENDENCIES: You must have a document open and some text selected ' for
this script to work.

.WordUnderline
MessageBox "Click OK undo word underline change.",MB_OK,"Example Script"
.WordUnderline
```

'Example: WordUnderline property

'This example script has not yet been created.

'Example: WorkingType property

'This example script has not yet been created.

```
'Example: WPDataSets property
' This example creates a dataset named 'ExampleDataSet' off of the active
' document. The 'FirstName' and 'LastName' items are created and filled with data.
' Finally the values for the dataset items are printed to the Script Editor
' Output panel. Since no dataset item named 'Address' was defined, the default
' dataset value will be printed in the last statement.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim DataSetName as String
Dim DefaultValue as String
Dim DataSet As WPDataSetCollection
Set DataSet = .ActiveDocument.WPDataSets

DataSetName = "ExampleDataSet"
DefaultValue = "Default"

DataSet(DataSetName).SetData "FirstName","John"
DataSet(DataSetName).SetData "LastName","Doe"

Print DataSet(DataSetName).GetData("FirstName",DefaultValue)
Print DataSet(DataSetName).GetData("LastName",DefaultValue)
Print DataSet(DataSetName).GetData("Address",DefaultValue)
```

'Example: WrapType property

'This example script has not yet been created.



'Example: WriteProfileString method

'This example has not yet been created.

```
'Example: Write method
' This example creates a bag in the active division and then writes some data
' to the bag. The data from the created bag is read and printed to the Lotus
' Script Output panel. Next, data from all bags in the Bag Collection is
' printed.
' RUNTIME DEPENDENCIES: You must have a document open for this script to work.
```

```
Dim BagName As String
Dim MyBag As Bag
Dim BagData As String
```

```
BagData = "This is data for the bag."
LenBagData = Len(BagData)
```

```
BagName = .Division.Foundry.Create($LwpFoundryCreateTypeBag)
Set MyBag = .Division.Foundry.Bags.Item(BagName)
```

```
Stat = MyBag.Write(BagData, LenBagData)
```

```
If Stat = True Then
    Print "BagData= " & MyBag.Read(LenBagData)
End If
```

