

'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

'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 IconMgr As IconBarManager

Set IconMgr = .ApplicationWindow.IconBarManager

' Select the icon to add.

' For this example to work, the icon and the script must already be linked

IconMgr.SelectCustomIcon "c:\lotus\wordpro\icons\mynew.bmp", "c:\lotus\wordpro\scripts\

mynew.lss"

' Add the icon to the Internet icon bar

IconMgr.IconBars("InternetTools").AddIcon 2

' Set this bar to show in its context

IconMgr.IconBars("InternetTools").ShowInContext = True

'This will force a redraw of IconBars

IconMgr.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 "Its", "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(\$LwpGetObjectTypeSentence,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: ApplyGalque 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: ApplyGliches 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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS':0)} See list of classes

{button .AL('H_ADDCHILDTO_LAYOUT_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,ServerN

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_ADDPOPUPPOINTSIEITEM_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: AddTOGEntry method

{button ,AL('H_TOCSUPERTABLELAYOUT_CLASS;H_WPAPPLICATION_CLASS';0)}

See list of classes

{button ,AL('H_ADDTOGENTRY_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].WPAApplication.AddTOGEntry(TOGEntry)

[objectreference].TOCSuperTableLayout.AddTOGEntry()

Parameters

TOGEntry

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_ADVISEONSAVE_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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

fbutton ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGR
OUPLAYOUT_CLASS;H_GELLYAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPCONTAIN
R_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTER
LAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_
FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_GL
ASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTECONTAINER_CLASS
;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELAYOUT_CLASS;
H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGRO
UPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_RUB

YLAYOUT_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_
CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGROUPLAYOUT_GL
ASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEHEA
DINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;H_T
EXT_CLASS;H_TEXTMARKER_CLASS;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

Cancels 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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_BASETABLE_CLASS;H_GELLCONTAINER_CLASS;H_FOOTNOTETA

BLE_CLASS;H_GLOSSARY_CLASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CL

ASS;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:

D L M

G o t S

A u W

/ s o r

R M d

F a f e

I n r

u W

scin

ri d

pt e

z. w

x s

1.

0

D L M

I o t S

F u W

s o r

Q d

rg f e

a r

ni W

z in

er d

1. e

~~x w~~

~~s~~

~~2~~

~~0~~

~~D L M~~

~~is of S~~

~~plu W~~

~~a s or~~

~~y Wd~~

~~Worfo~~

~~ritd r~~

~~e PrW~~

~~e in~~

~~d~~

~~e~~

~~w~~

~~s~~

~~6~~

~~0~~

~~H L M~~

~~T of S~~

~~Mu W~~

~~L s or~~

~~Wd~~

~~orfo~~

~~d r~~

~~PrW~~

~~e in~~

~~S d~~

~~m e~~

~~arw~~

~~t s~~

~~M 9~~

~~a 5~~

~~st 7~~

er 0
L MM
otS S
u E W
s xcor
1 el d
= P
2 a
= d
3 1.
0
L MO
otS ffi
u E e
s xce
1 el W
= 3. rit
2 0 er
= 4.
3 5.
fo 6
r
0
S
/2
L MRi
otS c
u E h
s xcT
1 el e
= 4. xt
2 0 F
= or
3 m
R at

3 (
 R
 I
 E)

L MS
 otS A
 u E M
 s xcN
 1 el A
 - 5. W
 2 0 or
 - d

3

R

4.

5

L MW
 otS or
 u E d
 s xcP
 1 el er
 - 7. fe
 2 0 et
 - 5.
 3 0

R

6

L MW
 otS or
 u Wd
 s in P
 A d er
 me fe
 i w et
 P s 5.

r W1

e rit

e

3.

x

L MW

otS or

u Wd

s orP

A d er

mfo fe

i r et

P D 6.

r O x

e S

3.3.

x 4.

M5.

a 6

er

e

L MW

otS or

u Wd

s orSt

A d ar

mfo 2

i r 0

P O 0

r S/O

e 2 R

3. 3

x

S

ty

le
s

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

WPApplication 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_DIVI

SION_CLASS;H_FONT_CLASS;H_FOUNDRY_CLASS;H_MERGEOPTIONS_CLASS;

H_OUTLINESTYLESEQUENCE_CLASS;H_PARAGRAPHSTYLE_CLASS;H_TEXT_G

LASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLE

LAYOUT_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].TOGSuperTableLayout.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 WPAplication, 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_DOCWIN

DOW_CLASS;H_STATUSBAR_CLASS;H_TEXTDOCUMENT_CLASS;H_WINDOW_C

LASS;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:

DL M

Got S

A u W

/ s or

RM d

F a fe

I n r

u W

scin

ri d

pt e

z. w

x s

1.

0

DL M

! ot S

F u W

~~s or~~

~~O d~~

~~rg fo~~

~~a r~~

~~ni W~~

~~z in~~

~~er d~~

~~1. o~~

~~x w~~

~~s~~

~~2.~~

~~o~~

~~D L M~~

~~is ot S~~

~~plu W~~

~~a s or~~

~~y W d~~

~~W or fo~~

~~rit d r~~

~~e Pr W~~

~~o in~~

~~d~~

~~e~~

~~w~~

~~s~~

~~6.~~

~~o~~

~~H L M~~

~~I ot S~~

~~Mu W~~

~~L s or~~

~~W d~~

~~or fo~~

~~d r~~

~~Pr W~~

e in
S d
m e
ar w
t s
M 9
a 5
st 7
er 0
L M M
ot S S
u E W
s x e or
1 e d
= P
2 a
= d
3 1
0
L M 0
ot S ffi
u E e
s x e
1 e W
= 3 rit
2 0 er
= 4
3 5
fo 6
r
0
S
/2
L M Ri
ot S e

u E h
s-xcT
1 el e
- 4. xt
2 0 F
= or
3 m
R at
3 (
R
F
F)
L MS
otS A
u E M
s-xcN
1 el A
- 5. W
2 0 or
= d
3
R
4.
5
L MW
otS or
u E d
s-xcP
1 el or
- 7. fe
2 0 et
= 5.
3 0
R
6

L MW
otS or
u Wd
s in P
A d er
mo fe
i w et
P s 5.
r W1
e rit

e

3.

x

L MW
otS or
u Wd
s or P
A d er
mfo fe
i r et
P D 6.
r Q x
e S

3.3.

x 4.

M5.

a 6

er

e

L MW
otS or
u Wd
s or St
A d ar
mfo 2

~~i r 0~~
~~P 0 0~~
~~r S/0~~
~~o-2 R~~
~~3. 3~~
~~x~~
~~S~~
~~ty~~
~~le~~
~~s~~

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_GLOSSARY_G

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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:

[WPApplication]

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_WPAPPLICATI

ON_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_BASETABLE_CLASS;H_DIVISION_CLASS;H_FOOTNOTETABLE_CLASS;H_FOUNDRY_CLASS;H_GLOSSARY_CLASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TEXTDOCUMENT_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:

D L M

G o t S

A u W

/ s o r

R M d

F a f o

I n r

u W

s c i n

r i d

p t e

z. w

x s

t.

o

D L M

I o t S

F u W

s o r

Q d

rg fo

a r

ni W

z in

er d

1. e

x w

s

2.