```
Forall ThisBag In .Division.Foundry.Bags
    ThisBag.Reset
    Print "Name = " ThisBag.Name
    Print "Length = " ThisBag.Length
    Print ThisBag.Read(ThisBag.Length)
End Forall
```

'Example: XOffset property

'This example script has not yet been created.

'Example: XPosition property

'This example script has not yet been created.

'Example: YOffset property

'This example script has not yet been created.

'Example: YPosition property

'This example script has not yet been created.

'Example: Zero property

'This example script has not yet been created.

'Example: ZipCode property

'This example script has not yet been created.



## Word Pro LotusScript A-Z

A  
B  
C  
D  
E  
F  
G  
H  
I  
J  
K  
L  
M  
N  
O  
P  
Q  
R  
S  
T  
U  
V  
W  
X  
Y  
Z

### A

[Abandon method](#)  
[Abilities property](#)  
[AbsoluteOn property](#)  
[AbsoluteTextOrientation property](#)  
[AbsoluteXPos property](#)  
[AbsoluteYPos property](#)  
[Accelerators property](#)  
[Accelerators class](#)  
[AccessRights property](#)  
[Action property](#)  
[Activate method](#)  
[ActivateAs method](#)  
[Active property](#)  
[ActiveDocument property](#)  
[ActiveDocWindow property](#)  
[ActualEnumName property](#)  
[ActualName property](#)  
[Add method](#)  
[AddAccelerators method](#)  
[AddACondition method](#)  
[AddBookmark method](#)  
[AddChildToLayout method](#)  
[AddDdeLink method](#)  
[AddDivision method](#)  
[AddDivisionToPrint method](#)  
[AddEditorManager method](#)  
[AddField method](#)  
[AddIcon method](#)  
[AddIndexEntry method](#)

[AddLayoutOverride method](#)  
[AddNewSectionTabs method](#)  
[AddOutlineSequenceItem method](#)  
[AddPopupGraphicItem method](#)  
[AddPopupPointSizeItem method](#)  
[AddPopupTextItem method](#)  
[Address1 property](#)  
[Address2 property](#)  
[AddSectionTabs method](#)  
[AddSmartCorrect method](#)  
[AddStringToList method](#)  
[AddTOCEntry method](#)  
[AddVerbMenu method](#)  
[AdjustShade method](#)  
[Adopt method](#)  
[AdviseOnRename method](#)  
[AdviseOnSave method](#)  
[Afid property](#)  
[AfidClassName property](#)  
[Align property](#)  
[Alignment property](#)  
[Alignment class](#)  
[AlignmentChar property](#)  
[AlignmentType property](#)  
[AlignStyleName property](#)  
[All property](#)  
[AllBorders property](#)  
[AllowAlternateVerification property](#)  
[AllowListEdit property](#)  
[AllowListMultiValues property](#)  
[AlternateName property](#)  
[Always property](#)  
[Amikake property](#)  
[Amikake class](#)  
[AmikakeName property](#)  
[AmikakeType property](#)  
[Amount property](#)  
[AmountOfSpaceAbove property](#)  
[AmountOfSpaceAboveLine property](#)  
[AmountOfSpaceBelow property](#)  
[AmtTether property](#)  
[AmtToRotateContent property](#)  
[AmtToTetherFrom property](#)  
[Anchor method](#)  
[AnswerMsgBox method](#)  
[AnyEdits method](#)  
[AnyNumber property](#)  
[AnyOLEDELinks property](#)  
[AppendMacro method](#)  
[AppFoundry property](#)  
[Application class](#)  
[Application property](#)  
[ApplicationWindow property](#)  
[ApplicationWindow class](#)  
[ApplyAdjectivePos property](#)  
[ApplyAdjectNounPart property](#)  
[ApplyAgreementWithHereThere property](#)  
[ApplyAnglicisms property](#)

[ApplyArchaicExpressions property](#)  
[ApplyArticleAgreement property](#)  
[ApplyBadComparatives property](#)  
[ApplyBadInflection property](#)  
[ApplyBadNoun property](#)  
[ApplyBadNounGender property](#)  
[ApplyBadPlural property](#)  
[ApplyBadPrepositions property](#)  
[ApplyBelgianExpression property](#)  
[ApplyBorrowedForeign property](#)  
[ApplyBureuaJargon property](#)  
[ApplyCalque property](#)  
[ApplyCapitalizationCheck property](#)  
[ApplyClauseErrors property](#)  
[ApplyCliches property](#)  
[ApplyColloquialExpression property](#)  
[ApplyCommonlyConfusedWords property](#)  
[ApplyCommonMisspell property](#)  
[ApplyComplexWords property](#)  
[ApplyConfusedEasy property](#)  
[ApplyConfusedEnglish property](#)  
[ApplyConfusedHard property](#)  
[ApplyConfusedMedium property](#)  
[ApplyConfusedVerb property](#)  
[ApplyConsecutiveNouns property](#)  
[ApplyContractions property](#)  
[ApplyDerogatory property](#)  
[ApplyDifferentPrep property](#)  
[ApplyDoubleNegative property](#)  
[ApplyDoublePlural property](#)  
[ApplyDoubleWordCheck property](#)  
[ApplyElision property](#)  
[ApplyEnglishDerived property](#)  
[ApplyEnglishWords property](#)  
[ApplyExotic property](#)  
[ApplyExtraPrepositionCheck property](#)  
[ApplyFalseFriend property](#)  
[ApplyFemaleOccupation property](#)  
[ApplyFixedExpression property](#)  
[ApplyForeignWord property](#)  
[ApplyFormalTerms property](#)  
[ApplyFormatErrors property](#)  
[ApplyGallicisms property](#)  
[ApplyGenderExpressions property](#)  
[ApplyGermanisms property](#)  
[ApplyHomoGraphs property](#)  
[ApplyHomonyms property](#)  
[ApplyHomonymsEasy property](#)  
[ApplyHomonymsHard property](#)  
[ApplyHomoPhone1 property](#)  
[ApplyHomoPhone2 property](#)  
[ApplyHomoPhones property](#)  
[ApplyIncorrectPlural property](#)  
[ApplyInformalExpressions property](#)  
[ApplyJargonWords property](#)  
[ApplykSplitInfinitives property](#)  
[ApplyLowercaseAdjective property](#)  
[ApplyLowercaseColor property](#)

[ApplyLowercaseNumbers property](#)  
[ApplyLowercasePhrases property](#)  
[ApplyLowercasePronouns property](#)  
[ApplyMassVsCount property](#)  
[ApplyMisspelledExpressions property](#)  
[ApplyMisspelledForeignExpressions property](#)  
[ApplyMisspelledItalian property](#)  
[ApplyMisspelledWords property](#)  
[ApplyMisusedWords property](#)  
[ApplyNonStandardExpression property](#)  
[ApplyNonStandardModifiers property](#)  
[ApplyNoudModifierOrderCheck property](#)  
[ApplyNounConsistency property](#)  
[ApplyNounPhraseAgree property](#)  
[ApplyNSAdjective property](#)  
[ApplyNSClause property](#)  
[ApplyNSCompare property](#)  
[ApplyNSContract property](#)  
[ApplyNSGeography property](#)  
[ApplyNSInflection property](#)  
[ApplyNSNegation property](#)  
[ApplyNSPrep property](#)  
[ApplyNSPronoun property](#)  
[ApplyNSSpell property](#)  
[ApplyNSUsage property](#)  
[ApplyNSVerbForm property](#)  
[ApplyOpenClosedSpelling property](#)  
[ApplyOpenUsage property](#)  
[ApplyOverusedPhrases property](#)  
[ApplyPassiveVerbErrors property](#)  
[ApplyPostClitAgree property](#)  
[ApplyPrepExpression property](#)  
[ApplyPrepositionalPhrases property](#)  
[ApplyPretentiousWords property](#)  
[ApplyPronounErrors property](#)  
[ApplyPunctuationErrors property](#)  
[ApplyRedundantExpressions property](#)  
[ApplyRegionalExpression property](#)  
[ApplyRelatedWord property](#)  
[ApplySensitiveExp property](#)  
[ApplySexistExpressions property](#)  
[ApplySpellStandard property](#)  
[ApplyStockPhrase property](#)  
[ApplyStyleParameters property](#)  
[ApplySubjectVerbAgreement property](#)  
[ApplySwedishGender property](#)  
[ApplySwedishNegation property](#)  
[ApplySwedishUsage property](#)  
[ApplyTrite property](#)  
[ApplyTwoGender property](#)  
[ApplyTypicalMisspell property](#)  
[ApplyUnGrammaticalExpressions property](#)  
[ApplyVagueQuantifiers property](#)  
[ApplyVerbGroupConsistency property](#)  
[ApplyWeakModifiers property](#)  
[ApplyWordChoice property](#)  
[ApplyWordCompoundingCheck property](#)  
[ApplyWordConfusion property](#)

[ApplyWordGender property](#)  
[ApplyWordParts property](#)  
[ApplyWordyPhraseCheck property](#)  
[AppViewPrefs property](#)  
[AppViewPrefs class](#)  
[AreDisabledIconsGrayed property](#)  
[Ascent property](#)  
[AsciiCodePage property](#)  
[AsciiCRLFType property](#)  
[AtBeginning property](#)  
[AtBeginningOfLine property](#)  
[AtBeginningOfObject property](#)  
[AtBeginningOfParagraph property](#)  
[AtBeginningOfStream property](#)  
[AtBeginningOfWord property](#)  
[AtEnd property](#)  
[AtEndOfLine property](#)  
[AtEndOfObject property](#)  
[AtEndOfParagraph property](#)  
[AtEndOfStream property](#)  
[AtEndOfWord property](#)  
[Attributes property](#)  
[Attributes class](#)  
[AttrStyleName property](#)  
[AuthorName property](#)  
[AutoBackup property](#)  
[AutoHyphenate property](#)  
[AutomaticLink property](#)  
[AutoRunMacro property](#)  
[AutoRunMacro class](#)  
[AutoSave property](#)  
[AutoSaveMinutes property](#)  
[AutoVersion property](#)

## **B**

[BackColor property](#)  
[BackColorIndex property](#)  
[Background property](#)  
[Background class](#)  
[BackgroundPrintingOn property](#)  
[BackgroundSpellingOn property](#)  
[Backspace method](#)  
[BackupPath property](#)  
[BackupPaths property](#)  
[Backward method](#)  
[Bag class](#)  
[BagCollection class](#)  
[Bags property](#)  
[BaseCollection class](#)  
[BaseContainer class](#)  
[BaseLineOffset property](#)  
[BaselineOffset property](#)  
[BaseObject class](#)  
[BaseTable property](#)  
[BaseTable class](#)  
[BaseTables property](#)  
[BaseURL property](#)

[BeginChange method](#)  
[BeginCustomLines method](#)  
[BinName property](#)  
[BinNameFromNumber method](#)  
[BinNames property](#)  
[BinNumber property](#)  
[Bisect method](#)  
[BlockPaint property](#)  
[Blue property](#)  
[BodyOnly property](#)  
[Bold property](#)  
[Bold method](#)  
[BookletPrinting property](#)  
[Bookmark class](#)  
[BookmarkCollection class](#)  
[BookmarkManager property](#)  
[BookmarkManager class](#)  
[Bookmarks property](#)  
[BookmarksByMarkerName property](#)  
[Border class](#)  
[BorderLines property](#)  
[BorderLines class](#)  
[BorderOffset property](#)  
[BorderStyleName property](#)  
[BottomBorder property](#)  
[BottomExternalMargin property](#)  
[BottomIntArea property](#)  
[BreakLink method](#)  
[Breaks property](#)  
[Breaks class](#)  
[BreaksStyleName property](#)  
[BringFrameToFront method](#)  
[BringFrameToFrontOne method](#)  
[Bullet class](#)  
[Bullet property](#)  
[BulletFonts property](#)  
[BulletStyleName property](#)

## C

[CalcSmartLevels method](#)  
[CalculateSmartLevels method](#)  
[CanCreatePreviewBitmap property](#)  
[CanEditProperty property](#)  
[CanEmbed property](#)  
[CanHaveFootnotes method](#)  
[CanWePrint property](#)  
[Caption property](#)  
[Cascade method](#)  
[CascadeWindow method](#)  
[Case property](#)  
[Cell property](#)  
[CellCollection class](#)  
[CellContainer class](#)  
[CellEngine class](#)  
[CellEngine property](#)  
[CellEngines property](#)  
[CellGroupLayout class](#)

[CellLayout method](#)  
[CellLayout class](#)  
[CellLayout property](#)  
[CellLayoutCollection class](#)  
[CellLayouts property](#)  
[CellLayoutStyles property](#)  
[CellRevert method](#)  
[CellStyleName property](#)  
[Center property](#)  
[CenteredHorz property](#)  
[CenteredVert property](#)  
[ChangeAllEditsToEditor method](#)  
[Changed property](#)  
[ChangeKeyboardToLanguage property](#)  
[ChangeSmartMaster method](#)  
[ChangeTextToMatchkeyboard property](#)  
[CharacterBorder property](#)  
[CharacterBorder class](#)  
[CharacterBorderName property](#)  
[CharacterSet property](#)  
[CharacterSet class](#)  
[CharacterSetName property](#)  
[CharacterStyle property](#)  
[CharacterStyle class](#)  
[CharacterStyleCollection class](#)  
[CharacterStyleName property](#)  
[CharacterStyles property](#)  
[CharSet property](#)  
[Checked property](#)  
[CheckFieldEntries method](#)  
[CheckForScriptEnumError property](#)  
[CheckForScriptPropertyError property](#)  
[ChgLineStyle method](#)  
[ChildLayouts property](#)  
[CityState property](#)  
[ClassId property](#)  
[ClassName property](#)  
[CleanScreenMode property](#)  
[CleanScreenOnStartUp property](#)  
[Clear method](#)  
[ClearAll method](#)  
[ClearDivisionList method](#)  
[ClearInternalSpellInfo method](#)  
[ClearParaRevisionTags method](#)  
[ClearPopupData method](#)  
[ClearSplits method](#)  
[ClearTempFoundry method](#)  
[ClearUpdate method](#)  
[ClickHere class](#)  
[ClickHereCollection class](#)  
[ClickHerePrompts property](#)  
[ClickHeres property](#)  
[ClientHeight property](#)  
[ClientWidth property](#)  
[ClientWndHeight property](#)  
[ClientWndWidth property](#)  
[Close method](#)  
[CloseAll method](#)

[CloseDocMacroName property](#)  
[CloseDocWindow method](#)  
[CloseMergeDataFile method](#)  
[CloseObject method](#)  
[CodePage property](#)  
[Collapsible property](#)  
[Collate property](#)  
[Color property](#)  
[Color class](#)  
[ColorOverride property](#)  
[Column class](#)  
[ColumnBalance property](#)  
[ColumnGap property](#)  
[ColumnGroupLayout class](#)  
[ColumnLayouts property](#)  
[ColumnNumber property](#)  
[ColumnWidth property](#)  
[CombineDivisions method](#)  
[CombineSections method](#)  
[CommentColor property](#)  
[Company property](#)  
[CompareFiles method](#)  
[Condition property](#)  
[ConditionType property](#)  
[Configure method](#)  
[Connect method](#)  
[ConnectCells method](#)  
[ConnectContainer method](#)  
[ConnectedLayout class](#)  
[ConnectedLayoutCollection class](#)  
[ConnectedLayouts property](#)  
[ConnectRows method](#)  
[ConnectSectionTabs method](#)  
[ConsistencyCheck method](#)  
[Container property](#)  
[Content class](#)  
[Content property](#)  
[ContentCollection class](#)  
[ContentHeight property](#)  
[ContentLayouts property](#)  
[ContentName property](#)  
[Contents property](#)  
[ContentStyleName property](#)  
[ContentType property](#)  
[ContentWidth property](#)  
[ContextMenuOptions property](#)  
[ContextMenuOptions class](#)  
[ContinuedFromAlignment property](#)  
[ContinuedFromMessage property](#)  
[ContinuedFromStory property](#)  
[ContinuedOnAlignment property](#)  
[ContinuedOnMessage property](#)  
[ContinuedOnStory property](#)  
[Contract method](#)  
[ContractOutlineLevel method](#)  
[ConvertOnNew property](#)  
[ConvertTo method](#)  
[ConvertToClass method](#)



[Copies property](#)  
[Copy method](#)  
[CopyItem method](#)  
[CopyMeaning method](#)  
[CopySelection method](#)  
[Count property](#)  
[CountBlankLines property](#)  
[Create method](#)  
[Created event](#)  
[CreateDataFile method](#)  
[CreateDivision method](#)  
[CreateDocument method](#)  
[CreateDropCap method](#)  
[CreateEmptyList method](#)  
[CreateExternalDivision method](#)  
[CreateFrame method](#)  
[CreateFromBitmap method](#)  
[CreateFromClipBrd method](#)  
[CreateFromDataObject method](#)  
[CreateFromMetafile method](#)  
[CreateGlossary method](#)  
[CreateGlossaryEntry method](#)  
[CreateGraphic method](#)  
[CreateLayer method](#)  
[CreateNew method](#)  
[CreateNewButton method](#)  
[CreateOleEmbeddedFile method](#)  
[CreateOleLinkedFile method](#)  
[CreateOleNew method](#)  
[CreateParallelColumns method](#)  
[CreatePreviewBitmap property](#)  
[CreateRemark method](#)  
[CreateTable method](#)  
[CreateVersion method](#)  
[CreationDateString property](#)  
[CreationTimeString property](#)  
[CreationTimeValue property](#)  
[Crop property](#)  
[CurrentCell property](#)  
[CurrentColumn property](#)  
[CurrentEditor property](#)  
[CurrentLanguage property](#)  
[CurrentRow property](#)  
[CurrentRunningScriptName property](#)  
[CurrentRunningScriptPath property](#)  
[CurrentVersion property](#)  
[CustomLength property](#)  
[CustomViewLevel property](#)  
[CutSelection method](#)

## D

[DarkMode method](#)  
[DataFileFieldNames property](#)  
[DataFileName property](#)  
[DataFormat property](#)  
[DataNames property](#)  
[DataObjectGetData method](#)

[DataObjectGetDataHere method](#)  
[DateCreatedValue property](#)  
[DateRevisedValue property](#)  
[DbfUnderline method](#)  
[DdeEnabled property](#)  
[DdeLink class](#)  
[DdeLinkCollection class](#)  
[DdeLinkManager property](#)  
[DdeLinkManager class](#)  
[DdeLinks property](#)  
[DdeLinksFromMarker property](#)  
[DdeOutboundInfo property](#)  
[DebugVariable property](#)  
[DefaultBinName property](#)  
[DefaultCellStyleDescription property](#)  
[DefaultColumnStyleName property](#)  
[DefaultDropCapStyleDescription property](#)  
[DefaultFilePath property](#)  
[DefaultFrameStyleDescription property](#)  
[DefaultLatinFont property](#)  
[DefaultLeftColumnStyleName property](#)  
[DefaultNewCategory property](#)  
[DefaultNonLatinFont property](#)  
[DefaultPageHeight property](#)  
[DefaultPageStyleDescription property](#)  
[DefaultPageWidth property](#)  
[DefaultPitch property](#)  
[DefaultRightColumnStyleName property](#)  
[DefaultTableStyleDescription property](#)  
[DefaultTextStyleDescription property](#)  
[DefCellStyleName property](#)  
[DefColWidth property](#)  
[Definition property](#)  
[DefRowHeight property](#)  
[DelayedGreeting property](#)  
[DeleteBag method](#)  
[DeleteButton method](#)  
[DeleteChars method](#)  
[DeleteContainer method](#)  
[DeleteContent method](#)  
[DeleteContents method](#)  
[DeleteDivision method](#)  
[DeleteField method](#)  
[DeleteFont property](#)  
[DeleteItem method](#)  
[DeleteItemByPosition method](#)  
[DeleteKey method](#)  
[DeleteLayout method](#)  
[DeleteMacroAccelerator method](#)  
[DeleteMarker method](#)  
[DeleteParallelColumns method](#)  
[DeleteSection method](#)  
[DeleteSmartCorrect method](#)  
[DeleteTab method](#)  
[DeleteTable method](#)  
[DeleteVersion method](#)  
[DemandLoad property](#)  
[Demote method](#)

[DemoteOutlineLevel method](#)  
[Descent property](#)  
[Description property](#)  
[DescriptionFileName property](#)  
[Deselect method](#)  
[DeselectRuler method](#)  
[Destroy method](#)  
[DestroyDocWindow method](#)  
[DiagonalLines property](#)  
[DirectionDown property](#)  
[DirectionLeft property](#)  
[DirectionRight property](#)  
[DirectionUp property](#)  
[DirectiveColor property](#)  
[DisableClickHeres property](#)  
[DisableConsistencyCheck property](#)  
[DisableExportToNotes property](#)  
[DisableVersionReview property](#)  
[Disconnect method](#)  
[DisconnectCells method](#)  
[DisplayAsIcon property](#)  
[DistanceFromLeftMargin property](#)  
[DivideText method](#)  
[Division property](#)  
[Division class](#)  
[DivisionCollection class](#)  
[DivisionInfo class](#)  
[DivisionInfo property](#)  
[DivisionName property](#)  
[DivisionNames property](#)  
[DivisionOptions property](#)  
[DivisionOptions class](#)  
[Divisions property](#)  
[DivisionsRequired property](#)  
[DocControl property](#)  
[DocControl class](#)  
[DocControlRestrictedToEditor property](#)  
[DocInfo class](#)  
[DocInfo property](#)  
[DocInfoField class](#)  
[DocInfoFieldCollection class](#)  
[DocInfoFieldManager class](#)  
[DocOptions property](#)  
[DocPath property](#)  
[DocSize property](#)  
[Document class](#)  
[Document property](#)  
[DocumentClose event](#)  
[DocumentClosed event](#)  
[DocumentCreate event](#)  
[DocumentCreated event](#)  
[DocumentExport event](#)  
[DocumentExported event](#)  
[DocumentImport event](#)  
[DocumentImported event](#)  
[DocumentInsert event](#)  
[DocumentInserted event](#)  
[DocumentLevel property](#)

[DocumentOpen event](#)  
[DocumentOpened event](#)  
[DocumentPaths property](#)  
[DocumentPrint event](#)  
[DocumentPrinted event](#)  
[Documents class](#)  
[Documents property](#)  
[DocumentSave event](#)  
[DocumentSaveAs event](#)  
[DocumentSaved event](#)  
[DocumentSavedAs event](#)  
[DocVersionID property](#)  
[DocWindow class](#)  
[DocWindowCollection class](#)  
[DocWindows property](#)  
[DoesMarkerNameMatch method](#)  
[DoInitialCaps property](#)  
[DoneWithRightMouseMenu method](#)  
[DoSmartQuotes property](#)  
[DoubleUnderline property](#)  
[DragDropOn property](#)  
[DriverName property](#)  
[DropCapContainer class](#)  
[DropCapLayout class](#)  
[DropCapLayoutCollection class](#)  
[DropCapPosition property](#)  
[DropCaps property](#)  
[DropCapStyleName property](#)  
[DropCapStyles property](#)  
[DuplexType property](#)

## E

[Editable property](#)  
[EditClickHereLink method](#)  
[EditLinkInfo method](#)  
[Editor class](#)  
[EditorCollection class](#)  
[EditorInitials property](#)  
[EditorManager property](#)  
[EditorManager class](#)  
[EditorName property](#)  
[Editors property](#)  
[EditorVerificationType property](#)  
[EffectiveColumnWidth property](#)  
[EMail property](#)  
[Embed method](#)  
[Embedded property](#)  
[EmbedFonts property](#)  
[EmbedFormula method](#)  
[Enabled property](#)  
[EncryptPassword property](#)  
[EncryptPassword2 property](#)  
[EndChange method](#)  
[EndCustomLines method](#)  
[Ending method](#)  
[EndingColOfSelection property](#)  
[EndingRowOfSelection property](#)

[EndnoteDivisionGroupNum property](#)  
[EndnoteDivisionGroupNum class](#)  
[EndnoteDivisionNum property](#)  
[EndnoteDivisionNum class](#)  
[EndnoteDocNum property](#)  
[EndnoteDocNum class](#)  
[EndnoteLayout class](#)  
[EndnoteLayoutCollection class](#)  
[Endnotes property](#)  
[EndPrinting method](#)  
[EnterClickHere event](#)  
[EnterLayout event](#)  
[EnumerateChartLinks method](#)  
[EnumerateTerm method](#)  
[Enumeration property](#)  
[EnvelopeBarCode method](#)  
[EnvelopePrint method](#)  
[Epoch property](#)  
[EqnFontHeight property](#)  
[ErrorColor property](#)  
[ExchangItem method](#)  
[ExcludeRectBottom property](#)  
[ExcludeRectLeft property](#)  
[ExcludeRectRight property](#)  
[ExcludeRectTop property](#)  
[ExitClickHere event](#)  
[Expand method](#)  
[Expandable property](#)  
[ExpandOutline method](#)  
[ExpandOutlineLevel method](#)  
[ExportAllAsNotesFX method](#)  
[ExportAsNotesFX method](#)  
[ExportToNotesFX property](#)  
[ExternalFileID property](#)  
[ExternalFileName property](#)  
[ExternallyControlledUndo property](#)  
[ExternalName property](#)  
[ExternalType property](#)  
[ExtractText method](#)

## F

[FaceName property](#)  
[FaceNames property](#)  
[FaceStyleName property](#)  
[FastFormat method](#)  
[FastFormatType property](#)  
[FaxNumber property](#)  
[FieldDelimiter property](#)  
[FieldDelimiterText property](#)  
[FieldManager property](#)  
[FieldNumber property](#)  
[Fields property](#)  
[FieldType property](#)  
[FileFormat property](#)  
[FileProtectionType property](#)  
[FilesToCompare property](#)  
[FileType property](#)

[FillerPageText property](#)  
[FillStyle property](#)  
[Filter property](#)  
[Filter class](#)  
[FilterHelper class](#)  
[FilterHelper property](#)  
[FilterName property](#)  
[Find method](#)  
[FindAndReplace property](#)  
[FindAndReplace class](#)  
[FindCellLayout method](#)  
[FindClass method](#)  
[FindExactCase property](#)  
[FindFont property](#)  
[FindForwardDirection property](#)  
[FindParent method](#)  
[FindString property](#)  
[FindStyleName property](#)  
[FindTerm method](#)  
[FindWithProperties property](#)  
[FinishedSpellChecking property](#)  
[First property](#)  
[FirstChild property](#)  
[FirstDivision property](#)  
[FirstName property](#)  
[FirstPage property](#)  
[FirstSpellString property](#)  
[FitType property](#)  
[FixAcronymns property](#)  
[FixBullets property](#)  
[FixMargins property](#)  
[Font property](#)  
[Font class](#)  
[FontColor property](#)  
[FontMatching property](#)  
[FontMetrics property](#)  
[FontMetrics class](#)  
[FontName property](#)  
[FontSize property](#)  
[FontStyleName property](#)  
[FontUnitName property](#)  
[FontUnits property](#)  
[Footer class](#)  
[Footer property](#)  
[FooterLayout class](#)  
[FooterLayoutCollection class](#)  
[Footers property](#)  
[FooterStyleName property](#)  
[FooterStyles property](#)  
[Footnote class](#)  
[FootnoteCollection class](#)  
[FootnoteContSep property](#)  
[FootnoteContSep class](#)  
[FootnoteLayout class](#)  
[FootnoteLayoutCollection class](#)  
[FootnoteLayouts property](#)  
[FootnoteNumbering property](#)  
[FootnoteNumbering class](#)

[FootnoteNumOpt class](#)  
[FootnoteOptions property](#)  
[FootnoteOptions class](#)  
[Footnotes property](#)  
[FootnoteSeparator property](#)  
[FootnoteSeparator class](#)  
[FootnoteSepOpt class](#)  
[FootnoteTable class](#)  
[ForceDocToLoad method](#)  
[ForeColorIndex property](#)  
[Format property](#)  
[FormatCheckPref class](#)  
[FormatCheckPreferences property](#)  
[FormatPreferences class](#)  
[FormatType property](#)  
[Formula class](#)  
[Formula property](#)  
[Forward method](#)  
[Foundry property](#)  
[Foundry class](#)  
[Frame property](#)  
[FrameContainer class](#)  
[FrameGroupLayout class](#)  
[FrameLayout class](#)  
[FrameLayoutCollection class](#)  
[FrameRevert method](#)  
[Frames property](#)  
[FrameStyleName property](#)  
[FrameStyles property](#)  
[FreeMenus property](#)  
[FullName property](#)  
[FXGetNotesString method](#)  
[FXGetNotesWriteHandle method](#)  
[FXSetNotesString method](#)

## **G**

[GapBetweenPanels property](#)  
[Get method](#)  
[GetActiveList method](#)  
[GetAfidHelpFileName property](#)  
[GetAfidHelpInfo property](#)  
[GetArrayProp method](#)  
[GetAspectMetafilePict method](#)  
[GetButtonId property](#)  
[GetButtonType property](#)  
[GetContents method](#)  
[GetConversationHandle property](#)  
[GetCopyFormatCategories method](#)  
[GetCount method](#)  
[GetCurrentMarkerName method](#)  
[GetData method](#)  
[GetDefaultPageSize method](#)  
[GetDisplayableFilterName method](#)  
[GetEnumerator method](#)  
[GetEnvelopeDefaults method](#)  
[GetFileDescription method](#)  
[GetFilterExtension property](#)

[GetFilterExtForDialogBox property](#)  
[GetFilterId property](#)  
[GetFormatName property](#)  
[GetFormula method](#)  
[GetHomeDirectory property](#)  
[GetInternetFile method](#)  
[GetItemName property](#)  
[GetLastUsedFilter method](#)  
[GetLineMix method](#)  
[GetLineStyle method](#)  
[GetLinkDisplayNameFileLength method](#)  
[GetLinkName method](#)  
[GetLinkSourceName method](#)  
[GetListName method](#)  
[GetMarkedText method](#)  
[GetMarkerName method](#)  
[GetMisspelledWord method](#)  
[GetNamedProperty method](#)  
[GetNameFromPage method](#)  
[GetObjectList method](#)  
[GetOne method](#)  
[GetPageRange method](#)  
[GetParagraphNumber method](#)  
[GetParaNumber method](#)  
[GetPasteFormatCategories method](#)  
[GetPosition method](#)  
[GetPowerFieldValue method](#)  
[GetPrinterInfo method](#)  
[GetProfileString method](#)  
[GetRedoWhatDesc property](#)  
[GetRGB method](#)  
[GetRightContextMenu method](#)  
[GetServerName property](#)  
[GetSource method](#)  
[GetSpellStatus method](#)  
[GetSpellUserDictStatus method](#)  
[GetStandardButtonId method](#)  
[GetStatus property](#)  
[GetString method](#)  
[GetText method](#)  
[GetTOCProperties method](#)  
[GetTopicName property](#)  
[GetUndoWhatDesc property](#)  
[GetUniqueName method](#)  
[GetUserClassNameFull method](#)  
[GetUserClassNameShort method](#)  
[GetValue property](#)  
[GetWordMisspelled method](#)  
[Glossarize method](#)  
[Glossary class](#)  
[GlossaryCollection class](#)  
[GlossaryDataFileName property](#)  
[GlossaryDataFiles property](#)  
[GlossaryDataPaths property](#)  
[GlossaryInsert method](#)  
[GlossaryOpen method](#)  
[GlossaryPath property](#)  
[Glossarys property](#)



[GoTo method](#)  
[GoToBookmark method](#)  
[GoToClickHere method](#)  
[GoToContainer method](#)  
[GoToIndexLastItemAccessed property](#)  
[GoToItemName property](#)  
[GoToLayout method](#)  
[GotoNextParallelColumn method](#)  
[GoToObject method](#)  
[GoToPage method](#)  
[GotoPageLoadInBackground property](#)  
[GoToPageNumber property](#)  
[GoToSection method](#)  
[GoToSelection property](#)  
[GoToTableCell method](#)  
[Grammar class](#)  
[GrammarFormalityLevel property](#)  
[GrammarOptions property](#)  
[GrammarProofLevel property](#)  
[GrammarSetName property](#)  
[Graphic class](#)  
[Graphic property](#)  
[GraphicCollection class](#)  
[GraphicExports property](#)  
[GraphicImports property](#)  
[GraphicOleObject property](#)  
[GraphicOleObject class](#)  
[GraphicOleObjectCollection class](#)  
[GraphicPath property](#)  
[GraphicPaths property](#)  
[Graphics property](#)  
[Green property](#)  
[Greeting property](#)  
[GridDistance property](#)  
[GridType property](#)  
[GroupDivision method](#)  
[GroupLayout class](#)  
[GroupLayoutCollection class](#)  
[Groups property](#)  
[Gutter class](#)  
[Gutter property](#)

## H

[HandsOffStorage method](#)  
[Hang property](#)  
[HasContents property](#)  
[HasFocus property](#)  
[HasIndex property](#)  
[HasLocalTabs property](#)  
[HasNamedProperty method](#)  
[HasProperty method](#)  
[HasTabs property](#)  
[HasTOC property](#)  
[Header class](#)  
[Header property](#)  
[HeaderLayout class](#)  
[HeaderLayoutCollection class](#)

[Headers property](#)  
[HeaderStyleName property](#)  
[HeaderStyles property](#)  
[Heading property](#)  
[Height property](#)  
[Help method](#)  
[HelpText property](#)  
[HiddenMode property](#)  
[Hide method](#)  
[HideCaretAndSelection method](#)  
[HideFormula property](#)  
[HideIconBar method](#)  
[HideIconBars method](#)  
[HideOutlineLevels property](#)  
[HideStatusBar method](#)  
[HighLightMode property](#)  
[HighlightToggle method](#)  
[HiLiteColor property](#)  
[Hit method](#)  
[HorizontalSplitWindow property](#)  
[HorizScrollBarVisible property](#)  
[HorzRuler property](#)  
[HourGlass method](#)  
[Hwnd property](#)  
[HyphenateLastWordInColumnOrPage property](#)  
[HyphenateLastWordInPara property](#)  
[HyphenationOptions class](#)  
[HyphenationOptions property](#)  
[HyphZoneAfter property](#)  
[HyphZoneBefore property](#)

I

[IconBar class](#)  
[IconBarCollection class](#)  
[IconBarManager class](#)  
[IconBarManager property](#)  
[IconBarPositionState property](#)  
[IconBars property](#)  
[IconBarSets property](#)  
[IconHelpText property](#)  
[IconPath property](#)  
[IconPaths property](#)  
[IconScript property](#)  
[IconSetName property](#)  
[IconShowingBeforeCleanScreen property](#)  
[IconSize property](#)  
[ID property](#)  
[IdentifierColor property](#)  
[IDispatch property](#)  
[IgnoreSoftHyphens property](#)  
[IgnoreTab property](#)  
[ImportGraphic method](#)  
[ImportInsert event](#)  
[ImportInserted event](#)  
[ImportPicture method](#)  
[ImportWatermarkGraphic method](#)  
[IncludeInitialsInNotes property](#)

[IncludeList property](#)  
[Indent class](#)  
[Indent property](#)  
[IndentFromLeft property](#)  
[IndentStyleName property](#)  
[Index class](#)  
[IndexAll method](#)  
[IndexAlphabeticSeparator property](#)  
[IndexDivision property](#)  
[IndexIndentType property](#)  
[IndexParent property](#)  
[IndexPrimaryStyleName property](#)  
[IndexRange property](#)  
[IndexSecondaryStyleName property](#)  
[IndexSection class](#)  
[IndexSection property](#)  
[IndexSeparatorStyleName property](#)  
[IndexSource property](#)  
[InDocument property](#)  
[InfoBoxSelectionText property](#)  
[InititalizeRoute property](#)  
[InitFindAndReplace method](#)  
[Initialize property](#)  
[InitialsForFilters property](#)  
[InsertBreak method](#)  
[InsertBullet method](#)  
[InsertClickHere method](#)  
[InsertClickHereLink method](#)  
[InsertColumnBreak method](#)  
[InsertDate method](#)  
[InsertDocInfo method](#)  
[InsertDocument method](#)  
[InsertField method](#)  
[InsertFont property](#)  
[InsertFootnote method](#)  
[InsertFrame method](#)  
[InsertHardSpace method](#)  
[InsertIndex method](#)  
[InsertionMode property](#)  
[InsertLink method](#)  
[InsertMarker method](#)  
[InsertNote method](#)  
[InsertNumber method](#)  
[InsertOleDivision method](#)  
[InsertOne method](#)  
[InsertPageBreak method](#)  
[InsertPageLayout method](#)  
[InsertPageNumber method](#)  
[InsertPath method](#)  
[InsertRowOrColumn method](#)  
[InsertRuby method](#)  
[InsertSection method](#)  
[InsertSpecialTab method](#)  
[InsertTab method](#)  
[InsertText method](#)  
[InsertTOC method](#)  
[Interactive property](#)  
[InternalCopy method](#)

[InternalCut method](#)  
[InternalPaste method](#)  
[InternetExtraFile method](#)  
[Interval property](#)  
[Invalidate method](#)  
[InvalidateButton method](#)  
[InvalidateWholeBar method](#)  
[IsActionOnButtonDown property](#)  
[IsActive property](#)  
[IsAdopted property](#)  
[IsAmikake property](#)  
[IsAmiProTableImport property](#)  
[IsAsciiCRLF property](#)  
[IsAsciiKeepStyle property](#)  
[IsAutoGrow property](#)  
[IsBadReference property](#)  
[IsBorder property](#)  
[IsBothSidesEqual property](#)  
[IsBottomAligned property](#)  
[IsBreakable property](#)  
[IsBubbleHelp property](#)  
[IsCaseExact method](#)  
[IsCellMenuEnabled property](#)  
[IsCentered property](#)  
[IsChanged property](#)  
[IsChangedOtherThanLinkTo property](#)  
[IsChangedSinceTimeSave property](#)  
[IsChangedToLinkTo property](#)  
[IsChartLink property](#)  
[IsChildSpannable property](#)  
[IsCollapsed property](#)  
[IsCollapsible property](#)  
[IsColumnBreakable property](#)  
[IsColumnBreakAfter property](#)  
[IsColumnBreakBefore property](#)  
[IsComplex property](#)  
[IsConnected property](#)  
[IsContentReplaceable property](#)  
[IsContents property](#)  
[IsContinuedFrom property](#)  
[IsContinuedOn property](#)  
[IsCumulative property](#)  
[IsDataNameUsed method](#)  
[IsDebug property](#)  
[IsDisableWarningMessages property](#)  
[IsDisplayMisspelled property](#)  
[IsDivisionExternal property](#)  
[IsDocLoading property](#)  
[IsDocumentInRoute property](#)  
[IsDoubleWordError property](#)  
[IsDraw property](#)  
[IsEmpty method](#)  
[IsEmpty property](#)  
[IsEmptyDoc property](#)  
[IsEnabled property](#)  
[IsEndnoteDivision property](#)  
[IsEquation property](#)  
[IsErrorChecking property](#)

[IsExpandDown property](#)  
[IsExpandLeft property](#)  
[IsExpandRight property](#)  
[IsExpandUp property](#)  
[IsExportedAsNotesFX method](#)  
[IsExportedToNotesFX property](#)  
[IsExportedToOldNotesFX property](#)  
[IsExternalFile property](#)  
[IsFilterTypePresent method](#)  
[IsFixedLength property](#)  
[IsFooter property](#)  
[IsFrameMenuEnabled property](#)  
[IsGrabBar property](#)  
[IsGrammarError property](#)  
[IsGraphicMenuEnabled property](#)  
[IsHangover property](#)  
[IsHeader property](#)  
[IsHiddenMark property](#)  
[IsHideHeaderFooter property](#)  
[IsHighlightNote property](#)  
[IsHonorProtected property](#)  
[IsHorizontalScrollBarCleanScrn property](#)  
[IsHoverHelp property](#)  
[IsIconDepressible property](#)  
[IsInBulletEditMode property](#)  
[IsIndex property](#)  
[IsInDraft property](#)  
[IsInOutline property](#)  
[IsInPageSort property](#)  
[IsIntersectSiblings property](#)  
[IsLastStop property](#)  
[IsLesser property](#)  
[IsLinked property](#)  
[IsLocal property](#)  
[IsLocked property](#)  
[IsLockedForRevisions property](#)  
[IsLotusChart property](#)  
[IsMarginSameAsParent property](#)  
[IsMarginsInColor property](#)  
[IsMarkerEqualToSelection method](#)  
[IsMarkerValid property](#)  
[IsMirrorImage property](#)  
[IsMisspelled property](#)  
[IsNotCopyable property](#)  
[IsNotCopyImage property](#)  
[IsNotGroupable property](#)  
[IsNoUICommAllowed property](#)  
[IsOnClipboard property](#)  
[IsOpen property](#)  
[IsOverridden property](#)  
[IsOverride property](#)  
[IsPageBreak property](#)  
[IsPageBreakAfter property](#)  
[IsPageBreakBefore property](#)  
[IsPageBreakMarks property](#)  
[IsPageBreakWithin property](#)  
[IsParaDemandLoad property](#)  
[IsParagraphNumberingDown property](#)

[IsParagraphParent property](#)  
[IsParallelColumnsMenuEnabled property](#)  
[IsPartOfGroup property](#)  
[IsPersistent property](#)  
[IsPointWithin method](#)  
[IsPrePrintedForm property](#)  
[IsPrintable property](#)  
[IsPrintClickHereBlocks property](#)  
[IsPrivate property](#)  
[IsPrompting property](#)  
[IsProtected property](#)  
[IsRegistered property](#)  
[IsRepeat property](#)  
[IsReplaceable property](#)  
[IsReplacement property](#)  
[IsResetParagraphNumber property](#)  
[IsRevision property](#)  
[IsRevisionMark property](#)  
[IsSame property](#)  
[IsSave property](#)  
[IsScalable property](#)  
[IsScalableBorder property](#)  
[IsScripting property](#)  
[IsScrollable property](#)  
[IsShowing property](#)  
[IsSingleClickEntry property](#)  
[IsSizable property](#)  
[IsSizingViaMouse property](#)  
[IsSmartCorrectEnabled property](#)  
[IsSmartEditEnabled property](#)  
[IsSmartShadeEnabled property](#)  
[IsSnapTo property](#)  
[IsSortFromEnd property](#)  
[IsSpellBarUp property](#)  
[IsSpellMode property](#)  
[IsSqueeze property](#)  
[IsStyle property](#)  
[IsStyleSheet property](#)  
[IsSymbolic property](#)  
[IsTableHeading property](#)  
[IsTemp property](#)  
[IsTemporary method](#)  
[IsTextLocked property](#)  
[IsTextMenuEnabled property](#)  
[IsTOC property](#)  
[IsTrueType property](#)  
[IsUndoOn property](#)  
[IsUpdateAutomatic property](#)  
[IsUpdateIndex property](#)  
[IsUpdateTOC property](#)  
[IsValid property](#)  
[IsVerticalScrollBarCleanScrn property](#)  
[IsViewAnchor property](#)  
[IsViewBookmarks property](#)  
[IsViewClickHereBlocks property](#)  
[IsViewColGuides property](#)  
[IsViewColumnBreakMarks property](#)  
[IsViewDDELinks property](#)

[IsViewHorzRuler property](#)  
[IsViewHorzScrollBar property](#)  
[IsViewMenuCleanScrn property](#)  
[IsViewNotes property](#)  
[IsViewOutlineFlowToScreen property](#)  
[IsViewOutlineIndent property](#)  
[IsViewPageLayoutMarks property](#)  
[IsViewParallelColumnBorder property](#)  
[IsViewPictures property](#)  
[IsViewReturnIconCleanScrn property](#)  
[IsViewReturns property](#)  
[IsViewRubyMarks property](#)  
[IsViewRulerMarks property](#)  
[IsViewSectionBreakMarks property](#)  
[IsViewSectionTabs property](#)  
[IsViewSmartIconsCleanScrn property](#)  
[IsViewStatusBarCleanScrn property](#)  
[IsViewStatusSpell property](#)  
[IsViewTableCellSelection property](#)  
[IsViewTableGridLines property](#)  
[IsViewTableHeadings property](#)  
[IsViewTabs property](#)  
[IsViewThumbBarCleanScrn property](#)  
[IsViewTitleBarCleanScrn property](#)  
[IsViewVertRuler property](#)  
[IsViewVertScrollBar property](#)  
[IsWMMCommandValid method](#)  
[IsWordProChart property](#)  
[IsWorkingDir property](#)  
[Italic method](#)  
[Italic property](#)  
[Item method](#)  
[Items property](#)

## J

[Join class](#)  
[Join property](#)  
[JoinCorners property](#)  
[JoinHeight property](#)  
[JoinType property](#)  
[JoinWidth property](#)  
[Justifiable property](#)

## K

[KeepWithNext property](#)  
[KeepWithPrev property](#)  
[KeyboardLanguage property](#)  
[KeyStroke event](#)  
[KeywordColor property](#)  
[Keywords property](#)  
[Kinsoku class](#)  
[Kinsoku property](#)

## L

[LabelCreate method](#)

[LabelMerge method](#)  
[LandscapeMode property](#)  
[Language class](#)  
[Language property](#)  
[Last property](#)  
[LastChild property](#)  
[LastDivision property](#)  
[LastDocPath property](#)  
[LastEditorName property](#)  
[LastGraphicPath property](#)  
[LastGraphicType property](#)  
[LastMacroPath property](#)  
[LastName property](#)  
[LastPage property](#)  
[LastUsedDateFormula property](#)  
[Layer property](#)  
[LayerName property](#)  
[Layout class](#)  
[Layout property](#)  
[LayoutCollection class](#)  
[LayoutName property](#)  
[LayoutOverride class](#)  
[LayoutOverride property](#)  
[Layouts property](#)  
[LeaderDotDashChar property](#)  
[LeaderDotDotChar property](#)  
[LeaderDotType property](#)  
[LeaderDotUnderscoreChar property](#)  
[LeadingText property](#)  
[LeastRecentVersion method](#)  
[Left property](#)  
[LeftBorder property](#)  
[LeftExternalMargin property](#)  
[LeftIntArea property](#)  
[LeftPage property](#)  
[LeftTopCellId property](#)  
[Length property](#)  
[Level property](#)  
[LineLocation property](#)  
[LineMix property](#)  
[LineNumberOptions property](#)  
[LineNumberOptions class](#)  
[LinePlacement property](#)  
[LinesSpacedEveryNthUnit property](#)  
[LineValid property](#)  
[Link method](#)  
[LinkAvailable property](#)  
[LinkContainers method](#)  
[LinkDisplayName property](#)  
[LinkDisplayNameFileLength property](#)  
[Linked property](#)  
[LinkedFileName property](#)  
[LinkFrame property](#)  
[LinkGraphic property](#)  
[ListCount property](#)  
[LoadFilesMaximized property](#)  
[Localize method](#)  
[Location property](#)



[Locked property](#)  
[LockForNotesUserName property](#)  
[LockResult property](#)  
[Locks property](#)  
[LowerCase property](#)  
[LowerCase method](#)  
[LowerCaseAscent property](#)  
[LwpMenuBar property](#)  
[LWPTimer class](#)

## M

[Macro class](#)  
[Macro property](#)  
[MacroEndRecord method](#)  
[MacroName property](#)  
[MacroPath property](#)  
[MacroPaths property](#)  
[MacroPlay method](#)  
[MacroRecord method](#)  
[MacroStatus property](#)  
[MailDocument method](#)  
[MailRouting class](#)  
[MailRouting property](#)  
[MailRoutingPtr property](#)  
[MaintainAspectRatio property](#)  
[MaintainEditor property](#)  
[MakeTableFromText method](#)  
[MakeUniqueLinkName method](#)  
[ManualFrame method](#)  
[ManualLinkFrames method](#)  
[ManualTable method](#)  
[MarginBottom property](#)  
[MarginColor property](#)  
[MarginLeft property](#)  
[MarginRight property](#)  
[MarginTop property](#)  
[Mark method](#)  
[MarkCharacter property](#)  
[Marker class](#)  
[MarkerClass property](#)  
[MarkerCollection class](#)  
[MarkerName property](#)  
[Markers property](#)  
[MarkPosition property](#)  
[MarkRevisionInsert method](#)  
[MarkType property](#)  
[Master property](#)  
[MasterName property](#)  
[MatchType property](#)  
[MaxBottomBorder property](#)  
[MaxBottomGutter property](#)  
[MaxContentHeight property](#)  
[MaxContentWidth property](#)  
[MaxHorzPaneDistance property](#)  
[MaxHyphLines property](#)  
[MaxIdenticalConsecSentOpens property](#)  
[MaxIdenticalSentOpensWithin10 property](#)

[Maximize method](#)  
[MaximizeOnStartUp property](#)  
[MaximumWordsinaSentence property](#)  
[MaxLeftBorder property](#)  
[MaxLeftGutter property](#)  
[MaxNumColsAllowed property](#)  
[MaxNumRowsAllowed property](#)  
[MaxRightBorder property](#)  
[MaxRightGutter property](#)  
[MaxSplitCols property](#)  
[MaxSplitRows property](#)  
[MaxTopBorder property](#)  
[MaxTopGutter property](#)  
[MaxVertPaneDistance property](#)  
[MenuItem class](#)  
[MenuItemCollection class](#)  
[Merge method](#)  
[MergeAddDataRecord method](#)  
[MergeFileType property](#)  
[MergeInfoPtr property](#)  
[MergeOptions class](#)  
[MergeOptions property](#)  
[MergeSetDataFile method](#)  
[MergeStart method](#)  
[MergeStepNumber property](#)  
[MergeToFile property](#)  
[Messages method](#)  
[MetafilePict property](#)  
[MinBottomMargin property](#)  
[MinHeight property](#)  
[Minimize method](#)  
[MinLeftMargin property](#)  
[MinRightMargin property](#)  
[MinTopMargin property](#)  
[MirrorPage method](#)  
[ModifiedDateString property](#)  
[ModifiedTimeString property](#)  
[ModifiedTimeValue property](#)  
[MorphSelectionToTable method](#)  
[MostRecentVersion method](#)  
[MouseButtonForManipulation property](#)  
[MouseButtonForSelection property](#)  
[MouseDown event](#)  
[MouseProperty property](#)  
[MouseUp event](#)  
[Move method](#)  
[Moved event](#)  
[MoveDivision method](#)  
[MoveDown method](#)  
[MoveItem method](#)  
[MoveParagraph method](#)  
[MoveToBack method](#)  
[MoveToEnd method](#)  
[MoveToFront method](#)  
[MoveToStart method](#)  
[MoveUp method](#)  
[MultiCellPasteOn property](#)  
[MultiCompareParaTag property](#)

[MultiCompareParaTagSet property](#)

## N

[Name property](#)

[NameBasedOnStyle property](#)

[NameForFilters property](#)

[Negative property](#)

[NewDivision method](#)

[NewDocMacroName property](#)

[NewDocument method](#)

[NewFile property](#)

[NewFrame method](#)

[NewItem method](#)

[NewItemByPosition method](#)

[NewVersion property](#)

[NewWindow method](#)

[Next method](#)

[NextClickHere property](#)

[NextCycleAlign method](#)

[NextCycleAttribute method](#)

[NextCycleBullet method](#)

[NextCycleFont method](#)

[NextCycleFontSize method](#)

[NextCycleIndent method](#)

[NextCycleNumber method](#)

[NextCycleStyle method](#)

[NextItem method](#)

[NextName property](#)

[NextNeighbor property](#)

[NextSection property](#)

[NextSpellString property](#)

[NextStyleName property](#)

[NextText property](#)

[NextToObject method](#)

[NextVersion method](#)

[NoHyphenate property](#)

[NonUserDocument property](#)

[Normal property](#)

[NormalParagraph property](#)

[NormalText method](#)

[NoteColor property](#)

[NoteContainer class](#)

[NoteLayout class](#)

[NoteLayoutCollection class](#)

[NoteLayouts property](#)

[NotesFlow property](#)

[Number property](#)

[NumberEveryNthLine property](#)

[NumberFound property](#)

[Numbering class](#)

[Numbering property](#)

[NumberingPosition property](#)

[NumberingStyleName property](#)

[NumberOfCharacters property](#)

[NumberOfDataFields property](#)

[NumberOfLines property](#)

[NumberOfMergeConditions property](#)

[NumberOfReplacements property](#)  
[NumberOfRevisions property](#)  
[NumberOfVersions method](#)  
[NumberSequenceName property](#)  
[NumberWhichLines property](#)  
[NumCharsInDoc property](#)  
[NumCharsInParagraph property](#)  
[NumCols property](#)  
[NumColsSpannedOneCell property](#)  
[NumContainers property](#)  
[NumDecimalPlaces property](#)  
[NumericFormat class](#)  
[NumericFormat property](#)  
[NumericFormatSubset class](#)  
[NumFields property](#)  
[NumLinesOfSpace property](#)  
[NumLinesOfSpaceAbove property](#)  
[NumLinesOfSpaceAboveLine property](#)  
[NumLinesOfSpaceBelow property](#)  
[NumOfRecentFiles property](#)  
[NumPagesInDoc property](#)  
[NumParagraphs property](#)  
[NumRows property](#)  
[NumRowsSpannedOneCell property](#)  
[NumRowsThatFit property](#)  
[NumRowsToFit property](#)  
[NumTabs property](#)  
[NumUndoLevels property](#)  
[NumWindowsViewingDoc property](#)  
[NumWordsInDoc property](#)

## O

[Object property](#)  
[ObjectType property](#)  
[Oblique property](#)  
[OCXDesignMode property](#)  
[Offset property](#)  
[Ole1Object property](#)  
[OLEAutomation property](#)  
[OleContainerDocName property](#)  
[OLEEnabled property](#)  
[OleMinHeight property](#)  
[OleObject property](#)  
[OleObject class](#)  
[OleObjectCollection class](#)  
[OleObjects property](#)  
[OleObjectSize property](#)  
[Open method](#)  
[OpenDataFile method](#)  
[OpenDocMacroName property](#)  
[OpenDocsVisible property](#)  
[OpenDocument method](#)  
[OpenDocumentFromNotes method](#)  
[Opened event](#)  
[OpenExisingFileInWelcomeBox property](#)  
[OpenFromStorage method](#)  
[OpenObject method](#)

[OpenReadOnly property](#)  
[OpionDeclare property](#)  
[Options property](#)  
[Options class](#)  
[OrigFileType property](#)  
[OrigHeight property](#)  
[OrigWidth property](#)  
[Outline property](#)  
[OutlineBorderLines property](#)  
[OutlineButtons property](#)  
[OutlineHeadingButtonsOnly property](#)  
[OutlineLevel property](#)  
[OutlineMoveTextDown method](#)  
[OutlineMoveTextUp method](#)  
[OutlineOnlyHeadingsWhenCollapsed property](#)  
[OutlineSeqCollection class](#)  
[OutlineSeqItemCollection class](#)  
[OutlineSeqItems property](#)  
[OutlineStyleSequence class](#)  
[OutlineStyleSequence method](#)  
[OutlineStyleSequences property](#)  
[OutputToFile property](#)  
[OutSeqItem class](#)  
[Override property](#)  
[Overstrike property](#)  
[OverstrikeCharacter property](#)

## P

[Page property](#)  
[PageContainer class](#)  
[PageDown method](#)  
[PageLayout class](#)  
[PageLayoutCollection class](#)  
[PageNo property](#)  
[PageNum property](#)  
[PageNumber property](#)  
[PageNumberAsText property](#)  
[PageNumberStyle property](#)  
[PageNumFirstPageShowing property](#)  
[PageNumLastPageShowing property](#)  
[PageOrder property](#)  
[Pages property](#)  
[PageStyleName property](#)  
[PageStyles property](#)  
[PageToUseLayoutOn property](#)  
[PageUp method](#)  
[PairKerning property](#)  
[PaneColor property](#)  
[PaperNames property](#)  
[ParagraphBorder class](#)  
[ParagraphBorder property](#)  
[ParagraphHasDropCap property](#)  
[ParagraphHasText property](#)  
[ParagraphStyle class](#)  
[ParagraphStyle property](#)  
[ParagraphStyleCollection class](#)  
[ParagraphStyleName property](#)

[ParagraphStyles property](#)  
[ParagraphSymbolChar property](#)  
[ParallelColsCollection class](#)  
[ParallelColsContainer class](#)  
[ParallelColumns class](#)  
[ParallelColumns property](#)  
[Parent property](#)  
[ParentMenuHWND property](#)  
[ParentName property](#)  
[Partial property](#)  
[Paste method](#)  
[PasteLink method](#)  
[PasteSpecial method](#)  
[Path property](#)  
[PathName property](#)  
[Pattern property](#)  
[PColConnectCells method](#)  
[PColConnectRows method](#)  
[PColDisconnectCells method](#)  
[PColSelectColumn method](#)  
[PColSelectRow method](#)  
[PColSelectTable method](#)  
[Percentage property](#)  
[PersonalData1 property](#)  
[PersonalData2 property](#)  
[PersonalData3 property](#)  
[PersonalData4 property](#)  
[PhoneNumber property](#)  
[PitchAndFamily property](#)  
[Placement property](#)  
[Plain property](#)  
[Play method](#)  
[PointSize property](#)  
[Position property](#)  
[PositionType property](#)  
[PositionXInContainer property](#)  
[PositionXOnPage property](#)  
[PositionYInContainer property](#)  
[PositionYOnPage property](#)  
[PowerField property](#)  
[PowerField class](#)  
[PowerFieldCollection class](#)  
[PowerFields property](#)  
[PreClose event](#)  
[Preferences property](#)  
[Preferences class](#)  
[Prefix property](#)  
[PrepareToDestroy method](#)  
[PrePrint event](#)  
[Present method](#)  
[Presentation property](#)  
[Presentation class](#)  
[PrettyPrinting property](#)  
[Previous method](#)  
[PreviousClickHere property](#)  
[PreviousItem method](#)  
[PreviousName property](#)  
[PreviousNeighbor property](#)

[PreviousVersion method](#)  
[Print method](#)  
[PrintDestination property](#)  
[PrintDocDescription property](#)  
[Printed event](#)  
[PrinterName property](#)  
[PrintGraphics property](#)  
[PrintInBackground property](#)  
[PrintManager property](#)  
[PrintManager class](#)  
[PrintOut method](#)  
[PrintPagesFrom property](#)  
[PrintPagesTo property](#)  
[PrintPageType property](#)  
[PrintRange property](#)  
[PrintSettings property](#)  
[PrintSettings class](#)  
[Private property](#)  
[ProcessAccelKey method](#)  
[ProgID property](#)  
[Promote method](#)  
[PromoteOutlineLevel method](#)  
[Prompt property](#)  
[PromptHidden property](#)  
[ProtectedMode property](#)  
[Purge method](#)

## Q

[Query method](#)  
[QueryDrop method](#)  
[QueueName property](#)  
[QuickAlignFrame method](#)  
[QuickAlignTable method](#)  
[Quit event](#)  
[Quit method](#)

## R

[Read method](#)  
[ReadCompressed property](#)  
[ReadOnly property](#)  
[Red property](#)  
[Redo method](#)  
[Refresh method](#)  
[RegisterWPDataSet method](#)  
[Relative property](#)  
[RelativeIndent property](#)  
[RelativeIndent class](#)  
[RelativePageNum property](#)  
[RelativeType property](#)  
[RelativeXDistance property](#)  
[RelativeYDistance property](#)  
[Release method](#)  
[ReleaseNumber property](#)  
[Remove method](#)  
[RemoveBookmark method](#)  
[RemoveChildFromLayout method](#)

[RemoveDataFile method](#)  
[RemoveDdeLink method](#)  
[RemoveDepOnDocFile method](#)  
[RemoveDivision method](#)  
[RemoveEditor method](#)  
[RemoveIndexEntry method](#)  
[RemoveList method](#)  
[RemoveNamedProperty method](#)  
[RemovePersistentAccelerators method](#)  
[RemovePowerField method](#)  
[RemoveProperty method](#)  
[RemoveTOCEntry method](#)  
[Render method](#)  
[RenderClipBitmap method](#)  
[RenderClipDIB method](#)  
[RenderClipMetafile method](#)  
[RenderClipPalette method](#)  
[RenderedPageNumber property](#)  
[Repaint method](#)  
[Replace method](#)  
[ReplaceAll method](#)  
[ReplaceAttributes property](#)  
[ReplaceCmd method](#)  
[ReplaceContents method](#)  
[ReplaceExactCase property](#)  
[ReplaceFont property](#)  
[ReplaceLanguage property](#)  
[Replacements property](#)  
[ReplaceString property](#)  
[ReplaceStyleName property](#)  
[ReplaceWithProperties property](#)  
[RequestAndProcessData method](#)  
[RequestRemarkOnClose property](#)  
[RequireStartupScripts property](#)  
[Reset method](#)  
[ResetFindAndReplace method](#)  
[ResetNumberOpts method](#)  
[ResetOnEachPage property](#)  
[ResetPrinting method](#)  
[ResetWhen property](#)  
[Resize method](#)  
[Rest property](#)  
[RestartStyleName property](#)  
[Restore method](#)  
[RestorePreviousView method](#)  
[RestoreWindowFromCleanScreen method](#)  
[Result property](#)  
[ResumePausedMacro property](#)  
[RetainNameOfImportedFile property](#)  
[RetrieveInternetFile method](#)  
[RetrieveInternetFileAndOpen method](#)  
[RevertToSaved method](#)  
[RevertToStyle method](#)  
[ReviewVersions method](#)  
[ReviewVersions class](#)  
[ReviewVersions property](#)  
[ReviseAcceptAll method](#)  
[ReviseCancelAll method](#)



[Revision class](#)  
[RevisionAccept method](#)  
[RevisionAcceptLayoutChange method](#)  
[RevisionCancel method](#)  
[RevisionCancelLayoutChange method](#)  
[RevisionDisplay class](#)  
[RevisionDisplay property](#)  
[RevisionMark property](#)  
[RevisionMarkMode property](#)  
[RevisionType property](#)  
[RevMarkCharacter property](#)  
[RevMarkPosition property](#)  
[RevMarkType property](#)  
[Right property](#)  
[RightAlign property](#)  
[RightBorder property](#)  
[RightExternalMargin property](#)  
[RightIntArea property](#)  
[RightMouseMenus property](#)  
[RightMousePropId property](#)  
[RightMousePropText property](#)  
[RightPage property](#)  
[RotationAngle property](#)  
[RowContainer class](#)  
[RowGroupLayout class](#)  
[RowLayout class](#)  
[RowLayoutCollection class](#)  
[RowLayouts property](#)  
[Rows property](#)  
[RubyContainer class](#)  
[RubyLayout class](#)  
[RubyLayoutCollection class](#)  
[RubyLayouts property](#)  
[RubyMarker class](#)  
[Ruler class](#)  
[Run method](#)  
[RunMacroOnDocEvents property](#)  
[RunMacroOnLoad property](#)  
[RunOnCloseDoc property](#)  
[RunOnNewDoc property](#)  
[RunOnOpenDoc property](#)  
[RunScript method](#)

## S

[Save method](#)  
[Save event](#)  
[SaveAs event](#)  
[SaveAs method](#)  
[SaveAsToNotes method](#)  
[Saved event](#)  
[Saved property](#)  
[SavedAs event](#)  
[SaveData method](#)  
[SaveDivision method](#)  
[SaveEnvelopeMaster method](#)  
[SaveGlossary method](#)  
[SaveMacro method](#)

[SaveMergeDataFile method](#)  
[SaveSnapShot property](#)  
[SaveSnapshot method](#)  
[SaveThumbnailBitmap method](#)  
[SaveToStorage method](#)  
[SaveToStorageComplete method](#)  
[SaveUserDefaults method](#)  
[SaveVersion method](#)  
[ScaleHeight property](#)  
[ScaleMode property](#)  
[ScalePercentage property](#)  
[ScaleWidth property](#)  
[ScreenPositionX property](#)  
[ScreenPositionY property](#)  
[Script class](#)  
[Script property](#)  
[ScriptDataSet class](#)  
[ScriptName property](#)  
[SearchAttributes property](#)  
[SearchLanguage property](#)  
[Section class](#)  
[Section property](#)  
[SectionCollection class](#)  
[SectionName property](#)  
[Sections property](#)  
[SectionTabs property](#)  
[SectionTabs class](#)  
[Select method](#)  
[SelectAStyle method](#)  
[SelectCell method](#)  
[SelectColumn method](#)  
[SelectCustomIcon method](#)  
[SelectDoc method](#)  
[SelectedPages property](#)  
[SelectEntireCellRange method](#)  
[SelectEntireColumn method](#)  
[SelectEntirePCol method](#)  
[SelectEntirePColCellRange method](#)  
[SelectEntirePColColumn method](#)  
[SelectEntirePColRow method](#)  
[SelectEntireRow method](#)  
[SelectEntireTable method](#)  
[SelectionBorderColor1 property](#)  
[SelectionBorderColor2 property](#)  
[SelectionBorderColor3 property](#)  
[SelectionHidden property](#)  
[SelectionType property](#)  
[SelectMarker method](#)  
[SelectParagraph method](#)  
[SelectPColCell method](#)  
[SelectRow method](#)  
[SelectRuler method](#)  
[SelectSection method](#)  
[SelectSentence method](#)  
[SelectStandardIcon method](#)  
[SelectTab property](#)  
[SelectTable method](#)  
[SelectTableItem method](#)

[SelectType property](#)  
[SelectWord method](#)  
[SendFrameToBack method](#)  
[SendFrameToBackOne method](#)  
[SendMailAndAttach method](#)  
[SendMailSelectedText method](#)  
[ServerFormat property](#)  
[SetAllMargins method](#)  
[SetArrayProp method](#)  
[SetButtonText method](#)  
[SetContextOfBar property](#)  
[SetCustomNumber method](#)  
[SetData method](#)  
[SetDocumentEpoch method](#)  
[SetFieldFormula method](#)  
[SetFocus method](#)  
[SetFormula method](#)  
[SetIndexInfo method](#)  
[SetJapanIndexInfo method](#)  
[SetLastUsedFilter method](#)  
[SetLineOneSide method](#)  
[SetLinesAllSides method](#)  
[SetLinkSource method](#)  
[SetMinimumOrigin method](#)  
[SetNamedProperty method](#)  
[SetNoFields method](#)  
[SetNumberingLevelInfo method](#)  
[SetOverrideGraphic method](#)  
[SetOverrideText method](#)  
[SetPageBottomMargin method](#)  
[SetPageTopMargin method](#)  