0

DL M

is of S

plu W

a s or

y Wd

Wor fo

rit d r

e PrW

e in

d

e

w

s

6.

0

HL M

I of S

Mu W

L s or

Wd

or fo

d r

PrW

e in

S d
m e
ar w
t s
M 9
a 5
st 7
er 0
L M M
ot S S
u E W
s x cor
1 e d
- P
2 a
- d
3 1
0
L M O
ot S ffi
u E e
s x ce
1 e W
- 3 rit
2 0 er
- 4
3 5
fo 6
r
O
S
/2
L M Ri
ot S e
u E h

s-xcT

1 el e

- 4. xt

2 0 F

= or

3 m

R at

3 (

R

I

E)

L M S

otS A

u E M

s-xcN

1 el A

- 5. W

2 0 or

= d

3

R

4.

5

L M W

otS or

u E d

s-xcP

1 el er

- 7. fe

2 0 et

= 5.

3 0

R

6

L M W

~~otS-or~~

~~u Wd~~

~~s-in P~~

~~A d er~~

~~mo fe~~

~~i w et~~

~~P s 5.~~

~~r W1~~

~~o rit~~

~~e~~

~~3.~~

~~x~~

~~L MW~~

~~otS-or~~

~~u Wd~~

~~s-or P~~

~~A d er~~

~~mfo fe~~

~~i r et~~

~~P D 6.~~

~~r Q x~~

~~e S~~

~~3.3.~~

~~x 4.~~

~~M5.~~

~~a 6~~

~~er~~

~~e~~

~~L MW~~

~~otS-or~~

~~u Wd~~

~~s-or St~~

~~A d ar~~

~~mfo 2~~

~~i r 0~~

P 0 0
r S/0
0-2 R
3. 3
x
S
ty
le
s

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:

D L M

G o t S

A u W

/ s o r

R M d

F a f e

I n r

u W

s c i n

ri d

pt e

z. w

x s

1.

0

DL M

l et S

F u W

s or

Q d

rg fo

a r

ni W

z in

er d

1. e

x w

s

2.

0

DL M

is et S

pl u W

a s or

y W d

Wor fo

rit d r

e Pr W

e in

d

e

w

s

6.

0
H L M
T ot S
Mu W
L s or
Wd
or fo
d r
PrW
o in
S d
m o
arw
t s
M 9
a 5
st 7.
er 0
L M M
ot S S
u E W
s xcor
1 el d
= P
2 a
= d
3 1.
0
L M O
ot S ffi
u E e
s xce
1 el W
= 3. fit
2 0 er

= 4
~~3~~ 5
~~6~~ 6
~~r~~
Q
S
/2
L M Ri
otS-c
u E h-
s-xcT
1 el e
- 4 xt
2 0 F
= or
~~3~~ m
R at
3 (
R
T
F)
L M S
otS-A
u E M
s-xcN
1 el A
- 5 W
2 0 or
= d
~~3~~
R
4
5
L M W
otS-or

u E d
s-xcP
1 el er
- 7. fe
2 0 et
- 5.
3 0
R
6
L MW
otS-or
u Wd
s-in P
A d er
mo fe
i w et
P s 5.
r W1
e rit
e
3.
x
L MW
otS-or
u Wd
s-orP
A d er
mfo fe
i r et
P D 6.
r Q x
e-S
3.3.
x 4.
M5.

~~a 6~~
~~er~~
~~e~~
~~L MW~~
~~otS or~~
~~u Wd~~
~~s or St~~
~~A d ar~~
~~mfo 2~~
~~i r 0~~
~~P 0 0~~
~~r S/0~~
~~e 2 R~~
~~3. 3~~
~~x~~
~~S~~
~~ty~~
~~le~~
~~s~~

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.

LwpButtonSupportDbfClick (&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

ValEff

ue ect

\$L An

wp ass

Ed ign
itA ed
bil edit
Ed or
itG can
urr wor
entk in
Or the
Ne cur
wVren
er t
{2 doc
23 um
} ent
or
any
ne
w
ver
sio
ns
of
the
doc
um
ent.
\$L An
wp ass
Ed ign
itA ed
bil edit
Ed or
itG can
urr onl
enty

~~Ve wor~~
~~rsi k in~~
~~on the~~
~~On cur~~
~~ly ren~~
~~(2 t~~
~~21 ver~~
~~)} sio~~
~~n of~~
~~the~~
~~doc~~
~~um~~
~~ent~~
~~and~~
~~can~~
~~not~~
~~edit~~
~~pre~~
~~vio~~
~~us~~
~~ver~~
~~sio~~
~~ns~~
~~of~~
~~the~~
~~doc~~
~~um~~
~~ent.~~

~~\$L A~~
~~wp spe~~
~~Ed cifi~~
~~itA e~~
~~bil edit~~
~~Ed or~~
~~itin can~~

~~gN not~~
~~ot edit~~
~~All the~~
~~ow doc~~
~~ed um~~
~~{2 ent.~~
~~20~~
~~}~~
~~\$L An-~~
~~wp ass~~
~~Ed ign~~
~~itA ed-~~
~~bil edit~~
~~Ed or-~~
~~itN can~~
~~ew onl~~
~~Ve y-~~
~~rsi wor~~
~~on kin~~
~~sO a-~~
~~nly ne~~
~~{2 w-~~
~~22 ver~~
~~}~~ ~~sio~~
~~n-of~~
~~a-~~
~~doc~~
~~um~~
~~ent-~~
~~that~~
~~is-~~
~~aut~~
~~om~~
~~atic~~
~~ally~~

cre
ate
d-
wh
en-
the-
doc
um
ent
ope
ns-
Th
e-
edit
or-
can
revi
ew-
pre
vio
us-
ver
sio
ns-
of
the-
doc
um
ent-

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ValEff

ue ect

\$LtSet

sA ting

ee this

es val
sR ue-
igh ass
ts ign
Pu s-
bli pub
e lic-
(1 acc
05 ess
69 righ
64 ts-
81 to-a
4) lay
out
obj
ect.
\$LtSet
sA ting
ee this
es val
sR ue-
igh ass
ts ign
Pri s-
vat priv
e ate-
(1 acc
05 ess
69 righ
64 ts-
81 to-a
5) lay
out
obj
ect.

if
you
set
a
lay
out
s
acc
ess
ri
gh
ts
to
priv
ate.
you
can
not
acc
ess
a
lay
out
obj
ect
by
enu
me
rati
ng
its
coll
ecti
on.
Inst
ead

u
you
mu
st
acc
ess
the
lay
out
by
ind
exi
ng
into
its
coll
ecti
on
wit
h
the
lay
out
obj
ect'
s
na
me.