[SetPattern method](#)  
[SetPopupAlignment method](#)  
[SetPopupIndex method](#)  
[SetPopupWidth method](#)  
[SetPopupWidthType method](#)  
[SetPowerFieldValue method](#)  
[SetRGB method](#)  
[SetStorage method](#)  
[SetStyle method](#)  
[SetTableFill method](#)  
[SetTabsDialog property](#)  
[SetTabsDialog class](#)  
[SetTOCLevelContent method](#)  
[SetTOCLevelPageInfo method](#)  
[SetTOCProperties method](#)  
[SetUpEnvelopeMerge method](#)  
[Shade method](#)  
[Shadow property](#)  
[Shadow class](#)  
[Shape property](#)  
[Show method](#)  
[ShowAnyGreeting method](#)  
[ShowBubbleHelp property](#)  
[ShowCaretAndSelection method](#)  
[ShowContainers method](#)  
[ShowCursor method](#)  
[ShowDivisionTabs property](#)

[ShowFileNew property](#)  
[ShowGraphicPreview property](#)  
[ShowHiddenText property](#)  
[ShowIconBars method](#)  
[ShowInContext property](#)  
[ShowMailDisabled property](#)  
[ShowNoWelcomeBox property](#)  
[ShowScrollBars method](#)  
[ShowStatistics property](#)  
[ShowStatusBar method](#)  
[ShowTabs property](#)  
[SilverBullet class](#)  
[SilverBullet property](#)  
[SilverBulletCollection class](#)  
[SilverBullets property](#)  
[SimulateButtonClick method](#)  
[SingleCellSelected property](#)  
[Size property](#)  
[SizeStyleName property](#)  
[SizingUnitName property](#)  
[SizingUnits property](#)  
[Skip method](#)  
[Skipped property](#)  
[SkipWordMode property](#)  
[SmallCaps property](#)  
[SmallCaps method](#)  
[SmallFileFormat property](#)  
[SmartCorrect class](#)  
[SmartCorrect property](#)  
[SmartCorrectCollection class](#)  
[SmartCorrects property](#)  
[SmartFill class](#)  
[SmartFill property](#)  
[SmartFillCollection class](#)  
[SmartLevel property](#)  
[SmartSumColumn method](#)  
[SmartSumRow method](#)  
[SnapshotOffset property](#)  
[SnapshotPath property](#)  
[SnapshotSaveOptions property](#)  
[SnapshotSize property](#)  
[SortKey class](#)  
[SortLevel1 property](#)  
[SortLevel2 property](#)  
[SortLevel3 property](#)  
[SortNumbers property](#)  
[SortOptions property](#)  
[SortOptions class](#)  
[SortOrder property](#)  
[SortParagraphs method](#)  
[SortType property](#)  
[SortWord property](#)  
[SortWordOption property](#)  
[SpaceAbove property](#)  
[SpaceBelow property](#)  
[SpacesBetweenSentences property](#)  
[Spacing property](#)  
[Spacing class](#)

[SpacingStyleName property](#)  
[SpacingUnitName property](#)  
[SpacingUnits property](#)  
[Span property](#)  
[SpecialView method](#)  
[SpellAddToUserDict method](#)  
[SpellCheckInitialCaps property](#)  
[SpellCheckRepeatedWords property](#)  
[SpellCheckUserDictAlternatives property](#)  
[SpellCheckWordsWithNums property](#)  
[SpellClearSkippedWords method](#)  
[SpellColor property](#)  
[SpellFocusedColor property](#)  
[SpellMarkSkippedWords method](#)  
[SpellSkipAll method](#)  
[SpellString property](#)  
[SpellWord method](#)  
[Split method](#)  
[SplitDivision method](#)  
[SplitParagraph method](#)  
[SplitPercentage property](#)  
[SplitWindow method](#)  
[SRRReplace method](#)  
[Start property](#)  
[Start method](#)  
[StartColumns property](#)  
[StartEditMergeData method](#)  
[StartEnvelopeDiv method](#)  
[StartFieldInsert method](#)  
[StartingColOfSelection property](#)  
[StartingColStringOfSelection property](#)  
[StartingNumber property](#)  
[StartingRowOfSelection property](#)  
[StartRow property](#)  
[StateID property](#)  
[StatusBar class](#)  
[StatusBar property](#)  
[StatusBarButton class](#)  
[StatusBarButtonClicked event](#)  
[StatusBarButtonCollection class](#)  
[StatusBarButtonFillPopupList event](#)  
[StatusBarButtonItemSelected event](#)  
[StatusBarButtonOverrideGraphic event](#)  
[StatusBarButtonOverrideText event](#)  
[StatusBarButtonOverrideTextAndGraphic event](#)  
[StatusBarButtons property](#)  
[StatusBarVisible property](#)  
[StatusSpellReplaceAll property](#)  
[StoreInternetFile method](#)  
[StrikeThrough property](#)  
[StrikeThru method](#)  
[StringCollection class](#)  
[Style property](#)  
[StyleExceptions property](#)  
[StyleName property](#)  
[StylePath property](#)  
[StylePaths property](#)  
[StyleSheetFullPath property](#)

[StyleSheetName property](#)  
[StyleSheetPath property](#)  
[SubPageContainer class](#)  
[SubScript method](#)  
[Subscript property](#)  
[Suffix property](#)  
[Suggestions property](#)  
[Sum method](#)  
[SuperPageContainer class](#)  
[SuperScript method](#)  
[Superscript property](#)  
[SuperTable class](#)  
[SuperTableCollection class](#)  
[SuperTableContainer property](#)  
[SuperTableContainer class](#)  
[SuperTableGroupLayout class](#)  
[SuperTableLayout class](#)  
[SuperTableLayoutCollection class](#)  
[SuperTableLayouts property](#)  
[SuperTables property](#)  
[SuppressHeaders property](#)

## T

[TabExits property](#)  
[Table class](#)  
[Table property](#)  
[TableCollection class](#)  
[TableContainer property](#)  
[TableContainer class](#)  
[TableExports property](#)  
[TableFill class](#)  
[TableFill property](#)  
[TableHeading class](#)  
[TableHeadingCollection class](#)  
[TableHeadingLayout class](#)  
[TableHeadingLayoutCollection class](#)  
[TableHeadingLayouts property](#)  
[TableHeadings property](#)  
[TableImports property](#)  
[TableLayout class](#)  
[TableLayoutCollection class](#)  
[TableLayouts property](#)  
[TableLine property](#)  
[TableLine class](#)  
[TableMarker class](#)  
[TableMarkerCollection class](#)  
[TableMarkers property](#)  
[TableOnlyCollection class](#)  
[TableOnlyCont class](#)  
[TableOnlyContainer property](#)  
[Tables property](#)  
[TableStyleName property](#)  
[TableStyles property](#)  
[TabOrder property](#)  
[TabRack property](#)  
[TabRack class](#)  
[TabRelativeTo property](#)

[TabSpacing property](#)  
[TabSymbolChar property](#)  
[TabType property](#)  
[TabWidth property](#)  
[TeamMail method](#)  
[TempFindAndReplace property](#)  
[TempFoundry property](#)  
[Text class](#)  
[Text property](#)  
[TextandTableExports property](#)  
[TextandTableImports property](#)  
[TextAttributes property](#)  
[TextCollection class](#)  
[TextDocument class](#)  
[TextMarker class](#)  
[TextMarkerCollection class](#)  
[TextMarkers property](#)  
[TextNumber method](#)  
[TextOrient property](#)  
[TextOrientation property](#)  
[Texts property](#)  
[TextStyleCollection class](#)  
[TextStyleName property](#)  
[TextStyles property](#)  
[TextTightness property](#)  
[TextViewAttributes property](#)  
[TheoreticalScaledSize method](#)  
[ThesMaxNumSynonymsReturned property](#)  
[Tile method](#)  
[Tile property](#)  
[TileWindow property](#)  
[TileWindowHorz method](#)  
[TileWindowVert method](#)  
[Time property](#)  
[TimedSave method](#)  
[TimedSaveFileExtension property](#)  
[TimerTick event](#)  
[Title property](#)  
[TitleBarDocNumber property](#)  
[TitleBarVisible property](#)  
[TOCRange property](#)  
[TOCSource property](#)  
[TOCSuperTableLayout class](#)  
[ToggleCleanScreen method](#)  
[ToggleIconBar method](#)  
[Top property](#)  
[TopBorder property](#)  
[TopExternalMargin property](#)  
[TopIntArea property](#)  
[TopLeftCellRowId property](#)  
[TotalEditingTime property](#)  
[TrailingText property](#)  
[Type method](#)  
[Type property](#)  
[TypeAbove property](#)  
[TypeAboveLine property](#)  
[TypeBelow property](#)  
[TypeRight property](#)

[Typos property](#)

## U

[Underline property](#)

[Underline method](#)

[Undo method](#)

[UndoEnable property](#)

[UndoLevels property](#)

[UndoRedo method](#)

[Units property](#)

[Unlink method](#)

[UnLinkContainers method](#)

[UnregisterWPDataSet method](#)

[Update method](#)

[UpdateFields property](#)

[UpdateFooterContent method](#)

[UpdateFootersText method](#)

[UpdateHeaderContent method](#)

[UpdateHeadersText method](#)

[UpdateIndexSection method](#)

[UpdateLink method](#)

[UpdateOle method](#)

[UpdateOnLoad property](#)

[UpdateOnLoadImmediate property](#)

[UpdatePageSizeChange method](#)

[UpdatePowerFields method](#)

[UpdatePowerFieldsOnNew method](#)

[UpdatePrinterBins method](#)

[UpdatePrinterChanges method](#)

[UpdateSelectedFields method](#)

[UpdateTabs method](#)

[UpdateTOC method](#)

[UpdateUI method](#)

[UpperCase method](#)

[UpperCase property](#)

[UseConsistentSpaceBetweenSentences property](#)

[UseContents property](#)

[UseDefaultPrinter property](#)

[UsedFirstPageHeight property](#)

[UseEncrypt property](#)

[UseExcludeRect property](#)

[UseFindStyle property](#)

[UseFooter property](#)

[UseGreeting property](#)

[UseHeader property](#)

[UseNextStyle property](#)

[UsePrinterSettings property](#)

[UserClassNameApp property](#)

[UserClassNameFull property](#)

[UserClassNameShort property](#)

[UserDefinedFilter property](#)

[UserDictFiles property](#)

[UserDictionaryFiles property](#)

[UserDictionaryPath property](#)

[UserDictionaryPaths property](#)

[UserDictStates property](#)

[UseRelative property](#)



[UseReplaceStyle property](#)  
[UserInitials property](#)  
[UserInterfacePrefs property](#)  
[UserInterfacePrefs class](#)  
[UserLinkName property](#)  
[UserName property](#)  
[UseSeparatorLine property](#)  
[UsesHelp property](#)  
[UsesPalette property](#)  
[UseSuperscriptReferenceNum property](#)  
[UseTwoSpacesBetweenSentences property](#)  
[UseWhen class](#)  
[UseWhen property](#)

## V

[Valid property](#)  
[ValidateValue method](#)  
[Version class](#)  
[VersionCollection class](#)  
[VersionID property](#)  
[VersionManager class](#)  
[VersionManager property](#)  
[VersionName property](#)  
[VersionRemarks property](#)  
[Versions property](#)  
[VertAlign property](#)  
[VerticalSplitWindow property](#)  
[VertRuler property](#)  
[VertScrollBarVisible property](#)  
[ViewLevel property](#)  
[ViewType property](#)  
[Visible property](#)

## W

[WasDeletedInRevMarkMode property](#)  
[WasInsertedInRevMarkMode property](#)  
[WasPasted property](#)  
[WaterMarkName property](#)  
[Weight property](#)  
[Where property](#)  
[WidowOrphan property](#)  
[Width property](#)  
[WidthAbove property](#)  
[WidthBelow property](#)  
[WidthInLongtwips property](#)  
[WidthOfColumn property](#)  
[Window class](#)  
[WindowId property](#)  
[WindowsName property](#)  
[WinViewPrefs property](#)  
[WinViewPrefs class](#)  
[WithComments property](#)  
[WMCommand event](#)  
[WMCommand method](#)  
[WordCount method](#)  
[WordDoubleUnderline property](#)

[WordUnderline method](#)  
[WordUnderline property](#)  
[WorkingType property](#)  
[WPAApplication class](#)  
[WPDataSet class](#)  
[WPDataSetCollection class](#)  
[WPDataSets property](#)  
[WrapType property](#)  
[Write method](#)  
[WriteProfileString method](#)

## X

[XOffset property](#)  
[XPosition property](#)

## Y

[YOffset property](#)  
[YPosition property](#)

## Z

[Zero property](#)  
[ZipCode property](#)

## Word Pro Classes

A  
B  
C  
D  
E  
F  
G  
H  
I  
J  
K  
L  
M  
N  
O  
P  
Q  
R  
S  
T  
U  
V  
W  
X  
Y  
Z

### A

[Accelerators class](#)  
[Alignment class](#)  
[Amikake class](#)  
[Application class](#)  
[ApplicationWindow class](#)  
[AppViewPrefs class](#)  
[Attributes class](#)  
[AutoRunMacro class](#)

### B

[Background class](#)  
[Bag class](#)  
[BagCollection class](#)  
[BaseCollection class](#)  
[BaseContainer class](#)  
[BaseObject class](#)  
[BaseTable class](#)  
[Bookmark class](#)  
[BookmarkCollection class](#)  
[BookmarkManager class](#)  
[Border class](#)  
[BorderLines class](#)  
[Breaks class](#)  
[Bullet class](#)

### C

[CellCollection class](#)

CellContainer class  
CellEngine class  
CellGroupLayout class  
CellLayout class  
CellLayoutCollection class  
CharacterBorder class  
CharacterSet class  
CharacterStyle class  
CharacterStyleCollection class  
ClickHere class  
ClickHereCollection class  
Color class  
Column class  
ColumnGroupLayout class  
ConnectedLayout class  
ConnectedLayoutCollection class  
Content class  
ContentCollection class  
ContextMenuOptions class

## D

DdeLink class  
DdeLinkCollection class  
DdeLinkManager class  
Division class  
DivisionCollection class  
DivisionInfo class  
DivisionOptions class  
DocControl class  
DocInfo class  
DocInfoField class  
DocInfoFieldCollection class  
DocInfoFieldManager class  
Document class  
Documents class  
DocWindow class  
DocWindowCollection class  
DropCapContainer class  
DropCapLayout class  
DropCapLayoutCollection class

## E

Editor class  
EditorCollection class  
EditorManager class  
EndnoteDivisionGroupNum class  
EndnoteDivisionNum class  
EndnoteDocNum class  
EndnoteLayout class  
EndnoteLayoutCollection class

## F

Filter class  
FilterHelper class  
FindAndReplace class

[Font class](#)  
[FontMetrics class](#)  
[Footer class](#)  
[FooterLayout class](#)  
[FooterLayoutCollection class](#)  
[Footnote class](#)  
[FootnoteCollection class](#)  
[FootnoteContSep class](#)  
[FootnoteLayout class](#)  
[FootnoteLayoutCollection class](#)  
[FootnoteNumbering class](#)  
[FootnoteNumOpt class](#)  
[FootnoteOptions class](#)  
[FootnoteSeparator class](#)  
[FootnoteSepOpt class](#)  
[FootnoteTable class](#)  
[FormatCheckPref class](#)  
[FormatPreferences class](#)  
[Formula class](#)  
[Foundry class](#)  
[FrameContainer class](#)  
[FrameGroupLayout class](#)  
[FrameLayout class](#)  
[FrameLayoutCollection class](#)

## G

[Glossary class](#)  
[GlossaryCollection class](#)  
[Grammar class](#)  
[Graphic class](#)  
[GraphicCollection class](#)  
[GraphicOleObject class](#)  
[GraphicOleObjectCollection class](#)  
[GroupLayout class](#)  
[GroupLayoutCollection class](#)  
[Gutter class](#)

## H

[Header class](#)  
[HeaderLayout class](#)  
[HeaderLayoutCollection class](#)  
[HyphenationOptions class](#)

## I

[IconBar class](#)  
[IconBarCollection class](#)  
[IconBarManager class](#)  
[Indent class](#)  
[Index class](#)  
[IndexSection class](#)

## J

[Join class](#)

## K

[Kinsoku class](#)

## L

[Language class](#)

[Layout class](#)

[LayoutCollection class](#)

[LayoutOverride class](#)

[LineNumberOptions class](#)

[LWPTimer class](#)

## M

[Macro class](#)

[MailRouting class](#)

[Marker class](#)

[MarkerCollection class](#)

[MenuItem class](#)

[MenuItemCollection class](#)

[MergeOptions class](#)

## N

[NoteContainer class](#)

[NoteLayout class](#)

[NoteLayoutCollection class](#)

[Numbering class](#)

[NumericFormat class](#)

[NumericFormatSubset class](#)

## O

[OleObject class](#)

[OleObjectCollection class](#)

[Options class](#)

[OutlineSeqCollection class](#)

[OutlineSeqItemCollection class](#)

[OutlineStyleSequence class](#)

[OutSeqItem class](#)

## P

[PageContainer class](#)

[PageLayout class](#)

[PageLayoutCollection class](#)

[ParagraphBorder class](#)

[ParagraphStyle class](#)

[ParagraphStyleCollection class](#)

[ParallelColsCollection class](#)

[ParallelColsContainer class](#)

[ParallelColumns class](#)

[PowerField class](#)

[PowerFieldCollection class](#)

[Preferences class](#)

[Presentation class](#)

PrintManager class  
PrintSettings class

## Q

## R

RelativeIndent class  
ReviewVersions class  
Revision class  
RevisionDisplay class  
RowContainer class  
RowGroupLayout class  
RowLayout class  
RowLayoutCollection class  
RubyContainer class  
RubyLayout class  
RubyLayoutCollection class  
RubyMarker class  
Ruler class

## S

Script class  
ScriptDataSet class  
Section class  
SectionCollection class  
SectionTabs class  
SetTabsDialog class  
Shadow class  
SilverBullet class  
SilverBulletCollection class  
SmartCorrect class  
SmartCorrectCollection class  
SmartFill class  
SmartFillCollection class  
SortKey class  
SortOptions class  
Spacing class  
StatusBar class  
StatusBarButton class  
StatusBarButtonCollection class  
StringCollection class  
SubPageContainer class  
SuperPageContainer class  
SuperTable class  
SuperTableCollection class  
SuperTableContainer class  
SuperTableGroupLayout class  
SuperTableLayout class  
SuperTableLayoutCollection class

## T

Table class  
TableCollection class  
TableContainer class

[TableFill class](#)  
[TableHeading class](#)  
[TableHeadingCollection class](#)  
[TableHeadingLayout class](#)  
[TableHeadingLayoutCollection class](#)  
[TableLayout class](#)  
[TableLayoutCollection class](#)  
[TableLine class](#)  
[TableMarker class](#)  
[TableMarkerCollection class](#)  
[TableOnlyCollection class](#)  
[TableOnlyCont class](#)  
[TabRack class](#)  
[Text class](#)  
[TextCollection class](#)  
[TextDocument class](#)  
[TextMarker class](#)  
[TextMarkerCollection class](#)  
[TextStyleCollection class](#)  
[TOCSuperTableLayout class](#)

## **U**

[UserInterfacePrefs class](#)  
[UseWhen class](#)

## **V**

[Version class](#)  
[VersionCollection class](#)  
[VersionManager class](#)

## **W**

[Window class](#)  
[WinViewPrefs class](#)  
[WPAApplication class](#)  
[WPDataSet class](#)  
[WPDataSetCollection class](#)

## **X**

## **Y**

## **Z**



## Word Pro Events

Created event  
DocumentClose event  
DocumentClosed event  
DocumentCreate event  
DocumentCreated event  
DocumentExport event  
DocumentExported event  
DocumentImport event  
DocumentImported event  
DocumentInsert event  
DocumentInserted event  
DocumentOpen event  
DocumentOpened event  
DocumentPrint event  
DocumentPrinted event  
DocumentSave event  
DocumentSaveAs event  
DocumentSaved event  
DocumentSavedAs event  
EnterClickHere event  
EnterLayout event  
ExitClickHere event  
ImportInsert event  
ImportInserted event  
KeyStroke event  
MouseDown event  
MouseUp event  
Moved event  
Opened event  
PreClose event  
PrePrint event  
Printed event  
Quit event  
Save event  
SaveAs event  
Saved event  
SavedAs event  
StatusBarButtonClicked event  
StatusBarButtonFillPopuList event  
StatusBarButtonItemSelected event  
StatusBarButtonOverrideGraphic event  
StatusBarButtonOverrideText event  
StatusBarButtonOverrideTextAndGraphic event  
TimerTick event  
WMCommand event



## Word Pro Methods

A  
B  
C  
D  
E  
F  
G  
H  
I  
J  
K  
L  
M  
N  
O  
P  
Q  
R  
S  
T  
U  
V  
W  
X  
Y  
Z

### A

[Abandon method](#)  
[Activate method](#)  
[ActivateAs method](#)  
[Add method](#)  
[AddAccelerators method](#)  
[AddACondition method](#)  
[AddBookmark method](#)  
[AddChildToLayout method](#)  
[AddDdeLink method](#)  
[AddDivision method](#)  
[AddDivisionToPrint method](#)  
[AddEditorManager method](#)  
[AddField method](#)  
[AddIcon method](#)  
[AddIndexEntry method](#)  
[AddLayoutOverride method](#)  
[AddNewSectionTabs method](#)  
[AddOutlineSequenceItem method](#)  
[AddPopupGraphicItem method](#)  
[AddPopupPointSizeItem method](#)  
[AddPopupTextItem method](#)  
[AddSectionTabs method](#)  
[AddSmartCorrect method](#)  
[AddStringToList method](#)  
[AddTOCEntry method](#)  
[AddVerbMenu method](#)  
[AdjustShade method](#)  
[Adopt method](#)  
[AdviseOnRename method](#)

[AdviseOnSave method](#)  
[Anchor method](#)  
[AnswerMsgBox method](#)  
[AnyEdits method](#)  
[AppendMacro method](#)

## **B**

[Backspace method](#)  
[Backward method](#)  
[BeginChange method](#)  
[BeginCustomLines method](#)  
[BinNameFromNumber method](#)  
[Bisect method](#)  
[Bold method](#)  
[BreakLink method](#)  
[BringFrameToFront method](#)  
[BringFrameToFrontOne method](#)

## **C**

[CalcSmartLevels method](#)  
[CalculateSmartLevels method](#)  
[CanHaveFootnotes method](#)  
[Cascade method](#)  
[CascadeWindow method](#)  
[CellLayout method](#)  
[CellRevert method](#)  
[ChangeAllEditsToEditor method](#)  
[ChangeSmartMaster method](#)  
[CheckFieldEntries method](#)  
[ChgLineStyle method](#)  
[Clear method](#)  
[ClearAll method](#)  
[ClearDivisionList method](#)  
[ClearInternalSpellInfo method](#)  
[ClearParaRevisionTags method](#)  
[ClearPopupData method](#)  
[ClearSplits method](#)  
[ClearTempFoundry method](#)  
[ClearUpdate method](#)  
[Close method](#)  
[CloseAll method](#)  
[CloseDocWindow method](#)  
[CloseMergeDataFile method](#)  
[CloseObject method](#)  
[CombineDivisions method](#)  
[CombineSections method](#)  
[CompareFiles method](#)  
[Configure method](#)  
[Connect method](#)  
[ConnectCells method](#)  
[ConnectContainer method](#)  
[ConnectRows method](#)  
[ConnectSectionTabs method](#)  
[ConsistencyCheck method](#)  
[Contract method](#)  
[ContractOutlineLevel method](#)

[ConvertTo method](#)  
[ConvertToClass method](#)  
[Copy method](#)  
[CopyItem method](#)  
[CopyMeaning method](#)  
[CopySelection method](#)  
[Create method](#)  
[CreateDataFile method](#)  
[CreateDivision method](#)  
[CreateDocument method](#)  
[CreateDropCap method](#)  
[CreateEmptyList method](#)  
[CreateExternalDivision method](#)  
[CreateFrame method](#)  
[CreateFromBitmap method](#)  
[CreateFromClipBrd method](#)  
[CreateFromDataObject method](#)  
[CreateFromMetafile method](#)  
[CreateGlossary method](#)  
[CreateGlossaryEntry method](#)  
[CreateGraphic method](#)  
[CreateLayer method](#)  
[CreateNew method](#)  
[CreateNewButton method](#)  
[CreateOleEmbeddedFile method](#)  
[CreateOleLinkedFile method](#)  
[CreateOleNew method](#)  
[CreateParallelColumns method](#)  
[CreateRemark method](#)  
[CreateTable method](#)  
[CreateVersion method](#)  
[CutSelection method](#)

## D

[DarkMode method](#)  
[DataObjectGetData method](#)  
[DataObjectGetDataHere method](#)  
[DbfUnderline method](#)  
[DeleteBag method](#)  
[DeleteButton method](#)  
[DeleteChars method](#)  
[DeleteContainer method](#)  
[DeleteContent method](#)  
[DeleteContents method](#)  
[DeleteDivision method](#)  
[DeleteField method](#)  
[DeletItem method](#)  
[DeletItemByPosition method](#)  
[DeleteKey method](#)  
[DeletLayout method](#)  
[DeleteMacroAccelerator method](#)  
[DeleteMarker method](#)  
[DeleteParallelColumns method](#)  
[DeleteSection method](#)  
[DeleteSmartCorrect method](#)  
[DeleteTab method](#)  
[DeleteTable method](#)

[DeleteVersion method](#)  
[Demote method](#)  
[DemoteOutlineLevel method](#)  
[Deselect method](#)  
[DeselectRuler method](#)  
[Destroy method](#)  
[DestroyDocWindow method](#)  
[Disconnect method](#)  
[DisconnectCells method](#)  
[DivideText method](#)  
[DoesMarkerNameMatch method](#)  
[DoneWithRightContextMenu method](#)

## E

[EditClickHereLink method](#)  
[EditLinkInfo method](#)  
[Embed method](#)  
[EmbedFormula method](#)  
[EndChange method](#)  
[EndCustomLines method](#)  
[Ending method](#)  
[EndPrinting method](#)  
[EnumerateChartLinks method](#)  
[EnumerateTerm method](#)  
[EnvelopeBarCode method](#)  
[EnvelopePrint method](#)  
[ExchangeItem method](#)  
[Expand method](#)  
[ExpandOutline method](#)  
[ExpandOutlineLevel method](#)  
[ExportAllAsNotesFX method](#)  
[ExportAsNotesFX method](#)  
[ExtractText method](#)

## F

[FastFormat method](#)  
[Find method](#)  
[FindCellLayout method](#)  
[FindClass method](#)  
[FindParent method](#)  
[FindTerm method](#)  
[ForceDocToLoad method](#)  
[Forward method](#)  
[FrameRevert method](#)  
[FXGetNotesString method](#)  
[FXGetNotesWriteHandle method](#)  
[FXSetNotesString method](#)

## G

[Get method](#)  
[GetActiveList method](#)  
[GetArrayProp method](#)  
[GetAspectMetafilePict method](#)  
[GetContents method](#)  
[GetCopyFormatCategories method](#)

[GetCount method](#)  
[GetCurrentMarkerName method](#)  
[GetData method](#)  
[GetDefaultPageSize method](#)  
[GetDisplayableFilterName method](#)  
[GetEnumerator method](#)  
[GetEnvelopeDefaults method](#)  
[GetFileDescription method](#)  
[GetFormula method](#)  
[GetInternetFile method](#)  
[GetLastUsedFilter method](#)  
[GetLineMix method](#)  
[GetLineStyle method](#)  
[GetLinkDisplayNameFileLength method](#)  
[GetLinkName method](#)  
[GetLinkSourceName method](#)  
[GetListName method](#)  
[GetMarkedText method](#)  
[GetMarkerName method](#)  
[GetMisspelledWord method](#)  
[GetNamedProperty method](#)  
[GetNameFromPage method](#)  
[GetObjectList method](#)  
[GetOne method](#)  
[GetPageRange method](#)  
[GetParagraphNumber method](#)  
[GetParaNumber method](#)  
[GetPasteFormatCategories method](#)  
[GetPosition method](#)  
[GetPowerFieldValue method](#)  
[GetPrinterInfo method](#)  
[GetProfileString method](#)  
[GetRGB method](#)  
[GetRightMouseMenu method](#)  
[GetSource method](#)  
[GetSpellStatus method](#)  
[GetSpellUserDictStatus method](#)  
[GetStandardButtonId method](#)  
[GetString method](#)  
[GetText method](#)  
[GetTOCProperties method](#)  
[GetUniqueName method](#)  
[GetUserClassNameFull method](#)  
[GetUserClassNameShort method](#)  
[GetWordMisspelled method](#)  
[Glossarize method](#)  
[GlossaryInsert method](#)  
[GlossaryOpen method](#)  
[GoTo method](#)  
[GoToBookmark method](#)  
[GoToClickHere method](#)  
[GoToContainer method](#)  
[GoToLayout method](#)  
[GotoNextParallelColumn method](#)  
[GoToObject method](#)  
[GoToPage method](#)  
[GoToSection method](#)  
[GoToTableCell method](#)

[GroupDivision method](#)

## H

[HandsOffStorage method](#)

[HasNamedProperty method](#)

[HasProperty method](#)

[Help method](#)

[Hide method](#)

[HideCaretAndSelection method](#)

[HidelconBar method](#)

[HidelconBars method](#)

[HideStatusBar method](#)

[HighlightToggle method](#)

[Hit method](#)

[HourGlass method](#)

## I

[ImportGraphic method](#)

[ImportPicture method](#)

[ImportWatermarkGraphic method](#)

[IndexAll method](#)

[InitFindAndReplace method](#)

[InsertBreak method](#)

[InsertBullet method](#)

[InsertClickHere method](#)

[InsertClickHereLink method](#)

[InsertColumnBreak method](#)

[InsertDate method](#)

[InsertDocInfo method](#)

[InsertDocument method](#)

[InsertField method](#)

[InsertFootnote method](#)

[InsertFrame method](#)

[InsertHardSpace method](#)

[InsertIndex method](#)

[InsertLink method](#)

[InsertMarker method](#)

[InsertNote method](#)

[InsertNumber method](#)

[InsertOleDivision method](#)

[InsertOne method](#)

[InsertPageBreak method](#)

[InsertPageLayout method](#)

[InsertPageNumber method](#)

[InsertPath method](#)

[InsertRowOrColumn method](#)

[InsertRuby method](#)

[InsertSection method](#)

[InsertSpecialTab method](#)

[InsertTab method](#)

[InsertText method](#)

[InsertTOC method](#)

[InternalCopy method](#)

[InternalCut method](#)

[InternalPaste method](#)

[InternetExtraFile method](#)



[Invalidate method](#)  
[InvalidateButton method](#)  
[InvalidateWholeBar method](#)  
[IsCaseExact method](#)  
[IsDataNameUsed method](#)  
[IsEmpty method](#)  
[IsExportedAsNotesFX method](#)  
[IsFilterTypePresent method](#)  
[IsMarkerEqualToSelection method](#)  
[IsPointWithin method](#)  
[IsTemporary method](#)  
[IsWMCCommandValid method](#)  
[Italic method](#)  
[Item method](#)

## J

## K

## L

[LabelCreate method](#)  
[LabelMerge method](#)  
[LeastRecentVersion method](#)  
[Link method](#)  
[LinkContainers method](#)  
[Localize method](#)  
[LowerCase method](#)

## M

[MacroEndRecord method](#)  
[MacroPlay method](#)  
[MacroRecord method](#)  
[MailDocument method](#)  
[MakeTableFromText method](#)  
[MakeUniqueLinkName method](#)  
[ManualFrame method](#)  
[ManualLinkFrames method](#)  
[ManualTable method](#)  
[Mark method](#)  
[MarkRevisionInsert method](#)  
[Maximize method](#)  
[Merge method](#)  
[MergeAddDataRecord method](#)  
[MergeSetDataFile method](#)  
[MergeStart method](#)  
[Messages method](#)  
[Minimize method](#)  
[MirrorPage method](#)  
[MorphSelectionToTable method](#)  
[MostRecentVersion method](#)  
[Move method](#)  
[MoveDivision method](#)  
[MoveDown method](#)  
[MoveItem method](#)

[MoveParagraph method](#)  
[MoveToBack method](#)  
[MoveToEnd method](#)  
[MoveToFront method](#)  
[MoveToStart method](#)  
[MoveUp method](#)

## N

[NewDivision method](#)  
[NewDocument method](#)  
[NewFrame method](#)  
[NewItem method](#)  
[NewItemByPosition method](#)  
[NewWindow method](#)  
[Next method](#)  
[NextCycleAlign method](#)  
[NextCycleAttribute method](#)  
[NextCycleBullet method](#)  
[NextCycleFont method](#)  
[NextCycleFontSize method](#)  
[NextCycleIndent method](#)  
[NextCycleNumber method](#)  
[NextCycleStyle method](#)  
[NextItem method](#)  
[NextToObject method](#)  
[NextVersion method](#)  
[NormalText method](#)  
[NumberOfVersions method](#)

## O

[Open method](#)  
[OpenDataFile method](#)  
[OpenDocument method](#)  
[OpenDocumentFromNotes method](#)  
[OpenFromStorage method](#)  
[OpenObject method](#)  
[OutlineMoveTextDown method](#)  