\$LtSet
sA ting
ce this
es val
sR ue
igh ass
ts ign

Pr s
ote pro
te tect
d ed
(1 acc
05 ess
69 ri gh
64 ts
81 to a
6) lay
out
obj
ect
if
you
set
a
lay
out
s
acc
ess
ri gh
ts
to
pro
tect
ed
you
can
not
acc
ess
a
lay

~~out~~
~~obj~~
~~ect~~
~~by~~
~~enu~~
~~me~~
~~rati~~
~~ng~~
~~its~~
~~coll~~
~~ecti~~
~~on.~~
~~Inst~~
~~ead~~
=

~~you~~
~~mu~~
~~st~~
~~acc~~
~~ess~~
~~the~~
~~lay~~
~~out~~
~~by~~
~~ind~~
~~exi~~
~~ng~~
~~into~~
~~its~~
~~coll~~
~~ecti~~
~~on~~
~~wit~~
~~h~~
~~the~~

lay
out
obj
ect
s
na
me.

Usage

Word Pro: Active property

{button ,AL('H_APPLICATIONWINDOW_CLASS;H_DOCWINDOW_CLASS;H_STATUS

BAR_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

ValEff

ue ect

\$L Set

wp ting

W this

he val

re ue

~~Ty~~ ~~atta~~
~~pe~~ ~~che~~
~~Up~~ ~~s-~~
~~pe~~ ~~the~~
~~rL~~ ~~kno~~
~~eff~~ ~~t in~~
~~(1~~ ~~the~~
~~97~~ ~~upp~~
~~6)~~ ~~er~~
~~left~~
~~sid~~
~~e of~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~
~~\$L~~ ~~Set~~
~~wp~~ ~~ting~~
~~W~~ ~~this~~
~~he~~ ~~val~~
~~re~~ ~~ue-~~
~~Ty~~ ~~atta~~
~~pe~~ ~~che~~
~~Mi~~ ~~s-~~
~~ddl~~ ~~the~~
~~eT~~ ~~kno~~
~~op~~ ~~t in~~
~~(1~~ ~~the~~
~~97~~ ~~mid~~
~~7)~~ ~~dle-~~
~~of~~
~~the~~
~~top~~
~~are~~

a of

a

lay

out

obj

ect.

\$L Set

wp ting

W this

he val

re ue

Ty atta

pe che

La s

yo the

ut kno

Up t in

pe the

rRi upp

ghter

(1 righ

97 t

8) sid

e of

a

lay

out

obj

ect.

\$L Set

wp ting

W this

he val

re ue

Ty atta

~~pe che~~
~~Mi s~~
~~ddl the~~
~~eL kno~~
~~eft t in~~
~~(1 the~~
~~97 mid~~
~~9) dle~~
~~of~~
~~the~~
~~left~~
~~sid~~
~~e of~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~
~~\$L Set~~
~~wp ting~~
~~W this~~
~~he val~~
~~re ue~~
~~Ty atta~~
~~pe che~~
~~Mi s~~
~~ddl the~~
~~eR kno~~
~~ight in~~
~~t the~~
~~(1 mid~~
~~98 dle~~
~~0) of~~
~~the~~
~~righ~~

~~t~~
~~sid~~
~~e of~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~

~~\$L Set~~
~~wp ting~~
~~W this~~
~~he val~~
~~re ue~~
~~Ty atta~~
~~pe che~~
~~Lo s~~
~~we the~~
~~rL kno~~
~~eft t in~~
~~(1 the~~
~~98 low~~

~~1) er~~
~~left~~
~~sid~~
~~e of~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~

~~\$L Set~~
~~wp ting~~
~~W this~~
~~he val~~
~~re ue~~

~~Ty~~ ~~atta~~
~~pe~~ ~~che~~
~~Mi~~ ~~s-~~
~~ddl~~ ~~the~~
~~eB~~ ~~kno~~
~~ott~~ ~~t in-~~
~~om~~ ~~the~~
(1 ~~mid~~
98 ~~dle~~
2) ~~of~~
~~the~~
~~bott~~
~~om-~~
~~are~~
~~a~~ ~~of~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~
\$L ~~Set~~
~~wp~~ ~~ting~~
~~W~~ ~~this~~
~~he~~ ~~val~~
~~re~~ ~~ue-~~
~~Ty~~ ~~atta~~
~~pe~~ ~~che~~
~~Lo~~ ~~s-~~
~~we~~ ~~the~~
~~rRi~~ ~~kno~~
~~ghtt~~ ~~in-~~
(1 ~~the~~
98 ~~low~~
3) ~~er~~
~~righ~~

~~t~~
~~sid~~
~~e of~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~
~~\$L Set~~
~~writing~~
~~W this~~
~~he val~~
~~re ue~~
~~Ty atta~~
~~pe che~~
~~Mi s~~
~~ddl the~~
~~e kno~~
~~(1 t in~~
~~98 the~~
~~4) mid~~
~~dle~~
~~of a~~
~~lay~~
~~out~~
~~obj~~
~~ect.~~

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

amttorotatecontentvalue = [objectreference].AmtToRotateContent

[objectreference].AmtToRotateContent = amttorotatecontentvalue

Legal values

The legal values for this property are determined by its data type. For 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ValEff

ue ect

\$L Set

wp ting

W this

he val

re ue

~~Ty~~ ~~atta~~

~~pe~~ ~~che~~

~~Up~~ ~~s-~~

~~pe~~ ~~the~~

~~rL~~ ~~kno~~

~~eff~~ ~~t to~~

~~(1~~ ~~the~~

~~97~~ ~~inte~~

~~6)~~ ~~rior~~

~~of~~ ~~a~~

~~lay~~

~~out~~

~~obj~~

~~ect.~~

~~\$L~~ ~~Set~~

~~wp~~ ~~ting~~

~~W~~ ~~this~~

~~he~~ ~~val~~

~~re~~ ~~ue-~~

~~Ty~~ ~~atta~~

~~pe~~ ~~che~~

~~Mi~~ ~~s-~~

~~ddl~~ ~~the~~

~~eT~~ ~~kno~~

~~op~~ ~~t to~~

~~(1~~ ~~the~~

~~97~~ ~~ext~~

~~7)~~ ~~erio~~

~~r~~ ~~of~~

~~a~~

~~lay~~

~~out~~

~~obj~~

~~ect.~~

~~\$L~~ ~~Set~~

~~wp ting~~
~~W this~~
~~he val~~
~~re ue~~
~~Ty atta~~
~~pe che~~
~~La s~~
~~yo the~~
~~ut kno~~
~~Up t to~~
~~pe the~~
~~rRi bor~~
~~ghtder~~
~~(1 of a~~
~~97 lay~~
~~8) out~~
~~obj~~
~~ect.~~

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: ApplyGliches property

{button ,AL('H_GRAMMAR_CLASS',0)} See list of classes

{button ,AL('H_APPLYGLICHES_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

applylichesvalue = [objectreference].ApplyGliches

[objectreference].ApplyGliches = applylichesvalue

Legal values

The legal values for this property are -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_APPLYMISPELLEDEXPRESSIONS_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_EXSCRIP

T',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_APPLYMISSPELLEDITALIAN_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

applymisusedwordsvalue = [objectreference].ApplyMisusedWords

[objectreference].ApplyMisusedWords = applymisusedwordsvalue

Legal values

The legal values for this property are -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_APPLYNONSTANDARD_EXPRESSION_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

applysplinitinfinitivesvalue = [objectreference].ApplySplitInfinitives

[objectreference].ApplySplitInfinitives = applysplinitinfinitivesvalue

Legal values

The legal values for this property are -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: AnyOLEDDLinks property

'This example script has not yet been created.'

Example: ApplySplitInfinitives 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: ParagraphHasDropGap 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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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:

ValEff
ue ect
0 Indi
cat
es
that
the
Tex
tOri
ent
pro
per
ty
of
the
Lay

~~out~~
~~obj~~
~~ect~~
~~is~~
~~set~~
~~to~~
~~\$L~~
~~wp~~
~~Tex~~
~~tOri~~
~~ent~~
~~Left~~
~~tori~~
~~ght~~
~~Top~~
~~tob~~
~~otte~~
~~m~~

~~1 Indi~~
~~cat~~
~~es~~
~~that~~
~~the~~
~~Tex~~
~~tOri~~
~~ent~~
~~pro~~
~~per~~
~~ty~~
~~of~~
~~the~~
~~Lay~~
~~out~~
~~obj~~
~~ect~~

~~is~~
~~set~~
~~to~~
~~\$L~~
~~wp~~
~~Tex~~
~~tOri~~
~~ent~~
~~Top~~
~~tob~~
~~otto~~
~~mR~~
~~ight~~
~~tole~~
~~ft~~

2 ~~Indi~~
~~cat~~
~~es~~
~~that~~
~~the~~
~~Tex~~
~~tOri~~
~~ent~~
~~pro~~
~~per~~
~~ty~~
~~of~~
~~the~~
~~Lay~~
~~out~~
~~obj~~
~~ect~~
~~is~~
~~set~~
~~to~~

\$L

wp

Tex

tOri

ent

Rig

htto

left

Bot

tom

toto

p

3 Indi

cat

es

that

the

Tex

tOri

ent

pro

per

ty

of

the

Lay

out

obj

ect

is

set

to

\$L

wp

Tex

tOri
ent
Bot
tom
toto
pLe
ftto
righ
t

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_APPLYKSPLITINFINITIVES_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_ASCIIODEPAGE_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_CANGREATEPREVIEWBITMAP_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_G
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ValEff

ue ect

\$L Eq

wp uiv

Dr ale

op nt

ca to

~~pA cho~~
~~bo o*si*~~
~~ve ng~~
~~(2 Ab~~
~~23 ove~~
~~5) first~~
~~line~~
~~in~~
~~the~~
~~Dro~~
~~p~~
~~Ca~~
~~p~~
~~dial~~
~~og~~
~~box~~

=

~~\$L Eq~~
~~wp uiv~~
~~Dr ale~~
~~op nt~~
~~ca to~~
~~pB cho~~
~~elo o*si*~~
~~w ng~~
~~(2 Bel~~
~~23 ow~~
~~4) first~~
~~line~~
~~in~~
~~the~~
~~Dro~~
~~p~~
~~Ca~~
~~p~~

dial

eg-

box

z

\$L Eq

wp uiv

Dr ale

op nt

ca to-

pB cho

esi osi

de ng-

(2 Be

23 sid

6) e-

par

agr

aph

in-

the-

Dro

p-

Ca

p-

dial

eg-

box

z

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_DROPSTYLES_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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: OGXDesignMode property

{button ,AL('H_DOCCONTROL_CLASS',0)} See list of classes

{button ,AL('H_OGXDESIGNMODE_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_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_GLOSSARY_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_
CELLLAYOUT_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOU
T_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTE_LAYOUT_CLASS;H_FOOTER
LAYOUT_CLASS;H_FOOTNOTE_LAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLAS
S;H_FRAME_LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADER_LAYOUT_CLA
SS;H_LAYOUT_CLASS;H_NOTE_LAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_ROW
GROUPLAYOUT_CLASS;H_ROW_LAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUP
ERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLE_LAYOUT_CLASS;H_TABLEHEA
DING_LAYOUT_CLASS;H_TABLE_LAYOUT_CLASS;H_TOCSUPERTABLE_LAYOUT_CL
ASS';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

WPAApplication 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 WPAApplication class in the

ApplicationWindow property. WPAApplication 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.

WPApplication 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

WPAApplication 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

WPAApplication 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 WPAApplication object. You can never have more than one kind of container object at any given time.

The container object properties on the WPAApplication 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

ParallelGolsContainer 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

TOGSuperTableLayout

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

WPApplication 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

WPAApplication 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

WPAApplication 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

BaseObjectMarker

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

WPAApplication 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 WPAApplication 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

WPAApplication 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

WPAApplication 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\FootnoteNumOpt

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\FootnoteNumOpt

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\FootnoteNumOpt

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

WPAApplication 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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAplication

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 WPAApplication

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 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

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 WPAApplication

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 WPAApplication

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 WPAApplication

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

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

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 WPAplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAplication

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 WPAplication

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 WPAApplication

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

AnyOLEDDDELinks AS Integer (Boolean)

Application AS WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 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: 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 WPAplication

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 WPAplication

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 WPAApplication

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 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: FootnoteCollection class members

Properties

Application AS WPAApplication

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 WPAApplication

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 WPAplication

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 WPAApplication

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_WPAPP

LIGATION_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_G

RAPHICOLEBJECT_CLASS;H_OLEOBJECT_CLASS;H_PARAGRAPHSTYLE_GLA

SS;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_PARAGRAPH

STYLE_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 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 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_G

LASS;H_TEXTMARKER_CLASS' 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

fbutton ,AL('H_AMIKAKE_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT
_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_
DROPCAPLAYOUT_CLASS;H_ENDNOTE_LAYOUT_CLASS;H_FOOTERLAYOUT_GL
ASS;H_FOOTNOTE_LAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROW_LAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLE_LAYOUT_CLASS;H_TABLEFILL_CLASS;H_
TABLEHEADING_LAYOUT_CLASS;H_TABLE_LAYOUT_CLASS;H_TOCSUPERTABLEL
AYOUT_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].BoomarksByMarkerName

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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACT

ERBORDER_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT

_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERL

AYOUT_CLASS;H_FOOTNOTECONTSEP_CLASS;H_FOOTNOTELAYOUT_CLASS;H

_FOOTNOTESEPARATOR_CLASS;H_FOOTNOTESEPOPT_CLASS;H_FRAMEGRO

UPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEA

DERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOU

T_CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_RO

WLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CL

ASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABL

ELAYOUT_CLASS;H_TABLELINE_CLASS;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_CLAS

S;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_CLAS

S;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 WPApplication 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 WPApplication 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 WPApplication 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 WPAplication 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 WPAplication 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

GlassesH_WORD_PRO_LOTUSSCRIPT_COLLECTION_CLASSES_OVER.