[OutlineMoveTextUp method](#)  
[OutlineStyleSequence method](#)

## P

[PageDown method](#)  
[PageUp method](#)  
[Paste method](#)  
[PasteLink method](#)  
[PasteSpecial method](#)  
[PColConnectCells method](#)  
[PColConnectRows method](#)  
[PColDisconnectCells method](#)  
[PColSelectColumn method](#)  
[PColSelectRow method](#)  
[PColSelectTable method](#)  
[Play method](#)  
[PrepareToDestroy method](#)  
[Present method](#)

[Previous method](#)  
[PreviousItem method](#)  
[PreviousVersion method](#)  
[Print method](#)  
[PrintOut method](#)  
[ProcessAccelKey method](#)  
[Promote method](#)  
[PromoteOutlineLevel method](#)  
[Purge method](#)

## Q

[Query method](#)  
[QueryDrop method](#)  
[QuickAlignFrame method](#)  
[QuickAlignTable method](#)  
[Quit method](#)

## R

[Read method](#)  
[Redo method](#)  
[Refresh method](#)  
[RegisterWPDataSet method](#)  
[Release method](#)  
[Remove method](#)  
[RemoveBookmark method](#)  
[RemoveChildFromLayout method](#)  
[RemoveDataFile method](#)  
[RemoveDdeLink method](#)  
[RemoveDepOnDocFile method](#)  
[RemoveDivision method](#)  
[RemoveEditor method](#)  
[RemoveIndexEntry method](#)  
[RemoveList method](#)  
[RemoveNamedProperty method](#)  
[RemovePersistentAccelerators method](#)  
[RemovePowerField method](#)  
[RemoveProperty method](#)  
[RemoveTOCEntry method](#)  
[Render method](#)  
[RenderClipBitmap method](#)  
[RenderClipDIB method](#)  
[RenderClipMetafile method](#)  
[RenderClipPalette method](#)  
[Repaint method](#)  
[Replace method](#)  
[ReplaceAll method](#)  
[ReplaceCmd method](#)  
[ReplaceContents method](#)  
[RequestAndProcessData method](#)  
[Reset method](#)  
[ResetFindAndReplace method](#)  
[ResetNumberOpts method](#)  
[ResetPrinting method](#)  
[Resize method](#)  
[Restore method](#)  
[RestorePreviousView method](#)

[RestoreWindowFromCleanScreen method](#)  
[RetrieveInternetFile method](#)  
[RetrieveInternetFileAndOpen method](#)  
[RevertToSaved method](#)  
[RevertToStyle method](#)  
[ReviewVersions method](#)  
[ReviseAcceptAll method](#)  
[ReviseCancelAll method](#)  
[RevisionAccept method](#)  
[RevisionAcceptLayoutChange method](#)  
[RevisionCancel method](#)  
[RevisionCancelLayoutChange method](#)  
[Run method](#)  
[RunScript method](#)

## S

[Save method](#)  
[SaveAs method](#)  
[SaveAsToNotes method](#)  
[SaveData method](#)  
[SaveDivision method](#)  
[SaveEnvelopeMaster method](#)  
[SaveGlossary method](#)  
[SaveMacro method](#)  
[SaveMergeDataFile method](#)  
[SaveSnapshot method](#)  
[SaveThumbnailBitmap method](#)  
[SaveToStorage method](#)  
[SaveToStorageComplete method](#)  
[SaveUserDefaults method](#)  
[SaveVersion method](#)  
[Select method](#)  
[SelectAStyle method](#)  
[SelectCell method](#)  
[SelectColumn method](#)  
[SelectCustomIcon method](#)  
[SelectDoc method](#)  
[SelectEntireCellRange method](#)  
[SelectEntireColumn method](#)  
[SelectEntirePCol method](#)  
[SelectEntirePColCellRange method](#)  
[SelectEntirePColColumn method](#)  
[SelectEntirePColRow method](#)  
[SelectEntireRow method](#)  
[SelectEntireTable method](#)  
[SelectMarker method](#)  
[SelectParagraph method](#)  
[SelectPColCell method](#)  
[SelectRow method](#)  
[SelectRuler method](#)  
[SelectSection method](#)  
[SelectSentence method](#)  
[SelectStandardIcon method](#)  
[SelectTable method](#)  
[SelectTableItem method](#)  
[SelectWord method](#)  
[SendFrameToBack method](#)

[SendFrameToBackOne method](#)  
[SendMailAndAttach method](#)  
[SendMailSelectedText method](#)  
[SetAllMargins method](#)  
[SetArrayProp method](#)  
[SetButtonText method](#)  
[SetCustomNumber method](#)  
[SetData method](#)  
[SetDocumentEpoch method](#)  
[SetFieldFormula method](#)  
[SetFocus method](#)  
[SetFormula method](#)  
[SetIndexInfo method](#)  
[SetJapanIndexInfo method](#)  
[SetLastUsedFilter method](#)  
[SetLineOneSide method](#)  
[SetLinesAllSides method](#)  
[SetLinkSource method](#)  
[SetMinimumOrigin method](#)  
[SetNamedProperty method](#)  
[SetNoFields method](#)  
[SetNumberingLevelInfo method](#)  
[SetOverrideGraphic method](#)  
[SetOverrideText method](#)  
[SetPageBottomMargin method](#)  
[SetPageTopMargin method](#)  
[SetPattern method](#)  
[SetPopupAlignment method](#)  
[SetPopupIndex method](#)  
[SetPopupWidth method](#)  
[SetPopupWidthType method](#)  
[SetPowerFieldValue method](#)  
[SetRGB method](#)  
[SetStorage method](#)  
[SetStyle method](#)  
[SetTableFill method](#)  
[SetTOCLevelContent method](#)  
[SetTOCLevelPageInfo method](#)  
[SetTOCProperties method](#)  
[SetUpEnvelopeMerge method](#)  
[Shade method](#)  
[Show method](#)  
[ShowAnyGreeting method](#)  
[ShowCaretAndSelection method](#)  
[ShowContainers method](#)  
[ShowCursor method](#)  
[ShowIconBars method](#)  
[ShowScrollBars method](#)  
[ShowStatusBar method](#)  
[SimulateButtonClick method](#)  
[Skip method](#)  
[SmallCaps method](#)  
[SmartSumColumn method](#)  
[SmartSumRow method](#)  
[SortParagraphs method](#)  
[SpecialView method](#)  
[SpellAddToUserDict method](#)  
[SpellClearSkippedWords method](#)

[SpellMarkSkippedWords method](#)  
[SpellSkipAll method](#)  
[SpellWord method](#)  
[Split method](#)  
[SplitDivision method](#)  
[SplitParagraph method](#)  
[SplitWindow method](#)  
[SRRReplace method](#)  
[Start method](#)  
[StartEditMergeData method](#)  
[StartEnvelopeDiv method](#)  
[StartFieldInsert method](#)  
[StoreInternetFile method](#)  
[StrikeThru method](#)  
[SubScript method](#)  
[Sum method](#)  
[SuperScript method](#)

## T

[TeamMail method](#)  
[TextNumber method](#)  
[TheoreticalScaledSize method](#)  
[Tile method](#)  
[TileWindowHorz method](#)  
[TileWindowVert method](#)  
[TimedSave method](#)  
[ToggleCleanScreen method](#)  
[ToggleIconBar method](#)  
[Type method](#)

## U

[Underline method](#)  
[Undo method](#)  
[UndoRedo method](#)  
[Unlink method](#)  
[UnLinkContainers method](#)  
[UnregisterWPDataSet method](#)  
[Update method](#)  
[UpdateFooterContent method](#)  
[UpdateFootersText method](#)  
[UpdateHeaderContent method](#)  
[UpdateHeadersText method](#)  
[UpdateIndexSection method](#)  
[UpdateLink method](#)  
[UpdateOle method](#)  
[UpdatePageSizeChange method](#)  
[UpdatePowerFields method](#)  
[UpdatePowerFieldsOnNew method](#)  
[UpdatePrinterBins method](#)  
[UpdatePrinterChanges method](#)  
[UpdateSelectedFields method](#)  
[UpdateTabs method](#)  
[UpdateTOC method](#)  
[UpdateUI method](#)  
[UpperCase method](#)

## V

ValidateValue method

## W

WMCommand method

WordCount method

WordUnderline method

Write method

WriteProfileString method

## X

## Y

## Z

## Word Pro Properties

A  
B  
C  
D  
E  
F  
G  
H  
I  
J  
K  
L  
M  
N  
O  
P  
Q  
R  
S  
T  
U  
V  
W  
X  
Y  
Z

### A

[Abilities property](#)  
[AbsoluteOn property](#)  
[AbsoluteTextOrientation property](#)  
[AbsoluteXPos property](#)  
[AbsoluteYPos property](#)  
[Accelerators property](#)  
[AccessRights property](#)  
[Action property](#)  
[Active property](#)  
[ActiveDocument property](#)  
[ActiveDocWindow property](#)  
[ActualEnumName property](#)  
[ActualName property](#)  
[Address1 property](#)  
[Address2 property](#)  
[Afid property](#)  
[AfidClassName property](#)  
[Align property](#)  
[Alignment property](#)  
[AlignmentChar property](#)  
[AlignmentType property](#)  
[AlignStyleName property](#)  
[All property](#)  
[AllBorders property](#)  
[AllowAlternateVerification property](#)  
[AllowListEdit property](#)  
[AllowListMultiValues property](#)  
[AlternateName property](#)  
[Always property](#)



[Amikake property](#)  
[AmikakeName property](#)  
[AmikakeType property](#)  
[Amount property](#)  
[AmountOfSpaceAbove property](#)  
[AmountOfSpaceAboveLine property](#)  
[AmountOfSpaceBelow property](#)  
[AmtTether property](#)  
[AmtToRotateContent property](#)  
[AmtToTetherFrom property](#)  
[AnyNumber property](#)  
[AnyOLEDDLinks property](#)  
[AppFoundry property](#)  
[Application property](#)  
[ApplicationWindow property](#)  
[ApplyAdjectivePos property](#)  
[ApplyAdjectNounPart property](#)  
[ApplyAgreementWithHereThere property](#)  
[ApplyAnglicisms property](#)  
[ApplyArchaicExpressions property](#)  
[ApplyArticleAgreement property](#)  
[ApplyBadComparatives property](#)  
[ApplyBadInflection property](#)  
[ApplyBadNoun property](#)  
[ApplyBadNounGender property](#)  
[ApplyBadPlural property](#)  
[ApplyBadPrepositions property](#)  
[ApplyBelgianExpression property](#)  
[ApplyBorrowedForeign property](#)  
[ApplyBureuaJargon property](#)  
[ApplyCalque property](#)  
[ApplyCapitalizationCheck property](#)  
[ApplyClauseErrors property](#)  
[ApplyCliches property](#)  
[ApplyColloquialExpression property](#)  
[ApplyCommonlyConfusedWords property](#)  
[ApplyCommonMisspell property](#)  
[ApplyComplexWords property](#)  
[ApplyConfusedEasy property](#)  
[ApplyConfusedEnglish property](#)  
[ApplyConfusedHard property](#)  
[ApplyConfusedMedium property](#)  
[ApplyConfusedVerb property](#)  
[ApplyConsecutiveNouns property](#)  
[ApplyContractions property](#)  
[ApplyDerogatory property](#)  
[ApplyDifferentPrep property](#)  
[ApplyDoubleNegative property](#)  
[ApplyDoublePlural property](#)  
[ApplyDoubleWordCheck property](#)  
[ApplyElision property](#)  
[ApplyEnglishDerived property](#)  
[ApplyEnglishWords property](#)  
[ApplyExotic property](#)  
[ApplyExtraPrepositionCheck property](#)  
[ApplyFalseFriend property](#)  
[ApplyFemaleOccupation property](#)  
[ApplyFixedExpression property](#)

[ApplyForeignWord property](#)  
[ApplyFormalTerms property](#)  
[ApplyFormatErrors property](#)  
[ApplyGallicisms property](#)  
[ApplyGenderExpressions property](#)  
[ApplyGermanisms property](#)  
[ApplyHomoGraphs property](#)  
[ApplyHomonyms property](#)  
[ApplyHomonymsEasy property](#)  
[ApplyHomonymsHard property](#)  
[ApplyHomoPhone1 property](#)  
[ApplyHomoPhone2 property](#)  
[ApplyHomoPhones property](#)  
[ApplyIncorrectPlural property](#)  
[ApplyInformalExpressions property](#)  
[ApplyJargonWords property](#)  
[ApplykSplitInfinitives property](#)  
[ApplyLowercaseAdjective property](#)  
[ApplyLowercaseColor property](#)  
[ApplyLowercaseNumbers property](#)  
[ApplyLowercasePhrases property](#)  
[ApplyLowercasePronouns property](#)  
[ApplyMassVsCount property](#)  
[ApplyMisspelledExpressions property](#)  
[ApplyMisspelledForeignExpressions property](#)  
[ApplyMisspelledItalian property](#)  
[ApplyMisspelledWords property](#)  
[ApplyMisusedWords property](#)  
[ApplyNonStandardExpression property](#)  
[ApplyNonStandardModifiers property](#)  
[ApplyNoudModifierOrderCheck property](#)  
[ApplyNounConsistency property](#)  
[ApplyNounPhraseAgree property](#)  
[ApplyNSAdjective property](#)  
[ApplyNSClause property](#)  
[ApplyNSCompare property](#)  
[ApplyNSContract property](#)  
[ApplyNSGeography property](#)  
[ApplyNSInflection property](#)  
[ApplyNSNegation property](#)  
[ApplyNSPrep property](#)  
[ApplyNSPronoun property](#)  
[ApplyNSSpell property](#)  
[ApplyNSUsage property](#)  
[ApplyNSVerbForm property](#)  
[ApplyOpenClosedSpelling property](#)  
[ApplyOpenUsage property](#)  
[ApplyOverusedPhrases property](#)  
[ApplyPassiveVerbErrors property](#)  
[ApplyPostClitAgree property](#)  
[ApplyPrepExpression property](#)  
[ApplyPrepositionalPhrases property](#)  
[ApplyPretentiousWords property](#)  
[ApplyPronounErrors property](#)  
[ApplyPunctuationErrors property](#)  
[ApplyRedundantExpressions property](#)  
[ApplyRegionalExpression property](#)  
[ApplyRelatedWord property](#)

[ApplySensitiveExp property](#)  
[ApplySexistExpressions property](#)  
[ApplySpellStandard property](#)  
[ApplyStockPhrase property](#)  
[ApplyStyleParameters property](#)  
[ApplySubjectVerbAgreement property](#)  
[ApplySwedishGender property](#)  
[ApplySwedishNegation property](#)  
[ApplySwedishUsage property](#)  
[ApplyTrite property](#)  
[ApplyTwoGender property](#)  
[ApplyTypicalMisspell property](#)  
[ApplyUnGrammaticalExpressions property](#)  
[ApplyVagueQuantifiers property](#)  
[ApplyVerbGroupConsistency property](#)  
[ApplyWeakModifiers property](#)  
[ApplyWordChoice property](#)  
[ApplyWordCompoundingCheck property](#)  
[ApplyWordConfusion property](#)  
[ApplyWordGender property](#)  
[ApplyWordParts property](#)  
[ApplyWordyPhraseCheck property](#)  
[AppViewPrefs property](#)  
[AreDisabledIconsGrayed property](#)  
[Ascent property](#)  
[AsciiCodePage property](#)  
[AsciiCRLFType property](#)  
[AtBeginning property](#)  
[AtBeginningOfLine property](#)  
[AtBeginningOfObject property](#)  
[AtBeginningOfParagraph property](#)  
[AtBeginningOfStream property](#)  
[AtBeginningOfWord property](#)  
[AtEnd property](#)  
[AtEndOfLine property](#)  
[AtEndOfObject property](#)  
[AtEndOfParagraph property](#)  
[AtEndOfStream property](#)  
[AtEndOfWord property](#)  
[Attributes property](#)  
[AttrStyleName property](#)  
[AuthorName property](#)  
[AutoBackup property](#)  
[AutoHyphenate property](#)  
[AutomaticLink property](#)  
[AutoRunMacro property](#)  
[AutoSave property](#)  
[AutoSaveMinutes property](#)  
[AutoVersion property](#)

## **B**

[BackColor property](#)  
[BackColorIndex property](#)  
[Background property](#)  
[BackgroundPrintingOn property](#)  
[BackgroundSpellingOn property](#)  
[BackupPath property](#)

[BackupPaths property](#)  
[Bags property](#)  
[BaseLineOffset property](#)  
[BaselineOffset property](#)  
[BaseTable property](#)  
[BaseTables property](#)  
[BaseURL property](#)  
[BinName property](#)  
[BinNames property](#)  
[BinNumber property](#)  
[BlockPaint property](#)  
[Blue property](#)  
[BodyOnly property](#)  
[Bold property](#)  
[BookletPrinting property](#)  
[BookmarkManager property](#)  
[Bookmarks property](#)  
[BookmarksByMarkerName property](#)  
[BorderLines property](#)  
[BorderOffset property](#)  
[BorderStyleName property](#)  
[BottomBorder property](#)  
[BottomExternalMargin property](#)  
[BottomIntArea property](#)  
[Breaks property](#)  
[BreaksStyleName property](#)  
[Bullet property](#)  
[BulletFonts property](#)  
[BulletStyleName property](#)

## C

[CanCreatePreviewBitmap property](#)  
[CanEditProperty property](#)  
[CanEmbed property](#)  
[CanWePrint property](#)  
[Caption property](#)  
[Case property](#)  
[Cell property](#)  
[CellEngine property](#)  
[CellEngines property](#)  
[CellLayout property](#)  
[CellLayouts property](#)  
[CellLayoutStyles property](#)  
[CellStyleName property](#)  
[Center property](#)  
[CenteredHorz property](#)  
[CenteredVert property](#)  
[Changed property](#)  
[ChangeKeyboardToLanguage property](#)  
[ChangeTextToMatchkeyboard property](#)  
[CharacterBorder property](#)  
[CharacterBorderName property](#)  
[CharacterSet property](#)  
[CharacterSetName property](#)  
[CharacterStyle property](#)  
[CharacterStyleName property](#)  
[CharacterStyles property](#)

[CharSet property](#)  
[Checked property](#)  
[CheckForScriptEnumError property](#)  
[CheckForScriptPropertyError property](#)  
[ChildLayouts property](#)  
[CityState property](#)  
[ClassId property](#)  
[ClassName property](#)  
[CleanScreenMode property](#)  
[CleanScreenOnStartUp property](#)  
[ClickHerePrompts property](#)  
[ClickHeres property](#)  
[ClientHeight property](#)  
[ClientWidth property](#)  
[ClientWndHeight property](#)  
[ClientWndWidth property](#)  
[CloseDocMacroName property](#)  
[CodePage property](#)  
[Collapsible property](#)  
[Collate property](#)  
[Color property](#)  
[ColorOverride property](#)  
[ColumnBalance property](#)  
[ColumnGap property](#)  
[ColumnLayouts property](#)  
[ColumnNumber property](#)  
[ColumnWidth property](#)  
[CommentColor property](#)  
[Company property](#)  
[Condition property](#)  
[ConditionType property](#)  
[ConnectedLayouts property](#)  
[Container property](#)  
[Content property](#)  
[ContentHeight property](#)  
[ContentLayouts property](#)  
[ContentName property](#)  
[Contents property](#)  
[ContentStyleName property](#)  
[ContentType property](#)  
[ContentWidth property](#)  
[ContextMenuOptions property](#)  
[ContinuedFromAlignment property](#)  
[ContinuedFromMessage property](#)  
[ContinuedFromStory property](#)  
[ContinuedOnAlignment property](#)  
[ContinuedOnMessage property](#)  
[ContinuedOnStory property](#)  
[ConvertOnNew property](#)  
[Copies property](#)  
[Count property](#)  
[CountBlankLines property](#)  
[CreatePreviewBitmap property](#)  
[CreationDateString property](#)  
[CreationTimeString property](#)  
[CreationTimeValue property](#)  
[Crop property](#)  
[CurrentCell property](#)

[CurrentColumn property](#)  
[CurrentEditor property](#)  
[CurrentLanguage property](#)  
[CurrentRow property](#)  
[CurrentRunningScriptName property](#)  
[CurrentRunningScriptPath property](#)  
[CurrentVersion property](#)  
[CustomLength property](#)  
[CustomViewLevel property](#)

## D

[DataFileFieldNames property](#)  
[DataFileName property](#)  
[DataFormat property](#)  
[DataNames property](#)  
[DateCreatedValue property](#)  
[DateRevisedValue property](#)  
[DdeEnabled property](#)  
[DdeLinkManager property](#)  
[DdeLinks property](#)  
[DdeLinksFromMarker property](#)  
[DdeOutboundInfo property](#)  
[DebugVariable property](#)  
[DefaultBinName property](#)  
[DefaultCellStyleDescription property](#)  
[DefaultColumnName property](#)  
[DefaultDropCapStyleDescription property](#)  
[DefaultFilePath property](#)  
[DefaultFrameStyleDescription property](#)  
[DefaultLatinFont property](#)  
[DefaultLeftColumnName property](#)  
[DefaultNewCategory property](#)  
[DefaultNonLatinFont property](#)  
[DefaultPageHeight property](#)  
[DefaultPageStyleDescription property](#)  
[DefaultPageWidth property](#)  
[DefaultPitch property](#)  
[DefaultRightColumnName property](#)  
[DefaultTableStyleDescription property](#)  
[DefaultTextStyleDescription property](#)  
[DefCellStyleName property](#)  
[DefColWidth property](#)  
[Definition property](#)  
[DefRowHeight property](#)  
[DelayedGreeting property](#)  
[DeleteFont property](#)  
[DemandLoad property](#)  
[Descent property](#)  
[Description property](#)  
[DescriptionFileName property](#)  
[DiagonalLines property](#)  
[DirectionDown property](#)  
[DirectionLeft property](#)  
[DirectionRight property](#)  
[DirectionUp property](#)  
[DirectiveColor property](#)  
[DisableClickHeres property](#)

[DisableConsistencyCheck property](#)  
[DisableExportToNotes property](#)  
[DisableVersionReview property](#)  
[DisplayAsIcon property](#)  
[DistanceFromLeftMargin property](#)  
[Division property](#)  
[DivisionInfo property](#)  
[DivisionName property](#)  
[DivisionNames property](#)  
[DivisionOptions property](#)  
[Divisions property](#)  
[DivisionsRequired property](#)  
[DocControl property](#)  
[DocControlRestrictedToEditor property](#)  
[DocInfo property](#)  
[DocOptions property](#)  
[DocPath property](#)  
[DocSize property](#)  
[Document property](#)  
[DocumentLevel property](#)  
[DocumentPaths property](#)  
[Documents property](#)  
[DocVersionID property](#)  
[DocWindows property](#)  
[DoInitialCaps property](#)  
[DoSmartQuotes property](#)  
[DoubleUnderline property](#)  
[DragDropOn property](#)  
[DriverName property](#)  
[DropCapPosition property](#)  
[DropCaps property](#)  
[DropCapStyleName property](#)  
[DropCapStyles property](#)  
[DuplexType property](#)

## E

[Editable property](#)  
[EditorInitials property](#)  
[EditorManager property](#)  
[EditorName property](#)  
[Editors property](#)  
[EditorVerificationType property](#)  
[EffectiveColumnWidth property](#)  
[EMail property](#)  
[Embedded property](#)  
[EmbedFonts property](#)  
[Enabled property](#)  
[EncryptPassword property](#)  
[EncryptPassword2 property](#)  
[EndingColOfSelection property](#)  
[EndingRowOfSelection property](#)  
[EndnoteDivisionGroupNum property](#)  
[EndnoteDivisionNum property](#)  
[EndnoteDocNum property](#)  
[Endnotes property](#)  
[Enumeration property](#)  
[Epoch property](#)

[EqnFontHeight property](#)  
[ErrorColor property](#)  
[ExcludeRectBottom property](#)  
[ExcludeRectLeft property](#)  
[ExcludeRectRight property](#)  
[ExcludeRectTop property](#)  
[Expandable property](#)  
[ExportToNotesFX property](#)  
[ExternalFileID property](#)  
[ExternalFileName property](#)  
[ExternallyControlledUndo property](#)  
[ExternalName property](#)  
[ExternalType property](#)

## F

[FaceName property](#)  
[FaceNames property](#)  
[FaceStyleName property](#)  
[FastFormatType property](#)  
[FaxNumber property](#)  
[FieldDelimiter property](#)  
[FieldDelimiterText property](#)  
[FieldManager property](#)  
[FieldNumber property](#)  
[Fields property](#)  
[FieldType property](#)  
[FileFormat property](#)  
[FileProtectionType property](#)  
[FilesToCompare property](#)  
[FileType property](#)  
[FillerPageText property](#)  
[FillStyle property](#)  
[Filter property](#)  
[FilterHelper property](#)  
[FilterName property](#)  
[FindAndReplace property](#)  
[FindExactCase property](#)  
[FindFont property](#)  
[FindForwardDirection property](#)  
[FindString property](#)  
[FindStyleName property](#)  
[FindWithProperties property](#)  
[FinishedSpellChecking property](#)  
[First property](#)  
[FirstChild property](#)  
[FirstDivision property](#)  
[FirstName property](#)  
[FirstPage property](#)  
[FirstSpellString property](#)  
[FitType property](#)  
[FixAcronymns property](#)  
[FixBullets property](#)  
[FixMargins property](#)  
[Font property](#)  
[FontColor property](#)  
[FontMatching property](#)  
[FontMetrics property](#)



[FontName property](#)  
[FontSize property](#)  
[FontStyleName property](#)  
[FontUnitName property](#)  
[FontUnits property](#)  
[Footer property](#)  
[Footers property](#)  
[FooterStyleName property](#)  
[FooterStyles property](#)  
[FootnoteContSep property](#)  
[FootnoteLayouts property](#)  
[FootnoteNumbering property](#)  
[FootnoteOptions property](#)  
[Footnotes property](#)  
[FootnoteSeparator property](#)  
[ForeColorIndex property](#)  
[Format property](#)  
[FormatCheckPreferences property](#)  
[FormatType property](#)  
[Formula property](#)  
[Foundry property](#)  
[Frame property](#)  
[Frames property](#)  
[FrameStyleName property](#)  
[FrameStyles property](#)  
[FreeMenus property](#)  
[FullName property](#)

## G

[GapBetweenPanels property](#)  
[GetAfidHelpFileName property](#)  
[GetAfidHelpInfo property](#)  
[GetButtonId property](#)  
[GetButtonType property](#)  
[GetConversationHandle property](#)  
[GetFilterExtension property](#)  
[GetFilterExtForDialogBox property](#)  
[GetFilterId property](#)  
[GetFormatName property](#)  
[GetHomeDirectory property](#)  
[GetItemName property](#)  
[GetRedoWhatDesc property](#)  
[GetServerName property](#)  
[GetStatus property](#)  
[GetTopicName property](#)  
[GetUndoWhatDesc property](#)  
[GetValue property](#)  
[GlossaryDataFileName property](#)  
[GlossaryDataFiles property](#)  
[GlossaryDataPaths property](#)  
[GlossaryPath property](#)  
[Glossarys property](#)  
[GoToIndexLastItemAccessed property](#)  
[GoToItemName property](#)  
[GotoPageLoadInBackground property](#)  
[GoToPageNumber property](#)  
[GoToSelection property](#)

[GrammarFormalityLevel property](#)  
[GrammarOptions property](#)  
[GrammarProofLevel property](#)  
[GrammarSetName property](#)  
[Graphic property](#)  
[GraphicExports property](#)  
[GraphicImports property](#)  
[GraphicOleObject property](#)  
[GraphicPath property](#)  
[GraphicPaths property](#)  
[Graphics property](#)  
[Green property](#)  
[Greeting property](#)  
[GridDistance property](#)  
[GridType property](#)  
[Groups property](#)  
[Gutter property](#)

## H

[Hang property](#)  
[HasContents property](#)  
[HasFocus property](#)  
[HasIndex property](#)  
[HasLocalTabs property](#)  
[HasTabs property](#)  
[HasTOC property](#)  
[Header property](#)  
[Headers property](#)  
[HeaderStyleName property](#)  
[HeaderStyles property](#)  
[Heading property](#)  
[Height property](#)  
[HelpText property](#)  
[HiddenMode property](#)  
[HideFormula property](#)  
[HideOutlineLevels property](#)  
[HighLightMode property](#)  
[HiLiteColor property](#)  
[HorizontalSplitWindow property](#)  
[HorizScrollBarVisible property](#)  
[HorzRuler property](#)  
[Hwnd property](#)  
[HyphenateLastWordInColumnOrPage property](#)  
[HyphenateLastWordInPara property](#)  
[HyphenationOptions property](#)  
[HyphZoneAfter property](#)  
[HyphZoneBefore property](#)

## I

[IconBarManager property](#)  
[IconBarPositionState property](#)  
[IconBars property](#)  
[IconBarSets property](#)  
[IconHelpText property](#)  
[IconPath property](#)  
[IconPaths property](#)

[IconScript property](#)  
[IconSetName property](#)  
[IconShowingBeforeCleanScreen property](#)  
[IconSize property](#)  
[ID property](#)  
[IdentifierColor property](#)  
[IDispatch property](#)  
[IgnoreSoftHyphens property](#)  
[IgnoreTab property](#)  
[IncludeInitialsInNotes property](#)  
[IncludeList property](#)  
[Indent property](#)  
[IndentFromLeft property](#)  
[IndentStyleName property](#)  
[IndexAlphabeticSeparator property](#)  
[IndexDivision property](#)  
[IndexIndentType property](#)  
[IndexParent property](#)  
[IndexPrimaryStyleName property](#)  
[IndexRange property](#)  
[IndexSecondaryStyleName property](#)  
[IndexSection property](#)  
[IndexSeparatorStyleName property](#)  
[IndexSource property](#)  
[InDocument property](#)  
[InfoBoxSelectionText property](#)  
[InitializeRoute property](#)  
[Initialize property](#)  
[InitialsForFilters property](#)  
[InsertFont property](#)  
[InsertionMode property](#)  
[Interactive property](#)  
[Interval property](#)  
[IsActionOnButtonDown property](#)  
[IsActive property](#)  
[IsAdopted property](#)  
[IsAmikake property](#)  
[IsAmiProTableImport property](#)  
[IsAsciiCRLF property](#)  
[IsAsciiKeepStyle property](#)  
[IsAutoGrow property](#)  
[IsBadReference property](#)  
[IsBorder property](#)  
[IsBothSidesEqual property](#)  
[IsBottomAligned property](#)  
[IsBreakable property](#)  
[IsBubbleHelp property](#)  
[IsCellMenuEnabled property](#)  
[IsCentered property](#)  
[IsChanged property](#)  
[IsChangedOtherThanLinkTo property](#)  
[IsChangedSinceTimeSave property](#)  
[IsChangedToLinkTo property](#)  
[IsChartLink property](#)  
[IsChildSpannable property](#)  
[IsCollapsed property](#)  
[IsCollapsible property](#)  
[IsColumnBreakable property](#)

[IsColumnBreakAfter property](#)  
[IsColumnBreakBefore property](#)  
[IsComplex property](#)  
[IsConnected property](#)  
[IsContentReplaceable property](#)  
[IsContents property](#)  
[IsContinuedFrom property](#)  
[IsContinuedOn property](#)  
[IsCumulative property](#)  
[IsDebug property](#)  
[IsDisableWarningMessages property](#)  
[IsDisplayMisspelled property](#)  
[IsDivisionExternal property](#)  
[IsDocLoading property](#)  
[IsDocumentInRoute property](#)  
[IsDoubleWordError property](#)  
[IsDraw property](#)  
[IsEmpty property](#)  
[IsEmptyDoc property](#)  
[IsEnabled property](#)  
[IsEndnoteDivision property](#)  
[IsEquation property](#)  
[IsErrorChecking property](#)  
[IsExpandDown property](#)  
[IsExpandLeft property](#)  
[IsExpandRight property](#)  
[IsExpandUp property](#)  
[IsExportedToNotesFX property](#)  
[IsExportedToOldNotesFX property](#)  
[IsExternalFile property](#)  
[IsFixedLength property](#)  
[IsFooter property](#)  
[IsFrameMenuEnabled property](#)  
[IsGrabBar property](#)  
[IsGrammarError property](#)  
[IsGraphicMenuEnabled property](#)  
[IsHangover property](#)  
[IsHeader property](#)  
[IsHiddenMark property](#)  
[IsHideHeaderFooter property](#)  
[IsHighlightNote property](#)  
[IsHonorProtected property](#)  
[IsHorizontalScrollBarCleanScrn property](#)  
[IsHoverHelp property](#)  
[IsIconDepressible property](#)  
[IsInBulletEditMode property](#)  
[IsIndex property](#)  
[IsInDraft property](#)  
[IsInOutline property](#)  
[IsInPageSort property](#)  
[IsIntersectSiblings property](#)  
[IsLastStop property](#)  
[IsLesser property](#)  
[IsLinked property](#)  
[IsLocal property](#)  
[IsLocked property](#)  
[IsLockedForRevisions property](#)  
[IsLotusChart property](#)

IsMarginSameAsParent property  
IsMarginsInColor property  
IsMarkerValid property  
IsMirrorImage property  
IsMisspelled property  
IsNotCopyable property  