Word Pro: CellLayouts property

{button ,AL('H_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_FOUNDRY_GL

ASS;H_GLOSSARY_CLASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_T

ABLEHEADING_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_BASETABLE_CLASS;H_GELLCONTAINER_CLASS;H_FOOTNOTETA

BLE_CLASS;H_GLOSSARY_CLASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CL

ASS;H_TABLEHEADING_CLASS';0)} See list of classes

{button ,AL('H_GELLLAYOUT_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_PARAGRAPH

STYLE_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 CharSet 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:

^ Fin

? ds

A and

n repl

y ace

o s

n any

e one

e cha

h ract

a er

r Ga

a n

e be

t use

e d

r with
oth
er
text
to
find
and
repl
ace
vari
atio
ns
of a
wor
d.
△ Fin
* ds
A and
n repl
y ace
e s
h zer
a o or
r mor
a e
e cha
t ract
e ers
r in a
s wor
d.
Ga
n
be
use

d-
with
oth
er-
text
to-
find
and
repl
ace
vari
atio
ns-
of a
wor
d-
^ Fin
+ ds-
P and
h repl
f ace
a s-
s zer
e o or
mor
e-
cha
ract
ers-
acr
oss
mul
tipl
e-
wor

ds-

Ga

n-

be-

use

d-

with

oth

er-

text

to-

find

and

repl

ace

phr

ase

s-

^ Fin

p ds-

T and

o repl

e ace

n s-

d zer

o o-or

f mor

p e-

a cha

r ract

a ers-

g aer

f oss

a mul

p tipl

h e-
wor
ds-
to-
the-
end
of a
par
agr
aph

-
ΔtFin
T ds-
a and
b repl
ε ace
h s-
a spa
f ces
a with
ε a-
t tab-
e cha
r ract
er.

Δ Fin
r ds-
Rand
e repl
t ace
u s-a
r har
n d-
ε retu
h m.

a

f

a

e

t

e

f

△ Fin

△ ds

△ and

e repl

h ace

a s

r the

a △

e cha

t ract

e er

f

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 WPApplication 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 WPApplication 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 WPApplication 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_CLA

SS;H_INDEXSECTION_CLASS;H_NOTELAYOUT_CLASS;H_NUMERICFORMATSUB

SET_CLASS;H_SECTION_CLASS;H_SHADOW_CLASS';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_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_GLOSSARY_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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 WPApplication 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 WPApplication 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 WPApplication 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.BackgroundColor.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_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_GLOSSARY_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_G
LASS;H_ENDNOTE_LAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTE_G
LASS;H_FOOTNOTE_LAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAM
ELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAY
OUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPL
AYOUT_CLASS;H_ROW_LAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABL
EGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLA
YOUT_CLASS;H_TABLELAYOUT_CLASS;H_TOCSUPERTABLELAYOUT_CLASS;H_
WPAPPLICATION_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_G

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S;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_G

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S;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_G

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S;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-

ferences 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_CLICKHE

RE_CLASS;H_DIVISION_CLASS;H_DROPDOWNCONTAINER_CLASS;H_FRAMECON

TAINER_CLASS;H_MARKER_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTA

INER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_POWERFIELD_CLASS;H_

ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYSPLITTER_CLASS;

H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPER

ABLECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEMARKER_CLAS

S;H_TABLEONLYCONT_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_GL

ASS';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_FORMATPRE

FERENCES_CLASS;H_FORMULA_CLASS;H_GRAPHIC_CLASS;H_GRAPHICOLEO

BJECT_CLASS;H_OLEOBJECT_CLASS;H_PARAGRAPHSTYLE_CLASS;H_TEXT_G

LASS;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 WPApplication 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 WPApplication 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 WPApplication 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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 WPApplication 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 WPApplication 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 WPApplication 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: RemoveTOGEntry 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: SetTOGLLevelContent method

'This example script has not yet been created.'

Example: SetTOGLLevelPageInfo 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_ADDAGCONDITION_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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: DeleteItemByPosition 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_GOTOCLICKHERE_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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.

lf Us
yo e
u this
areval
im ue
porfor
tin the
g Fil
thi eF
s or
typ ma
e t
of par
graam
phi ete
e: f:
W .w
or mf
dP
re
W
M
F
W .ep
or s
dP
re
E

P
S
W .b
or mp
dP
re
Pa
int
W .tif
or
dP
rol
m
ag
e
W .sd
or w
dP
re
Dr
aw
Be .be
xe x
s
W .te
or x
dP
re
Eq
n
W .pe
or x
dP
re
P

G
X
W :gif
or
dP
re
Gl
F
G :cg
G m
M
H :plt
P
G
L
P :pe
G d
D
D :dr
R w
W
G :cd
D r
R
Pl :pi
G e
Im :w
p pg
W
P
G
Ex :sd
pS w
D
W

z
W
P
G
Ex .w
p mf
W
M
F2
W
P
G
Ex .b
pB mp
M
P2
W
P
G
Im .w
p p2
W
P2
Ex .sd
pS w
D
W
z
W
P2
Ex .w
p mf
W
M
F2

W
P2
Ex :b
pB mp
M
P2
W
P2
Im :jp
pJ g
P
G
Ex :b
pJ mp
P
G

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_REMOVETOENTRY_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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.

ValEff

ue ect

&HAllo

01 ws

you

to

set

the

left

ma

rgin
val
ue.
&HAllo
02 ws
you
to
set
the
righ
t
ma
rgin
val
ue.
&HAllo
04 ws
you
to
set
the
top
ma
rgin
val
ue.
&HAllo
08 ws
you
to
set
the
bott
om
ma

rgin

val

ue.

&HAllo

10 ws-

you

to-

set

the-

left

ext

ern

al-

ma

rgin

val

ue.

&HAllo

20 ws-

you

to-

set

the-

right

t

ext

ern

al-

ma

rgin

val

ue.

&HAllo

40 ws-

you

~~to~~
~~set~~
~~the~~
~~top~~
~~ext~~
~~ern~~
~~al~~
~~ma~~
~~rgin~~
~~val~~
~~ue.~~
~~&HAllo~~
~~80 ws~~
~~you~~
~~to~~
~~set~~
~~the~~
~~bott~~
~~om~~
~~ext~~
~~ern~~
~~al~~
~~ma~~
~~rgin~~
~~val~~
~~ue.~~

!

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: SetTOGLevelContent method

{button ,AL('H_WPAPPLICATION_CLASS':0)} See list of classes

{button ,AL('H_SETTOGLEVELCONTENT_METHOD_EXSCRIPT':1)} See example

This language element is not yet defined.

Syntax

Unknown

Parameters

Unknown

Return value

Unknown

Usage

Word Pro: SetTOGLevelPageInfo method

{button ,AL('H_WPAPPLICATION_CLASS':0)} See list of classes

{button ,AL('H_SETTOGLEVELPAGEINFO_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_CLASS',0)} See list of classes

CONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

S;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

CONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEONLYCONTAINER_CLASS',0)} See list of classes

LEGCONTAINER_CLASS;H_TABLEONLYCONTAINER_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"

DefstringO = ""

IniFileTypeO = \$LwpIniUserPrefs

WhichIniLocationO = ""

IniNameO = "lwpuser.ini"

For x = 1 To 6

Key = "LastOpen" & x

Msgbox .GetProfileString(Section, Key, DefstringO, IniFileTypeO,

WhichIniLocationO, IniNameO)

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(\$LwpGetObjectTypeWord, 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 looses 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 IconPallet As String

Dim MsgStr As String

Dim IconMgr As IconBarManager

IconPallet = "Comment Tools"

CR = Chr(10)

Set IconMgr = .ApplicationWindow.IconBarManager

With IconMgr.IconBars(IconPallet)

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 redisplay 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 IconPallet As String

Dim MsgStr As String

Dim IconMgr As IconBarManager

IconPallet = "Comment Tools"

Set IconMgr = .ApplicationWindow.IconBarManager

' Set the context and show the bar

IconMgr.IconBars(IconPallet).ShowInContext = True

IconMgr.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.'