IsNotCopyImage property  
IsNotGroupable property  
IsNoUICommAllowed property  
IsOnClipboard property  
IsOpen property  
IsOverridden property  
IsOverride property  
IsPageBreak property  
IsPageBreakAfter property  
IsPageBreakBefore property  
IsPageBreakMarks property  
IsPageBreakWithin property  
IsParaDemandLoad property  
IsParagraphNumberingDown property  
IsParagraphParent property  
IsParallelColumnsMenuEnabled property  
IsPartOfGroup property  
IsPersistent property  
IsPrePrintedForm property  
IsPrintable property  
IsPrintClickHereBlocks property  
IsPrivate property  
IsPrompting property  
IsProtected property  
IsRegistered property  
IsRepeat property  
IsReplaceable property  
IsReplacement property  
IsResetParagraphNumber property  
IsRevision property  
IsRevisionMark property  
IsSame property  
IsSave property  
IsScalable property  
IsScalableBorder property  
IsScripting property  
IsScrollable property  
IsShowing property  
IsSingleClickEntry property  
IsSizable property  
IsSizingViaMouse property  
IsSmartCorrectEnabled property  
IsSmartEditEnabled property  
IsSmartShadeEnabled property  
IsSnapTo property  
IsSortFromEnd property  
IsSpellBarUp property  
IsSpellMode property  
IsSqueeze property  
IsStyle property  
IsStyleSheet property  
IsSymbolic property

[IsTableHeading property](#)  
[IsTemp property](#)  
[IsTextLocked property](#)  
[IsTextMenuEnabled property](#)  
[IsTOC property](#)  
[IsTrueType property](#)  
[IsUndoOn property](#)  
[IsUpdateAutomatic property](#)  
[IsUpdateIndex property](#)  
[IsUpdateTOC property](#)  
[IsValid property](#)  
[IsVerticalScrollBarCleanScrn property](#)  
[IsViewAnchor property](#)  
[IsViewBookmarks property](#)  
[IsViewClickHereBlocks property](#)  
[IsViewColGuides property](#)  
[IsViewColumnBreakMarks property](#)  
[IsViewDDELinks property](#)  
[IsViewHorzRuler property](#)  
[IsViewHorzScrollBar property](#)  
[IsViewMenuCleanScrn property](#)  
[IsViewNotes property](#)  
[IsViewOutlineFlowToScreen property](#)  
[IsViewOutlineIndent property](#)  
[IsViewPageLayoutMarks property](#)  
[IsViewParallelColumnBorder property](#)  
[IsViewPictures property](#)  
[IsViewReturnIconCleanScrn property](#)  
[IsViewReturns property](#)  
[IsViewRubyMarks property](#)  
[IsViewRulerMarks property](#)  
[IsViewSectionBreakMarks property](#)  
[IsViewSectionTabs property](#)  
[IsViewSmartIconsCleanScrn property](#)  
[IsViewStatusBarCleanScrn property](#)  
[IsViewStatusSpell property](#)  
[IsViewTableCellSelection property](#)  
[IsViewTableGridLines property](#)  
[IsViewTableHeadings property](#)  
[IsViewTabs property](#)  
[IsViewThumbBarCleanScrn property](#)  
[IsViewTitleBarCleanScrn property](#)  
[IsViewVertRuler property](#)  
[IsViewVertScrollBar property](#)  
[IsWordProChart property](#)  
[IsWorkingDir property](#)  
[Italic property](#)  
[Items property](#)

## J

[Join property](#)  
[JoinCorners property](#)  
[JoinHeight property](#)  
[JoinType property](#)  
[JoinWidth property](#)  
[Justifiable property](#)

## K

[KeepWithNext property](#)  
[KeepWithPrev property](#)  
[KeyboardLanguage property](#)  
[KeywordColor property](#)  
[Keywords property](#)  
[Kinsoku property](#)

## L

[LandscapeMode property](#)  
[Language property](#)  
[Last property](#)  
[LastChild property](#)  
[LastDivision property](#)  
[LastDocPath property](#)  
[LastEditorName property](#)  
[LastGraphicPath property](#)  
[LastGraphicType property](#)  
[LastMacroPath property](#)  
[LastName property](#)  
[LastPage property](#)  
[LastUsedDateFormula property](#)  
[Layer property](#)  
[LayerName property](#)  
[Layout property](#)  
[LayoutName property](#)  
[LayoutOverride property](#)  
[Layouts property](#)  
[LeaderDotDashChar property](#)  
[LeaderDotDotChar property](#)  
[LeaderDotType property](#)  
[LeaderDotUnderscoreChar property](#)  
[LeadingText property](#)  
[Left property](#)  
[LeftBorder property](#)  
[LeftExternalMargin property](#)  
[LeftIntArea property](#)  
[LeftPage property](#)  
[LeftTopCellId property](#)  
[Length property](#)  
[Level property](#)  
[LineLocation property](#)  
[LineMix property](#)  
[LineNumberOptions property](#)  
[LinePlacement property](#)  
[LinesSpacedEveryNthUnit property](#)  
[LineValid property](#)  
[LinkAvailable property](#)  
[LinkDisplayName property](#)  
[LinkDisplayNameFileLength property](#)  
[Linked property](#)  
[LinkedFileName property](#)  
[LinkFrame property](#)  
[LinkGraphic property](#)  
[ListCount property](#)  
[LoadFilesMaximized property](#)

[Location property](#)  
[Locked property](#)  
[LockForNotesUserName property](#)  
[LockResult property](#)  
[Locks property](#)  
[LowerCase property](#)  
[LowerCaseAscent property](#)  
[LwpMenuBar property](#)

## M

[Macro property](#)  
[MacroName property](#)  
[MacroPath property](#)  
[MacroPaths property](#)  
[MacroStatus property](#)  
[MailRouting property](#)  
[MailRoutingPtr property](#)  
[MaintainAspectRatio property](#)  
[MaintainEditor property](#)  
[MarginBottom property](#)  
[MarginColor property](#)  
[MarginLeft property](#)  
[MarginRight property](#)  
[MarginTop property](#)  
[MarkCharacter property](#)  
[MarkerClass property](#)  
[MarkerName property](#)  
[Markers property](#)  
[MarkPosition property](#)  
[MarkType property](#)  
[Master property](#)  
[MasterName property](#)  
[MatchType property](#)  
[MaxBottomBorder property](#)  
[MaxBottomGutter property](#)  
[MaxContentHeight property](#)  
[MaxContentWidth property](#)  
[MaxHorzPaneDistance property](#)  
[MaxHyphLines property](#)  
[MaxIdenticalConsecSentOpens property](#)  
[MaxIdenticalSentOpensWithin10 property](#)  
[MaximizeOnStartUp property](#)  
[MaximumWordsinaSentence property](#)  
[MaxLeftBorder property](#)  
[MaxLeftGutter property](#)  
[MaxNumColsAllowed property](#)  
[MaxNumRowsAllowed property](#)  
[MaxRightBorder property](#)  
[MaxRightGutter property](#)  
[MaxSplitCols property](#)  
[MaxSplitRows property](#)  
[MaxTopBorder property](#)  
[MaxTopGutter property](#)  
[MaxVertPaneDistance property](#)  
[MergeFileType property](#)  
[MergeInfoPtr property](#)  
[MergeOptions property](#)



[MergeStepNumber property](#)  
[MergeToFile property](#)  
[MetafilePict property](#)  
[MinBottomMargin property](#)  
[MinHeight property](#)  
[MinLeftMargin property](#)  
[MinRightMargin property](#)  
[MinTopMargin property](#)  
[ModifiedDateString property](#)  
[ModifiedTimeString property](#)  
[ModifiedTimeValue property](#)  
[MouseButtonForManipulation property](#)  
[MouseButtonForSelection property](#)  
[MouseProperty property](#)  
[MultiCellPasteOn property](#)  
[MultiCompareParaTag property](#)  
[MultiCompareParaTagSet property](#)

## N

[Name property](#)  
[NameBasedOnStyle property](#)  
[NameForFilters property](#)  
[Negative property](#)  
[NewDocMacroName property](#)  
[NewFile property](#)  
[NewVersion property](#)  
[NextClickHere property](#)  
[NextName property](#)  
[NextNeighbor property](#)  
[NextSection property](#)  
[NextSpellString property](#)  
[NextStyleName property](#)  
[NextText property](#)  
[NoHyphenate property](#)  
[NonUserDocument property](#)  
[Normal property](#)  
[NormalParagraph property](#)  
[NoteColor property](#)  
[NoteLayouts property](#)  
[NotesFlow property](#)  
[Number property](#)  
[NumberEveryNthLine property](#)  
[NumberFound property](#)  
[Numbering property](#)  
[NumberingPosition property](#)  
[NumberingStyleName property](#)  
[NumberOfCharacters property](#)  
[NumberOfDataFields property](#)  
[NumberOfLines property](#)  
[NumberOfMergeConditions property](#)  
[NumberOfReplacements property](#)  
[NumberOfRevisions property](#)  
[NumberSequenceName property](#)  
[NumberWhichLines property](#)  
[NumCharsInDoc property](#)  
[NumCharsInParagraph property](#)  
[NumCols property](#)

[NumColsSpannedOneCell property](#)  
[NumContainers property](#)  
[NumDecimalPlaces property](#)  
[NumericFormat property](#)  
[NumFields property](#)  
[NumLinesOfSpace property](#)  
[NumLinesOfSpaceAbove property](#)  
[NumLinesOfSpaceAboveLine property](#)  
[NumLinesOfSpaceBelow property](#)  
[NumOfRecentFiles property](#)  
[NumPagesInDoc property](#)  
[NumParagraphs property](#)  
[NumRows property](#)  
[NumRowsSpannedOneCell property](#)  
[NumRowsThatFit property](#)  
[NumRowsToFit property](#)  
[NumTabs property](#)  
[NumUndoLevels property](#)  
[NumWindowsViewingDoc property](#)  
[NumWordsInDoc property](#)

## O

[Object property](#)  
[ObjectType property](#)  
[Oblique property](#)  
[OCXDesignMode property](#)  
[Offset property](#)  
[Ole1Object property](#)  
[OLEAutomation property](#)  
[OleContainerDocName property](#)  
[OLEEnabled property](#)  
[OleMinHeight property](#)  
[OleObject property](#)  
[OleObjects property](#)  
[OleObjectSize property](#)  
[OpenDocMacroName property](#)  
[OpenDocsVisible property](#)  
[OpenExistingFileInWelcomeBox property](#)  
[OpenReadOnly property](#)  
[OpionDeclare property](#)  
[Options property](#)  
[OrigFileType property](#)  
[OrigHeight property](#)  
[OrigWidth property](#)  
[Outline property](#)  
[OutlineBorderLines property](#)  
[OutlineButtons property](#)  
[OutlineHeadingButtonsOnly property](#)  
[OutlineLevel property](#)  
[OutlineOnlyHeadingsWhenCollapsed property](#)  
[OutlineSeqItems property](#)  
[OutlineStyleSequences property](#)  
[OutputToFile property](#)  
[Override property](#)  
[Overstrike property](#)  
[OverstrikeCharacter property](#)

## P

[Page property](#)  
[PageNo property](#)  
[PageNum property](#)  
[PageNumber property](#)  
[PageNumberAsText property](#)  
[PageNumberStyle property](#)  
[PageNumFirstPageShowing property](#)  
[PageNumLastPageShowing property](#)  
[PageOrder property](#)  
[Pages property](#)  
[PageStyleName property](#)  
[PageStyles property](#)  
[PageToUseLayoutOn property](#)  
[PairKerning property](#)  
[PaneColor property](#)  
[PaperNames property](#)  
[ParagraphBorder property](#)  
[ParagraphHasDropCap property](#)  
[ParagraphHasText property](#)  
[ParagraphStyle property](#)  
[ParagraphStyleName property](#)  
[ParagraphStyles property](#)  
[ParagraphSymbolChar property](#)  
[ParallelColumns property](#)  
[Parent property](#)  
[ParentMenuHWND property](#)  
[ParentName property](#)  
[Partial property](#)  
[Path property](#)  
[PathName property](#)  
[Pattern property](#)  
[Percentage property](#)  
[PersonalData1 property](#)  
[PersonalData2 property](#)  
[PersonalData3 property](#)  
[PersonalData4 property](#)  
[PhoneNumber property](#)  
[PitchAndFamily property](#)  
[Placement property](#)  
[Plain property](#)  
[PointSize property](#)  
[Position property](#)  
[PositionType property](#)  
[PositionXInContainer property](#)  
[PositionXOnPage property](#)  
[PositionYInContainer property](#)  
[PositionYOnPage property](#)  
[PowerField property](#)  
[PowerFields property](#)  
[Preferences property](#)  
[Prefix property](#)  
[Presentation property](#)  
[PrettyPrinting property](#)  
[PreviousClickHere property](#)  
[PreviousName property](#)  
[PreviousNeighbor property](#)

[PrintDestination property](#)  
[PrintDocDescription property](#)  
[PrinterName property](#)  
[PrintGraphics property](#)  
[PrintInBackground property](#)  
[PrintManager property](#)  
[PrintPagesFrom property](#)  
[PrintPagesTo property](#)  
[PrintPageType property](#)  
[PrintRange property](#)  
[PrintSettings property](#)  
[Private property](#)  
[ProgID property](#)  
[Prompt property](#)  
[PromptHidden property](#)  
[ProtectedMode property](#)

## Q

[QueueName property](#)

## R

[ReadCompressed property](#)  
[ReadOnly property](#)  
[Red property](#)  
[Relative property](#)  
[RelativeIndent property](#)  
[RelativePageNum property](#)  
[RelativeType property](#)  
[RelativeXDistance property](#)  
[RelativeYDistance property](#)  
[ReleaseNumber property](#)  
[RenderedPageNumber property](#)  
[ReplaceAttributes property](#)  
[ReplaceExactCase property](#)  
[ReplaceFont property](#)  
[ReplaceLanguage property](#)  
[Replacements property](#)  
[ReplaceString property](#)  
[ReplaceStyleName property](#)  
[ReplaceWithProperties property](#)  
[RequestRemarkOnClose property](#)  
[RequireStartupScripts property](#)  
[ResetOnEachPage property](#)  
[ResetWhen property](#)  
[Rest property](#)  
[RestartStyleName property](#)  
[Result property](#)  
[ResumePausedMacro property](#)  
[RetainNameOfImportedFile property](#)  
[ReviewVersions property](#)  
[RevisionDisplay property](#)  
[RevisionMark property](#)  
[RevisionMarkMode property](#)  
[RevisionType property](#)  
[RevMarkCharacter property](#)  
[RevMarkPosition property](#)

[RevMarkType property](#)  
[Right property](#)  
[RightAlign property](#)  
[RightBorder property](#)  
[RightExternalMargin property](#)  
[RightIntArea property](#)  
[RightMouseMenus property](#)  
[RightMousePropId property](#)  
[RightMousePropText property](#)  
[RightPage property](#)  
[RotationAngle property](#)  
[RowLayouts property](#)  
[Rows property](#)  
[RubyLayouts property](#)  
[RunMacroOnDocEvents property](#)  
[RunMacroOnLoad property](#)  
[RunOnCloseDoc property](#)  
[RunOnNewDoc property](#)  
[RunOnOpenDoc property](#)

## S

[Saved property](#)  
[SaveSnapShot property](#)  
[ScaleHeight property](#)  
[ScaleMode property](#)  
[ScalePercentage property](#)  
[ScaleWidth property](#)  
[ScreenPositionX property](#)  
[ScreenPositionY property](#)  
[Script property](#)  
[ScriptName property](#)  
[SearchAttributes property](#)  
[SearchLanguage property](#)  
[Section property](#)  
[SectionName property](#)  
[Sections property](#)  
[SectionTabs property](#)  
[SelectedPages property](#)  
[SelectionBorderColor1 property](#)  
[SelectionBorderColor2 property](#)  
[SelectionBorderColor3 property](#)  
[SelectionHidden property](#)  
[SelectionType property](#)  
[SelectTab property](#)  
[SelectType property](#)  
[ServerFormat property](#)  
[SetContextOfBar property](#)  
[SetTabsDialog property](#)  
[Shadow property](#)  
[Shape property](#)  
[ShowBubbleHelp property](#)  
[ShowDivisionTabs property](#)  
[ShowFileNew property](#)  
[ShowGraphicPreview property](#)  
[ShowHiddenText property](#)  
[ShowInContext property](#)  
[ShowMailDisabled property](#)

[ShowNoWelcomeBox property](#)  
[ShowStatistics property](#)  
[ShowTabs property](#)  
[SilverBullet property](#)  
[SilverBullets property](#)  
[SingleCellSelected property](#)  
[Size property](#)  
[SizeStyleName property](#)  
[SizingUnitName property](#)  
[SizingUnits property](#)  
[Skipped property](#)  
[SkipWordMode property](#)  
[SmallCaps property](#)  
[SmallFileFormat property](#)  
[SmartCorrect property](#)  
[SmartCorrects property](#)  
[SmartFill property](#)  
[SmartLevel property](#)  
[SnapshotOffset property](#)  
[SnapshotPath property](#)  
[SnapshotSaveOptions property](#)  
[SnapshotSize property](#)  
[SortLevel1 property](#)  
[SortLevel2 property](#)  
[SortLevel3 property](#)  
[SortNumbers property](#)  
[SortOptions property](#)  
[SortOrder property](#)  
[SortType property](#)  
[SortWord property](#)  
[SortWordOption property](#)  
[SpaceAbove property](#)  
[SpaceBelow property](#)  
[SpacesBetweenSentences property](#)  
[Spacing property](#)  
[SpacingStyleName property](#)  
[SpacingUnitName property](#)  
[SpacingUnits property](#)  
[Span property](#)  
[SpellCheckInitialCaps property](#)  
[SpellCheckRepeatedWords property](#)  
[SpellCheckUserDictAlternatives property](#)  
[SpellCheckWordsWithNums property](#)  
[SpellColor property](#)  
[SpellFocusedColor property](#)  
[SpellString property](#)  
[SplitPercentage property](#)  
[Start property](#)  
[StartColumns property](#)  
[StartingColOfSelection property](#)  
[StartingColStringOfSelection property](#)  
[StartingNumber property](#)  
[StartingRowOfSelection property](#)  
[StartRow property](#)  
[StateID property](#)  
[StatusBar property](#)  
[StatusBarButtons property](#)  
[StatusBarVisible property](#)

[StatusSpellReplaceAll property](#)  
[StrikeThrough property](#)  
[Style property](#)  
[StyleExceptions property](#)  
[StyleName property](#)  
[StylePath property](#)  
[StylePaths property](#)  
[StyleSheetFullPath property](#)  
[StyleSheetName property](#)  
[StyleSheetPath property](#)  
[Subscript property](#)  
[Suffix property](#)  
[Suggestions property](#)  
[Superscript property](#)  
[SuperTableContainer property](#)  
[SuperTableLayouts property](#)  
[SuperTables property](#)  
[SuppressHeaders property](#)

## T

[TabExits property](#)  
[Table property](#)  
[TableContainer property](#)  
[TableExports property](#)  
[TableFill property](#)  
[TableHeadingLayouts property](#)  
[TableHeadings property](#)  
[TableImports property](#)  
[TableLayouts property](#)  
[TableLine property](#)  
[TableMarkers property](#)  
[TableOnlyContainer property](#)  
[Tables property](#)  
[TableStyleName property](#)  
[TableStyles property](#)  
[TabOrder property](#)  
[TabRack property](#)  
[TabRelativeTo property](#)  
[TabSpacing property](#)  
[TabSymbolChar property](#)  
[TabType property](#)  
[TabWidth property](#)  
[TempFindAndReplace property](#)  
[TempFoundry property](#)  
[Text property](#)  
[TextandTableExports property](#)  
[TextandTableImports property](#)  
[TextAttributes property](#)  
[TextMarkers property](#)  
[TextOrient property](#)  
[TextOrientation property](#)  
[Texts property](#)  
[TextStyleName property](#)  
[TextStyles property](#)  
[TextTightness property](#)  
[TextViewAttributes property](#)  
[ThesMaxNumSynonymsReturned property](#)

[Tile property](#)  
[TileWindow property](#)  
[Time property](#)  
[TimedSaveFileExtension property](#)  
[Title property](#)  
[TitleBarDocNumber property](#)  
[TitleBarVisible property](#)  
[TOCRange property](#)  
[TOCSource property](#)  
[Top property](#)  
[TopBorder property](#)  
[TopExternalMargin property](#)  
[TopIntArea property](#)  
[TopLeftCellRowId property](#)  
[TotalEditingTime property](#)  
[TrailingText property](#)  
[Type property](#)  
[TypeAbove property](#)  
[TypeAboveLine property](#)  
[TypeBelow property](#)  
[TypeRight property](#)  
[Typos property](#)

## U

[Underline property](#)  
[UndoEnable property](#)  
[UndoLevels property](#)  
[Units property](#)  
[UpdateFields property](#)  
[UpdateOnLoad property](#)  
[UpdateOnLoadImmediate property](#)  
[UpperCase property](#)  
[UseConsistentSpaceBetweenSentences property](#)  
[UseContents property](#)  
[UseDefaultPrinter property](#)  
[UsedFirstPageHeight property](#)  
[UseEncrypt property](#)  
[UseExcludeRect property](#)  
[UseFindStyle property](#)  
[UseFooter property](#)  
[UseGreeting property](#)  
[UseHeader property](#)  
[UseNextStyle property](#)  
[UsePrinterSettings property](#)  
[UserClassNameApp property](#)  
[UserClassNameFull property](#)  
[UserClassNameShort property](#)  
[UserDefinedFilter property](#)  
[UserDictFiles property](#)  
[UserDictionaryFiles property](#)  
[UserDictionaryPath property](#)  
[UserDictionaryPaths property](#)  
[UserDictStates property](#)  
[UseRelative property](#)  
[UseReplaceStyle property](#)  
[UserInitials property](#)  
[UserInterfacePrefs property](#)



[UserLinkName property](#)  
[UserName property](#)  
[UseSeparatorLine property](#)  
[UsesHelp property](#)  
[UsesPalette property](#)  
[UseSuperscriptReferenceNum property](#)  
[UseTwoSpacesBetweenSentences property](#)  
[UseWhen property](#)

## V

[Valid property](#)  
[VersionID property](#)  
[VersionManager property](#)  
[VersionName property](#)  
[VersionRemarks property](#)  
[Versions property](#)  
[VertAlign property](#)  
[VerticalSplitWindow property](#)  
[VertRuler property](#)  
[VertScrollBarVisible property](#)  
[ViewLevel property](#)  
[ViewType property](#)  
[Visible property](#)

## W

[WasDeletedInRevMarkMode property](#)  
[WasInsertedInRevMarkMode property](#)  
[WasPasted property](#)  
[WaterMarkName property](#)  
[Weight property](#)  
[Where property](#)  
[WidowOrphan property](#)  
[Width property](#)  
[WidthAbove property](#)  
[WidthBelow property](#)  
[WidthInLongtwips property](#)  
[WidthOfColumn property](#)  
[WindowId property](#)  
[WindowsName property](#)  
[WinViewPrefs property](#)  
[WithComments property](#)  
[WordDoubleUnderline property](#)  
[WordUnderline property](#)  
[WorkingType property](#)  
[WPDataSets property](#)  
[WrapType property](#)

## X

[XOffset property](#)  
[XPosition property](#)

## Y

[YOffset property](#)  
[YPosition property](#)

## **Z**

Zero property

ZipCode property

**Developing SmartSuite Applications**

*Developing SmartSuite Applications Using LotusScript* is available in the SmartSuite CD package as an online book. To install *Developing SmartSuite Applications Using LotusScript*, see the SmartSuite installation instructions.

To order a printed version of *Developing SmartSuite Applications Using LotusScript* and other LotusScript user assistance in the SmartSuite Developer's Kit, complete the [order form](#) and return it to Lotus.

**Order Form for the SmartSuite 97 Application Developer's Kit**

To order the *SmartSuite Application Developer's Kit*, complete the following form and mail or fax it to Lotus. You will receive the *SmartSuite Application Developer's Kit* within 21 days.

**Ordering by US Mail**

Mail the completed form to:

Lotus Development Corporation  
55 Cambridge Parkway  
Cambridge, MA 02242

**Ordering by FAX**

Fax the completed form to:

(617) 537-8500

**Order Form**

Name: \_\_\_\_\_

Title: \_\_\_\_\_

Mailing address: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

E-mail address: \_\_\_\_\_

Product Registration Number: \_\_\_\_\_

**LotusScript Documentation as Online Books**

If you purchased the CD-ROM version of SmartSuite 97, you can use SmartSuite Install to install the following online Books about LotusScript:

- *Getting the Most Out of LotusScript in SmartSuite 97*
- *Developing SmartSuite Applications Using LotusScript*
- *LotusScript Language Reference*
- *LotusScript Programmer's Guide*

For more information about installing Online Books, see your SmartSuite 97 installation documentation.

### **LotusScript Documentation on the Web**

You can view updated versions of LotusScript documentation, or download updated sample applications or Help files from the LotusScript home page.

If you configured Windows to launch your Web browser automatically when you click a URL on your desktop, you can click the following button to go to the LotusScript home page.

 [Click to access the LotusScript home page](#)

If you did not configure Windows to launch your Web browser automatically, enter the following URL in the location field in your browser and press ENTER:

`http://webedit/applicat/lsmain.htm`

## Overview: Designing SmartSuite Applications

LotusScript provides a variety of tools and services to support you in developing applications for SmartSuite. Getting productive in a new programming environment often involves understanding how all the pieces work together -- the tools, the language conventions, the object dependencies, and so on. Understanding how to approach the problem and where to enter your script code is half the challenge in learning.

### Choosing a place to begin

Lotus Notes, 1-2-3, Approach, Freelance Graphics, and Word Pro all use the same underlying LotusScript language. Each product implements LotusObjects on top of the LotusScript language. To determine what product best supports the goals for your script application, consider using each of the SmartSuite products and reviewing its features. Read *Developing SmartSuite Applications Using LotusScript* for overviews of what each product can bring to your programming effort. Implement a few simple procedures in each product to get a feel for its features and objects. In the long run, you'll be better able to determine what product provides strengths where you need them most, and how you can develop cross-product applications that take advantage of the strengths of each product.

### Working the basics

LotusScript applications share the following common features.

- You need a Lotus product to run script applications.
- You need a Lotus product to store scripts in a product document such as a 1-2-3 workbook or Word Pro document.
- You need to run the Lotus Integrated Development Environment (IDE) to edit and debug scripts stored in a product document.
- You need to open an IDE window for each product document containing scripts that you want to modify.

To write a basic script application, therefore, you must run a Lotus product and load a document in that product. You can then write scripts for the product objects that you create in your product.

### Writing scripts in the Integrated Development Environment (IDE)

Your primary tool for developing script applications is the Lotus Integrated Development Environment (IDE). Beyond providing the basic tools such as an editor, a debugger, a browser, and a dialog editor, the IDE provides a high degree of integration with each Lotus product. It is easy to move between tasks that you perform in a product and those that you perform in the IDE.

### Writing global scripts

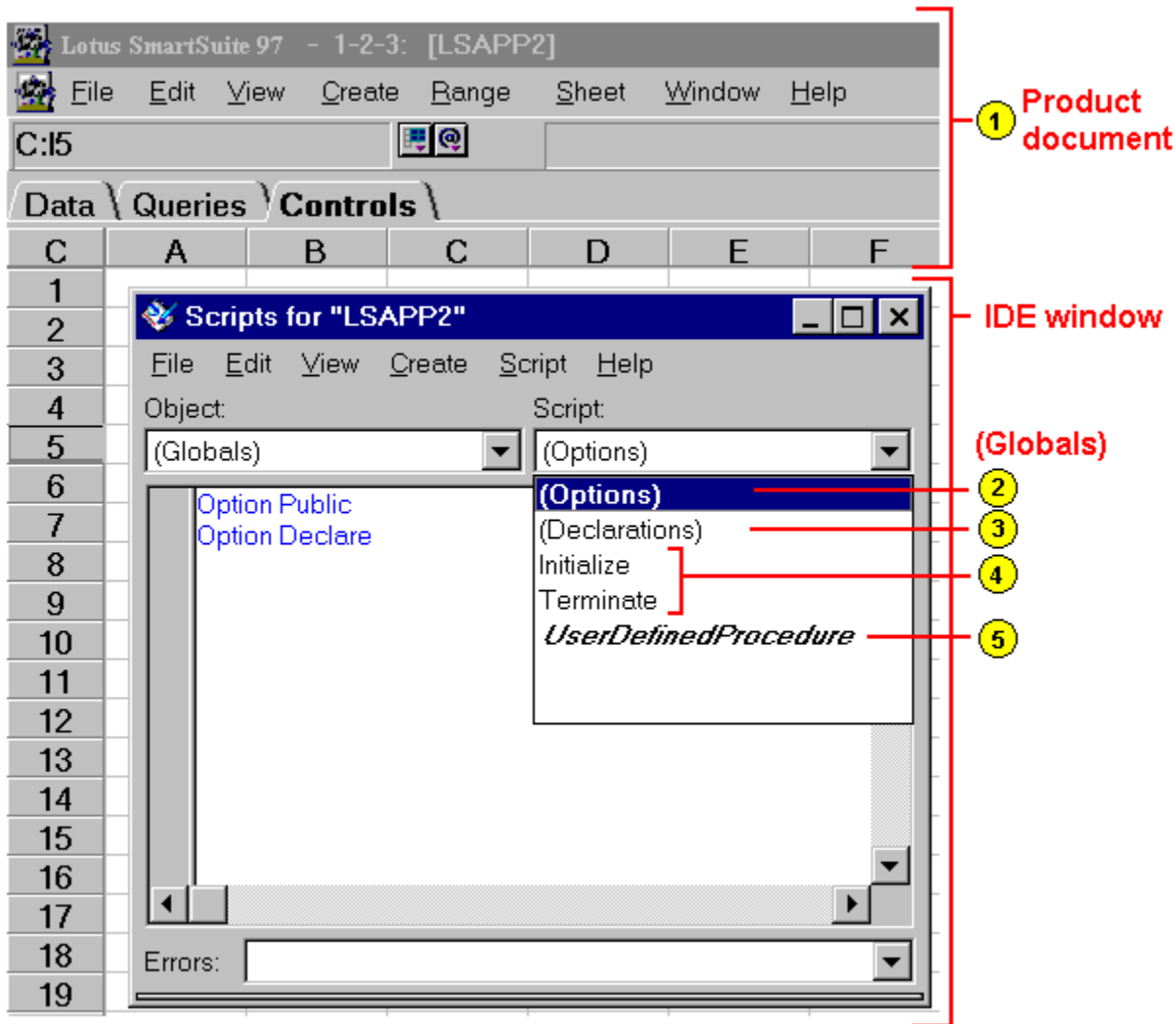
Global scripts make declarations, options, and procedures available to all scripts in your document. For example, to write global scripts for a 1-2-3 document named LSAPP2.123, you must first run 1-2-3, load the document LSAPP2.123, and then open an IDE window for that document. Choose Edit - Scripts & Macros - Show Script Editor in the 1-2-3 menu to activate an IDE window for your current document.

The IDE lists objects that you can script in the Object list and scripts for each of those objects in the Script list. You can add statements to predefined scripts in (Globals), such as (Options), (Declarations), Initialize, or Terminate, or you can create your own named procedures. You do not need to modify predefined scripts to write a basic script application.

The following illustration shows how to select a particular script for (Globals).

Click any item in the following list to learn more about it.

- |          |  |          |   |
|----------|--|----------|---|
| <b>1</b> | <a href="#">Product document</a>       | <b>4</b> | <a href="#">Initialize and Terminate subs</a> |
| <b>2</b> | <a href="#">(Options) scripts</a>      | <b>5</b> | <a href="#">User-defined procedures</a>       |
| <b>3</b> | <a href="#">(Declarations) scripts</a> |          |   |



### Writing scripts for product objects

You can also write scripts for product objects in your document. As with (Globals), you can add statements in the predefined scripts for an object, or create new procedures for that object. Unlike scripts that you write in (Globals), the declarations, options statements, and procedures that you write for a product object are not generally available to scripts attached to a different product object.

The predefined scripts for product objects include object event procedures. Script statements in an object event procedure are executed when an object, such as a button, receives a particular event in your product, such as being clicked, double-clicked, or moved. For example, if you add a button named Button 5 to the 1-2-3 document LSAPP2.123, and you want it to run some script when you click it, you must add script statements to the Click procedure for Button 5. To select this event procedure, choose the Button 5 object in the IDE Object list and choose Click in the Script list.

The following illustration shows how to select a predefined or user-defined script for a 1-2-3 product object named Button 5.

Click any item in the following list to learn more about it.

- |  |  |
|--|--|
| <b>1</b> <a href="#">User-defined procedures</a> | <b>4</b> <a href="#">Event procedures</a>              |
| <b>2</b> <a href="#">(Options) scripts</a>       | <b>5</b> <a href="#">Initialize and Terminate subs</a> |
| <b>3</b> <a href="#">(Declarations) scripts</a>  |  |



The image shows a screenshot of Lotus SmartSuite 97. The main window displays a spreadsheet with a 'Button 5' object in cell A1. A dialog box titled 'Scripts for "LSAPP2"' is open, showing the script for the selected object. The dialog box has a menu bar (File, Edit, View, Create, Script, Help) and two dropdown menus: 'Object' (set to 'Button 5') and 'Script' (set to 'UserDefinedSub'). The script editor shows the following code:

```

Sub UserDefinedSub
  Dim AppName As String
  AppName = "WeeklyBud
  Call AppSetup1(AppNam
End Sub

```

On the right side of the dialog box, there is a list of event handlers: (Options), (Declarations), Click, Deselected, Initialize, Methodinvoked, and Namechange. Red lines and yellow circles with numbers 1-5 point to the following elements:

- 1: Points to the 'Script' dropdown menu.
- 2: Points to the '(Options)' section.
- 3: Points to the '(Declarations)' section.
- 4: Points to the 'Click' event handler.
- 5: Points to the 'Initialize' event handler.

Red text labels on the right side of the image identify 'Button 5' as the 'Product object' and the script editor area as 'Object scripts'.

### Working with external script files

In many cases, the one-application-per-document approach is sufficient for working with objects and data in isolated documents. To develop more sophisticated applications that reuse important scripts or use multiple products, consider using the following types of external script files:

[LotusScript Script \(LSS\) files](#)

[LotusScript Object \(LSO\) files](#)

[LotusScript Extension \(LSX\) files](#)

[OLE Custom Control \(OCX\) files](#)

[Dynamic-link Library \(DLL\) files](#)

## Dynamic-link Library (DLL) files

If you develop useful functions in C and compiled them in a Dynamic-link Library (DLL), you can call them from your LotusScript application. For example, the following procedure declares and calls a LotusScript function named SendDLL, corresponding to a C function named \_SendExportedRoutine in the DLL file named MYEXPORTS.DLL.

```
Declare Function SendDLL Lib _  
    "C:\LOTUS\ADDINS\MYEXPORTS.DLL" _  
    Alias "_SendExportedRoutine" (i1 As Long, i2 As Long)  
SendDLL(5, 10)
```

For more information on using Dynamic-link Libraries, see *LotusScript Language Reference*.

**(Declarations) scripts in (Globals)**

The (Declarations) script is designed to contain the following statements:

- Dim statements for variables that you want to be available to all scripts in your document
- Public, Private, Type, Class, and Declare Lib statements (external C calls)
- Const statements for those constants that you want to be available to all scripts in your document and are not needed for Use or UseLSX statements in (Options)

By default, the (Declarations) script is initially empty.

If you enter Type, Class, or Declare Lib statements in any other script in (Globals), the IDE moves them to (Declarations) automatically. If you enter Dim, Public, Private, or Const statements outside the scope of a procedure in another script, the IDE moves them to (Declarations) automatically. Const statements in (Options) are the exception to this rule.

## **Initialize and Terminate subs in (Globals)**

### **Initialize script**

Use the Initialize sub in (Globals) to initialize variables that you declared in (Declarations). The Initialize sub executes before any of these variables are accessed and before any other scripts in (Globals) are executed. By default, the Initialize script is empty.

### **Terminate script**

Use the Terminate sub in (Globals) to clean up variables that you declared in (Declarations) when you close your document, or when you modify a script and execute it again. For example, you might use an Open statement to open a file containing data in Initialize, and use a Close statement in Terminate to close it. By default, the Terminate script is empty.

### **(Options) scripts in (Globals)**

The (Options) script in (Globals) is designed to contain these the following statements:

- Option statements  
**Note** (Options) contains the statement, Option Public, by default. This makes Const, Dim, Type, Class, Sub, Function, and Property statements public by default. You can use the Public form of these statements to make them public explicitly, or the Private form to make them unavailable to other scripts outside (Globals).
- Deftype statements
- Use and UseLSX statements
- Const statements needed for Use and UseLSX statements

If you enter any of these statements, except for Const, in any other script in (Globals), the IDE automatically moves them to (Options).

Option and Deftype statements that you enter in (Options) apply only to scripts for the current object. To make certain that an option is applied consistently throughout your document, enter the appropriate statement in the (Options) script for every object for which you are writing scripts.

### **User-defined procedures in (Globals)**

While you are working in (Globals), you can add procedures to make them available throughout your document. There are three ways to add procedures to (Globals) in the IDE:

- *Using the IDE menu:* Choose Create - New Sub or Create - New Function in the IDE menu to create new subs and functions in (Globals). The IDE automatically adds the name of the new procedure to the Script list.
- *Entering statements:* Enter a Sub, Function, or Property statement anywhere in (Globals) except within a class. The IDE automatically adds the name of the new procedure to the Script list for (Globals).
- *Importing procedures from a file:* Use File - Import Script in the IDE menu to import scripts when you are working in (Globals). These imported scripts will be available to all scripts in your document. The IDE automatically adds the name of any new procedures contained in the imported script to the Script list.

## **LotusScript User Assistance for SmartSuite 97**

To help you learn how to develop LotusScript applications for SmartSuite 97, Lotus provides a complete library of user assistance.

### ***Getting the Most Out of LotusScript in SmartSuite 97***

This publication explains how SmartSuite 97 products use the LotusScript programming language and how your business can take advantage of LotusScript in developing applications for SmartSuite.

*Getting the Most Out of LotusScript in SmartSuite 97* is available in hard copy, Adobe Acrobat, or HTML formats in your SmartSuite 97 package, in the [SmartSuite Application Developer's Kit](#), or on the [Worldwide Web](#).

### ***Developing SmartSuite Applications Using LotusScript***

This publication provides comprehensive information on key concepts and techniques for developing LotusScript applications. *Developing SmartSuite Applications Using LotusScript* focuses on programming tools, cross-application programming, Lotus Notes integration, and product-specific application development.

*Developing SmartSuite Applications Using LotusScript* is available in hard copy, Adobe Acrobat, or HTML formats in your SmartSuite 97 package, in the [SmartSuite Application Developer's Kit](#), or on the [Worldwide Web](#).

### ***LotusScript Language Reference***

This publication provides a comprehensive summary of conventions and basic commands for the LotusScript language. *LotusScript Language Reference* provides the foundation for programming any product that supports the LotusScript programming language.

*LotusScript Language Reference* is available in hard copy, Adobe Acrobat, Help, or HTML formats in your SmartSuite 97 package, in the [SmartSuite Application Developer's Kit](#), or on the [Worldwide Web](#).

### ***LotusScript Programmer's Guide***

This publication is a general introduction to LotusScript that describes basic building blocks in the language and explains how to use them to create powerful applications.

*LotusScript Programmer's Guide* is available in hard copy, Adobe Acrobat, or HTML formats in your SmartSuite 97 package, in the [SmartSuite Application Developer's Kit](#), or on the [Worldwide Web](#).

### **Class Reference Help and Frequently-asked Questions**

Each product provides comprehensive Help on product classes, frequently-asked questions about programming, and code examples. All this is delivered in an innovative Help system designed to enhance your work as a programmer.

Class reference Help and frequently-asked questions are available in Help or HTML formats in your SmartSuite 97 package, in the [SmartSuite Application Developer's Kit](#), or on the [Worldwide Web](#).

### **Example code and sample applications**

Most products also provide working code to illustrate important programming techniques. You can reuse and modify this code as you develop your own applications.

Example code is available in the SmartSuite CD-ROM package, in the [SmartSuite Application Developer's Kit](#), and on the [Worldwide Web](#).

## **LotusScript Object (LSO) files**

LotusScript Object (LSO) files contain public definitions that you can use in your script applications. If you develop a library of commonly-used declarations or procedures that you want to reuse across multiple script applications, you can collect them in a product document, and use the File - Export Globals as LSO menu command to create a compiled LotusScript Object file. If this file were named WKREPORT.LSO, you would make these public definitions available to your script application by entering the following statement in the appropriate (Options) script:

```
Use "C:\LOTUS\ADDINS\WKREPORT.LSO"
```

For more information on using LotusScript Object files, see *LotusScript Language Reference*.



## **LotusScript Script (LSS) files**

LotusScript Script (LSS) files are text files that contain LotusScript statements. You can create LSS files in any text editor. Use the %Include directive anywhere in a script to reference the contents of an LSS file. For example, to include the contents of a LotusScript Script file named STDSETUP.LSS in your application, enter the following statement:

```
%Include "C:\MYSCRIPTS\STDSETUP.LSS"
```

By default, LotusScript assumes that the LotusScript Script files referenced have an LSS file extension. You can actually use any extension for your text file or no extension at all.

For more information on using LotusScript Script files, see *LotusScript Language Reference*.

## LotusScript Extension (LSX) files

LotusScript Extension (LSX) files are Dynamic-link Libraries (DLLs) that contain public class definitions. LSX files are developed using the Lotus LSX Toolkit. To obtain a version of the LSX Toolkit for your operating system, connect to the Lotus home page on the WorldWide Web. Lotus ships LSX files for Lotus Notes and Approach; other LSX files are being developed for SmartSuite products by Lotus and by third-party developers. These extension files expand the range of classes that you can use in your LotusScript applications.

**Tip** You can enter a UseLSX statement in any script; the IDE automatically moves it to (Options).

## Loading and using class definitions in LSX files

There are two ways to load and use the public class definitions in an LSX file.

- If the LSX file that you want to load is not registered in the Windows Registry, you must refer to the LSX file directly in your UseLSX statement.