IconMgr.IconBars(IcnPallet).ShowInContext = False

IconMgr.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 IconPallet As String

Dim MsgStr As String

Dim IconMgr As IconBarManager

IconPallet = "Comment Tools"

CR = Chr(10)

Set IconMgr = .ApplicationWindow.IconBarManager

With IconMgr.IconBars(IconPallet)

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_APPLIC

ATIONWINDOW_CLASS;H_DOCWINDOW_CLASS';0)} See list of classes

{button ,AL('H_APPLICATIONWINDOW_DOCWINDOW_HWND_PROPERTY_EXSCRI

PT',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_FIELDDELIMITERTEXT_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

ValEff
ue ect
\$L Onl
wp y
Fil tho
eP se
rot list
ected
Ty as
pe edit
Ed ors
itor on
s the
{2 Edi
61 ting
} Rig
hts
pan
el
can
ope
n
the
doc
um
ent.
\$L An
wp yon
Fil e

~~eP~~ ~~wit~~
~~rot~~ ~~h-~~
~~ect~~ ~~acc~~
~~Ty~~ ~~ess~~
~~pe~~ ~~to-~~
~~No~~ ~~the-~~
~~ne~~ ~~fold~~
~~{2~~ ~~er~~
~~60~~ ~~wh~~
~~}~~ ~~ere~~
~~the-~~
~~doc~~
~~um~~
~~ent~~
~~is-~~
~~stor~~
~~ed-~~
~~can~~
~~ope~~
~~n-~~
~~the-~~
~~doc~~
~~um~~
~~ent.~~
~~\$L~~ ~~Onl~~
~~wp~~ ~~y-~~
~~Fil~~ ~~the-~~
~~eP~~ ~~per~~
~~rot~~ ~~son~~
~~ect~~ ~~wh~~
~~Ty~~ ~~e-~~
~~pe~~ ~~ere~~
~~Ori~~ ~~ate~~
~~gA~~ ~~d-~~
~~uth~~ ~~the-~~

~~or doc~~
~~(2 um~~
~~62 ent~~
~~) can~~
~~ope~~
~~n it.~~

~~\$L An~~
~~wp yon~~
~~File~~
~~eP wh~~
~~rot o~~
~~ect kno~~
~~Ty ws~~
~~pe the~~
~~Pa doc~~

~~ss um~~
~~we ent~~
~~rd pas~~
~~(2 sw~~
~~63 ord~~
~~) can~~

~~ope~~
~~n~~
~~the~~
~~doc~~
~~um~~
~~ent.~~

~~\$L We~~
~~wp rd~~
~~File Pro~~
~~eP use~~
~~rot s~~
~~ect this~~
~~Ty typ~~
~~pe e~~

~~Re inte~~
~~serrnal~~
~~ve ty~~
~~d wh~~
~~(2 ene~~
~~64 ver~~
~~) the~~
~~file~~
~~pro~~
~~tect~~
~~ion~~
~~typ~~
~~e~~
~~stor~~
~~ed~~
~~wit~~
~~hin~~
~~the~~
~~doc~~
~~um~~
~~ent~~
~~is~~
~~not~~
~~ree~~
~~ogn~~
~~ize~~
~~d.~~
~~For~~
~~exa~~
~~mpl~~
~~e.~~
~~an~~
~~exi~~
~~stin~~
~~g.~~

~~file-~~
~~pro~~
~~tect~~
~~ion-~~
~~typ~~
~~e-is~~
~~not-~~
~~rec~~
~~ogn~~
~~ize~~
~~d-in~~
~~a-~~
~~cor~~
~~rup~~
~~ted-~~
~~file-~~
~~or-~~
~~in-a~~
~~file-~~
~~cre~~
~~ate~~
~~d-~~
~~wit~~
~~h-a~~
~~late~~
~~r-~~
~~ver~~
~~sio~~
~~n-of~~
~~We~~
~~rd-~~
~~Pro~~
~~that~~
~~add~~
~~ed-~~

ne
w
veri
fica
tion
typ
es.
Be
cau
se
Wo
rd
Pro
use
s
this
pro
per
ty
typ
e
inte
rnal
ly.
a
use
r
wo
uld
nev
er
ass
ign
this
pro

per
ty
typ
e.

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

ValEff
ue ect
\$L Allo
wp ws-
Ta you
ble to-
Fill set
St a-
yle fill-
All styl
(1 e-
87 for-
0) an-
enti
re-
tabl
e-
obj
ect
\$L Allo
wp ws-
Ta you
ble to-
Fill set
St a-
yle fill-
Ev styl
erye-
oth for-

er ce ve

ol ry

(1 oth

87 er

2) col

um

n in

a

tabl

e

obj

ect.

\$L Allo

wp ws

Ta you

ble to

Fill set

St a

yle fill

Ev styl

erye

oth for

err eve

ow ry

(1 oth

87 er

1) row

in a

tabl

e

obj

ect.

\$L Allo

wp ws

Ta you

~~ble to~~
~~Fill set~~
~~St mo~~
~~yle re~~
~~Mi tha~~
~~xe n~~
~~d one~~
~~(1 typ~~
~~87 e of~~
~~3) fill~~
~~styl~~
~~e~~
~~for~~
~~a~~
~~tabl~~
~~e~~
~~cell~~
~~obj~~
~~ect.~~
~~\$L Indi~~
~~wp cat~~
~~Ta es~~
~~ble that~~
~~Fill no~~
~~St fill~~
~~yle styl~~
~~No e is~~
~~ne set~~
~~(1 for~~
~~87 a~~
~~4) tabl~~
~~e~~
~~cell~~
~~obj~~
~~ect.~~

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_FINDEXACTCASE_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

findstylenamevalue = [objectreference].FindStyleName

[objectreference].FindStyleName = findstylenamevalue

Legal values

The legal values for this property are determined by its data type. For 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_GLAS

S',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

fitypevalue = [objectreference].FitType

[objectreference].FitType = fitypevalue

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)

\$LwpNumberFormatNorwegiankrone (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_FORMUL

A_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_GETAFIDHELPINFO_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.

LwpButtonSupportDbClick (&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_GELLLAYOUT_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

gotopagenumbervalue = [objectreference].GoToPageNumber

[objectreference].GoToPageNumber = gotopagenumbervalue

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

~~ValEff~~

~~ue ect~~

~~\$L Set~~

~~wp ting~~

~~Gri this~~

~~dT val~~

~~yp ue~~

~~eD dis~~

~~ots pla~~

~~{2 ys~~

06 a-

1) spe

cifi

e-

typ

e-of

dott

ed-

grid

in-

the-

lay

out-

obj

ect.

\$L Set

wp ting

Gri this

dT val

yp ue-

eLi dis

ne pla

s- ys-

(2 a-

06 spe

0) cife

typ

e-of

line

d-

grid

in-

the-

lay

out-

obj
ect.
\$L Set
wp ting
Gri this
dT val
yp ue
eN pre
on ven
e ts a
(2 grid
05 fro
9) m
sho
win
g in
the
lay
out
obj
ect.

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_OUTLINESTYLESEQUENCE_CLASS:H_PAR

AGRAPHSTYLE_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

fbutton ,AL('H_APPLICATIONWINDOW_CLASS;H_BASECONTAINER_CLASS;H_GEL
LCONTAINER_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H
_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DOCWIND
OW_CLASS;H_DROPCAPCONTAINER_CLASS;H_DROPCAPLAYOUT_CLASS;H_E
NDNOTELAYOUT_CLASS;H_FONT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTN
OTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_FRAMEGROUPLAYOUT_G
LASS;H_FRAMELAYOUT_CLASS;H_GRAPHIC_CLASS;H_GRAPHICOLEBJECT_G
LASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_ICONBAR_GLAS
S;H_LAYOUT_CLASS;H_NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS;H_O
LEOBJECT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELAYOUT_CLASS;H_PAR

ALLELGOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGROUPLAY

OUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYLAYO

UT_CLASS;H_STATUSBAR_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPA

GECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGR

OUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLA

SS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYG

ONT_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_EXSCRI

PT',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.

\$LwplconBarPositionStateFloating (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_ICONHELPTEXT_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_CLICKHERE_CLAS

S: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

N_CLASS:H_DIVISION_CLASS:H_TEXTDOCUMENT_CLASS:H_WPAPPLICATION_C

LASS';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 WPApplication 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: ApplyHomoPhone1 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

applypretentiouswordsvalue = [objectreference].ApplyPretentiousWords

[objectreference].ApplyPretentiousWords = applypretentiouswordsvalue

Legal values

The legal values for this property are -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 thaty 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_ASCIIIRLFTYPE_PROPERTY_EXSCRIPT',1)} See example

(Read-write)

Data Type

Variant (Enumerated)

AsciiLineEnding

Syntax

asciicrlfypsvalue = [objectreference].AsciiCRLFType

[objectreference].AsciiCRLFType = asciicrlftypevalue

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

atendoflinevalue = [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

ValEff
ue ect
\$L Gre
wp ate
Au s-a
te ne
Ve w
rsi ver
on sio
Ev n-
eryand
da reu
y ses
(2 that
9) ver
sio
n-
unti
L
the
file
is
ope
ned
on-
the
fol
owi
ng
day
.If
you

lea

ve-

a-

file-

ope

n-

unti

t-

the-

nex

t-

day

=

We

rd-

Pro

doe

s-

not-

cre

ate-

a-

ne

w-

ver

sio

n-

unti

t-

you

clo

se-

and

ree

pen

~~the~~
~~file.~~
\$L Gre
wp ate
Au s-a
to ne
Ve w
rsi ver
on sio
Ev n-
eryand
we reu
ek ses
(3 that
0) ver
sio
n-
unti
t
the
file
is
ope
ned
afte
r
the
foll
owi
ng
Me
nda
y. If
you
lea

~~ve-~~

~~a-~~

~~file-~~

~~ope~~

~~n-~~

~~unti~~

~~t-~~

~~Mo~~

~~nda~~

~~y-~~

~~Wo~~

~~rd-~~

~~Pro~~

~~doe~~

~~s-~~

~~not~~

~~cre~~

~~ate-~~

~~a-~~

~~ne~~

~~w-~~

~~ver~~

~~sio~~

~~n-~~

~~unti~~

~~t-~~

~~you~~

~~clo~~

~~se-~~

~~and~~

~~ree~~

~~pen~~

~~the-~~

~~file-~~

~~In~~

We

rd

Pro

=

Mo

nda

y is

the

first

day

of

the

we

ek.

\$L Tur

wp ns

Au off

to the

Ve Aut

rsi oV

on ersi

No on

ne pro

(2 per

7) ty.

\$L Gre

wp ate

Au s-a

to ne

Ve w

rsi ver

on sio

On n

file eve

op ry

~~en tim~~
~~{2 e-~~
~~8) so~~
~~me~~
~~one~~
~~ope~~
~~ns-~~
~~the-~~
~~file-~~
~~\$L Gre~~
~~wp ate~~
~~Au s-a~~
~~to ne~~
~~Ve w-~~
~~rsi ver~~
~~on sio~~
~~On n-~~
~~file eve~~
~~sa ry-~~
~~ve tim~~
~~{3 e-~~
~~1) so~~
~~me~~
~~one~~
~~sav~~
~~es-~~
~~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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

bulletfontsvvalue = [objectreference].BulletFonts

[objectreference].BulletFonts = bulletfontsvvalue

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_CL

ASS;H_FORMULA_CLASS;H_GLOSSARY_CLASS;H_GRAPHIC_CLASS;H_GRAPHI

COLEOBJECT_CLASS;H_OLEOBJECT_CLASS;H_PARALLELCOLUMNS_CLASS;H_

SUPERTABLE_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TEXT_CLA

SS';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_MENUIT

EM_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 = centeredhorxvalue

Legal values

The legal values for this property are -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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

fbutton ,AL('H_BASSETABLE_CLASS;H_GELLGROUPLAYOUT_CLASS;H_GELLLAYO

UT_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;

H_CONTENT_CLASS;H_DIVISIONINFO_CLASS;H_DROPPLAYOUT_CLASS;H_E

NDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CL

ASS;H_FOOTNOTETABLE_CLASS;H_FORMULA_CLASS;H_FRAMEGROUPLAYOU

T_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_PA

GELAYOUT_CLASS;H_PARALLELCOLUMNS_CLASS;H_ROWGROUPLAYOUT_CLA

SS;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_TABLE

LAYOUT_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_CLEANSREENONSTARTUP_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_CLAS

S: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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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

closedocmacronamevalue = [objectreference].CloseDocMacroName

[objectreference].CloseDocMacroName = closedocmacronamevalue

Legal values

The legal values for this property are determined by its data type. For 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_
DROPGAPLAYOUT_CLASS;H_ENDNOTE_LAYOUT_CLASS;H_FOOTERLAYOUT_CL
ASS;H_FOOTNOTE_LAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTE_LAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROW_LAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLE_LAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLE_LAYOUT_CLASS;H_TOCSUPERTABLE_LAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

ValEff

ue ect

\$L Set

wp ting

Go this

ndi val

tio ue

nT ins

yp erts

eA a

~~llb cert~~
~~uts ain~~
~~pe lay~~
~~cifi out~~
~~cp obj~~
~~ag ect~~
~~e in~~
~~(1 all~~
~~56 pag~~
~~) es.~~
~~exc~~
~~ept~~
~~a~~
~~spe~~
~~cifi~~
~~e~~
~~pag~~
~~e in~~
~~a~~
~~doc~~
~~um~~
~~ent.~~
~~\$L Set~~
~~wp ting~~
~~Co this~~
~~ndi val~~
~~tio ue~~
~~nT ins~~
~~yp erts~~
~~eA a~~
~~llp cert~~
~~ag ain~~
~~es lay~~
~~(1 out~~
~~54 obj~~