```
UseLSX "C:\MYSRIPTS\LSX4DB2.DLL"
```

- If an LSX is registered and you want to reference a class definition directly, you can enter the name of the class definition.

```
UseLSX "ObjectName"
```

In this example, LotusScript searches all entries under "LotusScriptExtensions" in the Windows Registry for the specified class definition, and loads that definition.

**Note** If the LSX file you want to load is registered in the Windows Registry, you can reference its Registry name and have Windows provide the appropriate DLL name and file path. SmartSuite 97 registers an LSX file that contains Notes public class definitions. To use these Notes class definitions in your cross-product script applications, enter the following statement:

```
UseLSX "*Notes"
```

## Viewing class definitions

Once you run a script containing a UseLSX statement and loaded an LSX file, you can browse its class definitions in the IDE Browser panel.

For more information on using LotusScript Extension files, see *LotusScript Language Reference*.

**(Declaration) scripts in object scripts**

The (Declarations) script for an object is designed to contain the following statements:

- Dim statements for variables that you want to be available to all scripts for the current object
- Const statements for those constants that you want to be available to all scripts for the current object and that are not needed for Use or UseLSX statements in (Options)

By default, the (Declarations) script is initially empty.

**Event procedures in object scripts**

If you are writing a script for an object, the Script list displays default event procedures for the selected object. In the IDE, you cannot create new event procedures for an existing product object because valid events for that object are defined by the product.

## **Initialize and Terminate subs in object scripts**

### **Initialize sub**

Use the Initialize sub to set up variables declared in the object's (Declarations) script. The Initialize sub for an object executes before any of its event procedures. By default, the Initialize script is empty.

**Note** Scripts for controls created in the Lotus Dialog Editor do not have Initialize subs.

### **Terminate sub**

Use the Terminate sub to clean up variables that you declared in the object's (Declarations) script. By default, the Terminate script is empty.

**Note** Scripts for controls created in the Lotus Dialog Editor do not have Terminate subs.

**(Options) scripts in object scripts**

The (Options) script for an object is designed to contain these the following statements:

- Option statements
- *Deftype* statements
- Use and UseLSX statements
- Const statements needed for Use and UseLSX statements

### **User-defined procedures in object scripts**

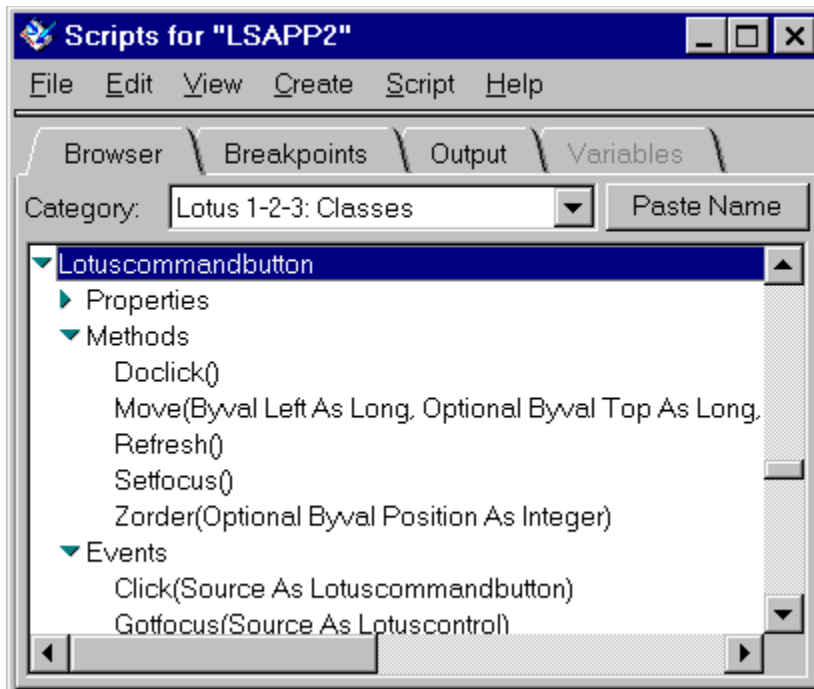
You can create other named subs, functions, and properties for objects, in addition to the predefined scripts or event procedures. Because these procedures are not in (Globals), they can be called only from other scripts for the object.

There are three ways to create object scripts in the IDE:

- *Using the IDE menu:* Use Create - New Sub and Create - New Function to create new subs and functions for an object. The IDE automatically adds the name of the new procedure to the Script list for that object.
- *Entering statements:* Enter a Sub, Function, or Property statement anywhere in a script for the current object. The IDE automatically adds the name of the new procedure to the Script list for that object.
- *Importing procedures from a file:* Use File - Import Script when you are working with object scripts to import scripts for that object. The IDE automatically adds the name of any new procedures contained in the imported script to the Script list.

## OLE Custom Control (OCX) files

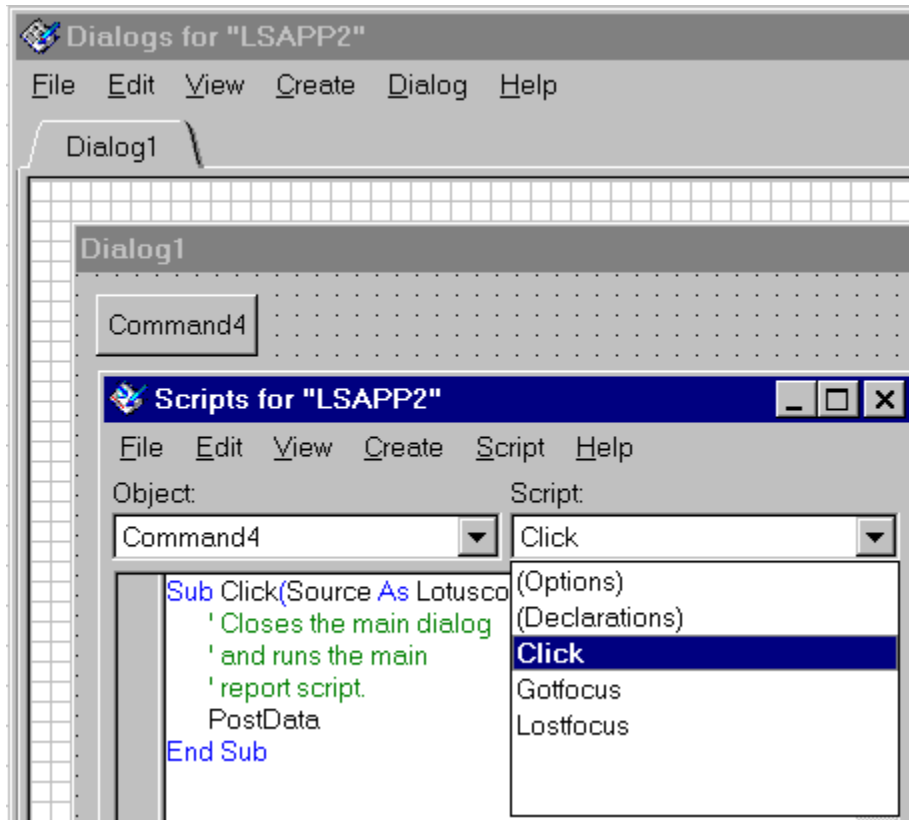
OLE Custom Controls extend the number of objects that you can script in Lotus products. For example, the Lotus dialog controls listed under product classes in the IDE Browser panel are OCX controls that you can add to the Lotus Dialog Editor.



Once you add an OCX control to your product, you can script its properties, methods, and events in the IDE Script Editor.

The following illustration shows how the properties, methods, and events of an Lotus CommandButton OCX named Command4 are available to you in the IDE.





**Tip** You can add OCX controls registered on your system to the Lotus Dialog Editor Toolbox by choosing File - Toolbox Setup in the Lotus Dialog Editor menu.

**Product Document**

To edit scripts in the IDE or to execute them in one or more products, you must create or use a document in your product that contains the scripts. Lotus products supporting LotusScript use the following document extensions:

<i>Lotus Product</i>	<i>Document extension(s)</i>
1-2-3	123
Approach	APR
Freelance Graphics	SMC
Notes	NSF
Word Pro	LWP

## Using LotusScript Examples

Code examples provide working models for the scripts that you write. Whether the example is listed in a Help example or available as a product document on disk, you can copy statements or entire scripts from the examples and use them in your own script applications.

There are three types of LotusScript examples, each designed to illustrate a different aspect of the LotusScript language or the classes available for each SmartSuite product.

### Examples in reference Help

Most examples appear in reference Help for the LotusScript language and for product classes. These brief examples focus on individual elements in the language or members of a product class. They illustrate how to use correct syntax for a working example, how to enter appropriate values for parameters, and how dependencies between elements operate.

**Note** Although you can copy examples from reference Help and paste them into your scripts, they are not designed primarily to be self-contained. Sometimes there are dependencies between a piece of example code and the larger sample application from which it is derived.

### Examples in Frequently-asked Questions (FAQs) Help

Frequently-asked questions (FAQs) illustrate how to complete common programming tasks using LotusScript. Examples in FAQs not only illustrate how individual statements work, but they also illustrate how these statements form a complete application or procedure. Most examples in FAQs are designed to be self-sufficient; you can copy one or more procedures from Help, paste them into your own scripts in the Script Editor, and execute them.

**Note** When there are dependencies in an example that would require you to modify the example to make it run, these dependencies are documented in the Help topic or at the beginning of the first script in the example.

### Sample applications

The *Developing SmartSuite Applications Using LotusScript* book includes numerous sample applications for SmartSuite and for individual products. These examples are designed to illustrate more sophisticated tasks for an individual product or tasks that utilize more than one product. They illustrate how to develop script applications that take advantage of embedded OLE objects, OLE automation, Notes, Visual Basic, the Worldwide Web, and custom Dynamic-link libraries (DLLs). Lotus develops new sample applications for SmartSuite on an ongoing basis; these new samples and updated versions of the ones in *Developing SmartSuite Applications Using LotusScript* are available on the [Worldwide Web](#).

To copy scripts from these sample applications and paste them into your own script applications, you must first open the sample application document and then display its scripts by opening the IDE window for that document.

**Note** All sample applications in *Developing SmartSuite Applications Using LotusScript* are designed to run without modification.

## Using LotusScript Help

The design for LotusScript Help supports three of the most frequent activities that you perform as a programmer:

- Searching for objects and elements to use in your scripts
- Writing scripts
- Debugging scripts

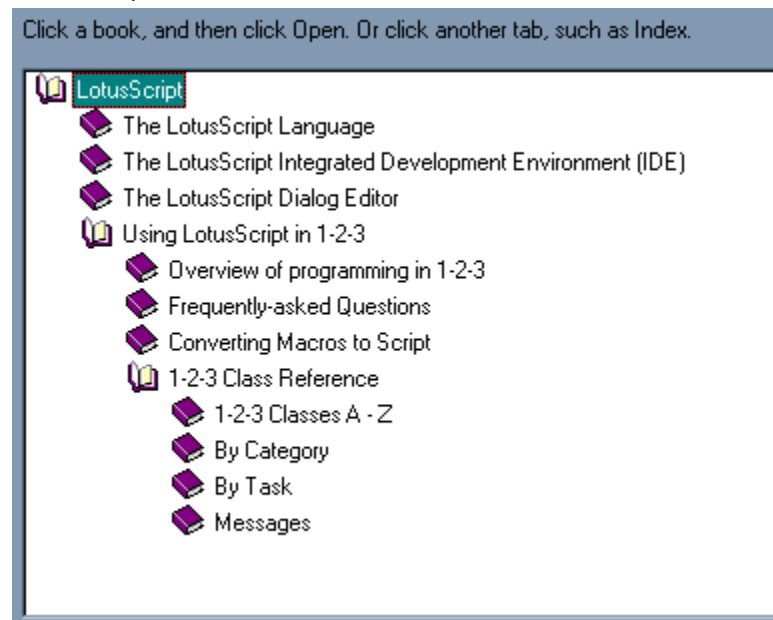
LotusScript Help uses different types of windows to display different types of information, so it is important to know what each type of window contains and how to navigate between them.

## Using Help to search for objects and elements

There are areas in Help designed to help you search for objects and language elements to use in your scripts:

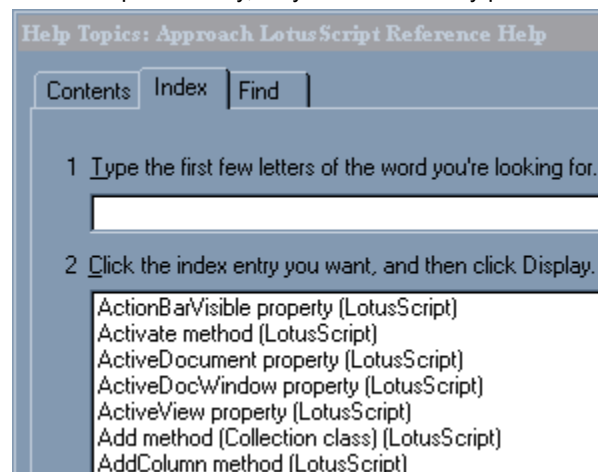
### LotusScript Help Contents

You can use Contents in Help to examine the overall structure of Help and to browse for Help topics relevant to your current script.



### LotusScript Index

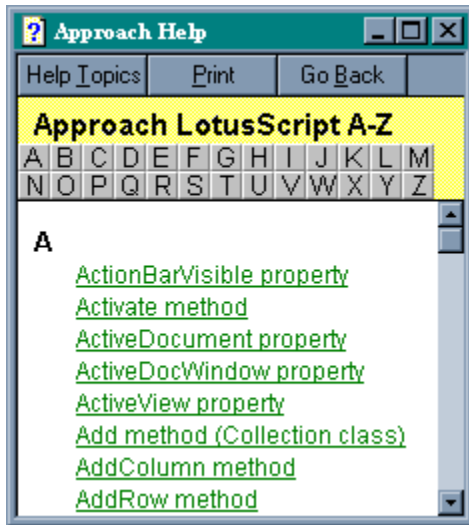
Indexes are one of the most popular ways that programmers search for information. Topics in LotusScript Help are indexed alphabetically, so you can enter key phrases or keywords and navigate to the corresponding Help topics.



### LotusScript A - Z lists

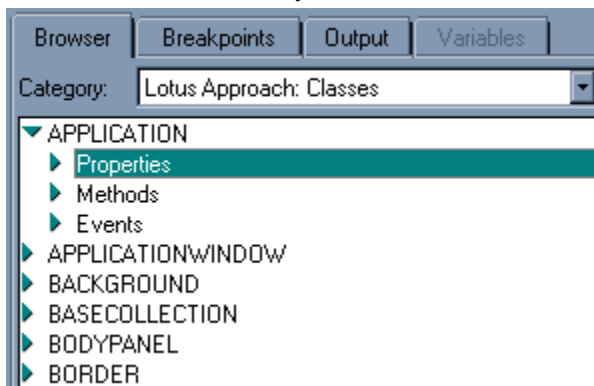
LotusScript Help for each product provides A - Z lists of its classes, properties, methods, and events, including a

comprehensive list of all the elements in the product.



### IDE Browser Help

The Browser panel in the Integrated Development Environment (IDE) displays lists of LotusScript language elements and classes for products. You can expand and collapse entries in the Browser to view the associated properties, methods, and events for objects.



Highlight an element in the Browser panel and press F1 (HELP) to get context-sensitive Help on that element.

### Using Help to write scripts

Help focuses on objects. As you are writing scripts, you explore the relationships between product classes and the behaviors of objects in that product.

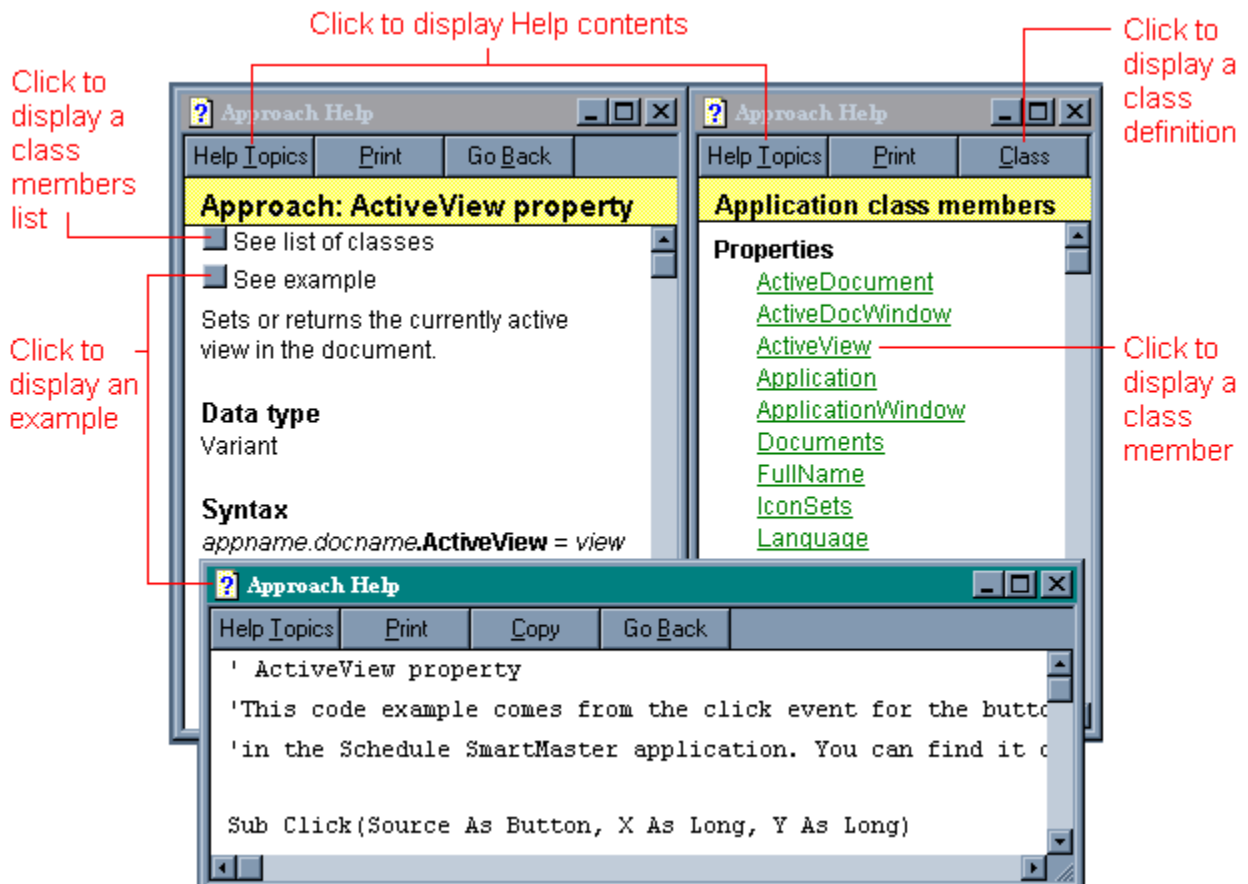
### Types of Help windows

To support this exploration, Help separates information about classes into four types of windows:

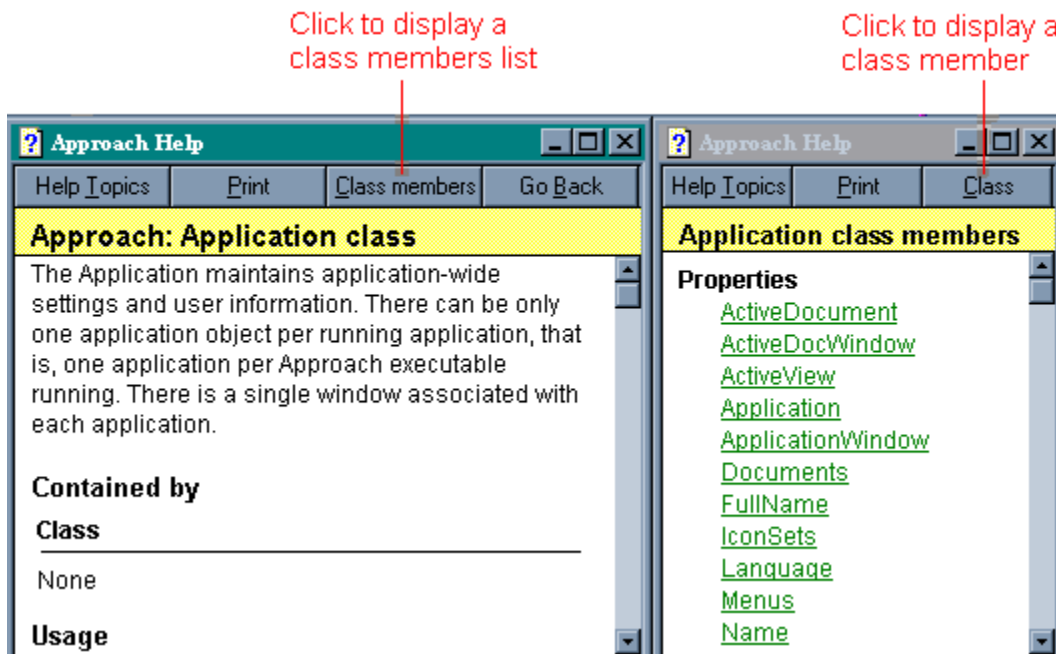
- Class definition windows define what a class does in a product and how it works in the product's containment hierarchy. The class definition topic for the 1-2-3 Range object describes what ranges do in 1-2-3, how they are contained by larger objects, and how they contain smaller objects.
- Class member list windows list all the properties, methods, and events that are members of a particular class.
- Class member windows focus on particular properties, methods, or events.
- Example windows contain one or more scripts for a particular property, method, or event. You can copy and paste script statements from these example windows into the IDE Script Editor.

### Displaying Help windows

To display different types of LotusScript Help windows, use buttons in Help topics and in the Help window that are labeled by the type of Help window. The following illustration shows how to use buttons to display class member, class member list, and example windows in Help.



The following illustration shows how to display class definition and class member list windows in Help.



### Help for editing and debugging scripts

You can also get context-sensitive Help about keywords and messages when you are editing or debugging your

scripts in the IDE.

### **Context-sensitive Help in the Script Editor and Script Debugger**

If you need help on a keyword while you are writing or debugging a script in the Script Editor and Script Debugger, place the insertion point on the keyword and press F1 (HELP) to get context-sensitive Help on that keyword.

### **Context-sensitive Help on messages**

You can also get context-sensitive Help on two types of messages in the IDE. In the Script Editor, you can get context-sensitive Help on syntax errors. Navigate to the statement that caused the error and press F1 (HELP). When you are debugging your scripts and the IDE reports a run-time error, press F1 (HELP) to display information about that error and suggestions about fixing it.