~~lect~~
~~in-~~
~~all-~~
~~pag~~
~~es-~~
~~of a~~
~~doc~~
~~um~~
~~ent.~~

~~Set~~
~~working~~
~~Go this~~
~~ndival~~
~~tio ue-~~
~~nT ins~~
~~yp erts~~
~~eOa-~~
~~nly cert~~
~~ev ain-~~
~~en lay~~
~~pa out-~~
~~ge obj~~
~~s- ect~~
~~(1 in-~~
~~57 eve~~

~~n-~~
~~pag~~
~~es-~~
~~of a~~
~~doc~~
~~um~~
~~ent.~~

~~Set~~
~~working~~
~~Go this~~

ndi val
tio ue
nT ins
yp erts
eQa
nly cert
od ain
dp lay
ag out
es obj
(1 ect
58 in
) odd
pag
es
of a
doc
um
ent.
\$L Set
wp ting
Go this
ndi val
tio ue
nT ins
yp erts
eQa
nly cert
sp ain
eci lay
fic out
pa obj
ge ect
(1 in
55 spe

l cifi
e
pag
es
of a
doc
um
ent.
\$L Set
wp ting
Go this
ndi val
tio ue
nT spe
yp cifi
eS es
tar on
tat whi
pa ch
ge pag
(1 e to
59 ins
l ert
a
cert
ain
lay
out
obj
ect.

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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: BorderOffset 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.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"

.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 = \$LtsFillSolid

.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: CharSetName property

'This example script has not yet been created.

'Example: CharSet 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 it's 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_LOTUSSCR

!PT_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 WPAApplication object because the Parent property for the Table

contains a reference to WPAApplication.

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 WPAplication 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.Back

Color.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

$$\underline{1 \text{ point} = 1/72''}$$

$$\underline{1 \text{ point} = 20 \text{ twips}}$$

$$\underline{1 \text{ point} = 65536 \text{ units}}$$

Twip equivalents

$$\underline{1 \text{ twip} = 1/1440''}$$

$$\underline{1 \text{ twip} = 1/20 \text{ point}}$$

$$\underline{1 \text{ twip} = 3276.8 \text{ units}}$$

Unit equivalents

$$\underline{1 \text{ unit} = 1/4718592''}$$

$$\underline{1 \text{ unit} = 1/65536 \text{ point}}$$

$$\underline{1 \text{ unit} = 1/3276.8 \text{ twips}}$$

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.

Lw&Lw&

p Hp H

M 6M E

en 4en 0

uF u

ile MI

m Fi

en el

u da

ut

e

Lw&Lw&

p Hp H

M 6M E

en 5en 1

u u

Mf MI

Ne Fi

w el

ds

he

wrl

t
Lw&Lw&
p Hp H
M 6M E
en 6en 2
u u
Mf MI
Op Fi
en el
dd
oa
ut
e

Lw&Lw&
p Hp H
M 6M E
en 7en 3
u u
Mf MI
Sa Fi
ve el
df
e
m
ov
e

Lw&Lw&
p Hp H
M 6M 2
en 8en 3
u u E
Mf MI
Sa Ex
ve ee
as ha

nd
Lw&Lw&
p Hp H
M 6M E
en 9en 4
u u
Mf Ml
Re In
ve de
rt xø
pt
s

Lw&Lw&
p Hp H
M 6M 2
en Aen 2
u uH6
Mf el
Ap p
pe m
nd en
tex u
t

Lw&Lw&
p Hp H
M 6M 2
en Ben 2
u u 7
Mf M
lm hl
pe dx
rt he
lp

Lw&Lw&
p Hp H

M 6M 2

en Gen 2

u u 8

Mf M

Fil hA

e bē

m ut

an

ag

e

Lw&Lw&

p Hp H

M 6M 2

en Den 2

u u 9

Mf M

De h

ed M

es aē

e rē

he

lp

Lw&Lw&

p Hp H

M 6M 2

en Een 2

u u B

Mf M

Gr hG

ea e

te m

da pa

tafi tibl

le e

Lw&Lw&

p Hp H
M 6M 3
en Fen 3
u u E
Mf M
Pri hU
nt sin
gH
el
p

Lw&Lw&
p Hp H
M 7M 3
en 0en 3
u u F
Mf M
Gh hK
gp ey
rin b0
ter ar
d

Lw&Lw&
p Hp H
M 7M 3
en 1en 4
u u 0
Mf M
Ex hH
it ow
Do
t

Lw&Lw&
p Hp H
M 7M 3
en 2en 4

u u 1

Mf M

Pri hD

ntr oH

es el

et p

Lw&Lw&

p Hp H

M 7M 3

en 3en 4

u u 2

Mf M

Dd hF

e or

Up

gr

ad

er

s

Lw&Lw&

p Hp H

M 7M 3

en 4en 4

u u 5

Mf M

M hT

er ut

ge ori

act al

ion

Lw&Lw&

p Hp H

M 7M 3

en 5en 6

u u F

Mf M
Ju hS
stp ea
rin re
t h
Lw&Lw&
p Hp H
M 1M 2
en 4en 5
u 7u 9
Mf M
St sG
an on
dal tro
on !
ep
Lw&Lw&
p Hp H
M 7M 2
en 6en 5
u u A
Mf M
All sF
fld ile
sr m
en an
a
m
e
Lw&Lw&
p Hp H
M 7M 2
en 7en 7
u u 2
Mf Mt

Pri Tb
nt lla
wr ye
on ut
ge
ap
er

Lw&Lw&
p Hp H
M 7M 2
en 8en 7
u u 3
Mf Mt

Pri In
nte se
an otr
cel ow

Lw&Lw&
p Hp H
M 7M 2
en 9en 7
u u 4
Mf Mt

Du De
m lco
my lro
mr w
gs

el
Lw&Lw&
p Hp H
M 7M 2
en Aen 7
u u 5
Mf Mt

Du Ed
m itfr
my ml
mr a
ge
xt

Lw&Lw&
p Hp H
M 7M 2
en Ben 7
u u 6
Mf Mt
As Tb
ei la
op dd
ts

Lw&Lw&
p Hp H
M 7M 2
en Gen 7
u u 7
Mf Mt
Ga He
np ad
rin in
t g

Lw&Lw&
p Hp H
M 7M 2
en Den 7
u u 9
Mf Mt
Fl Ta
dr bl
en ei

a nf
m ob
e ox
Lw&Lw&
p Hp H
M 7M 2
en Een 7
u u A
Mf Mt
Pri Si
nt ze
op col
t re
w
Lw&Lw&
p Hp H
M 7M 2
en Fen 7
u u B
Mf Mt
Sa Co
ve nn
as ee
nw t
Lw&Lw&
p Hp H
M 8M 2
en Oen 7
u u D
Mf Mt
Im Sa
pe ve
rtn fr
w ml
a

Lw&Lw&

p Hp H

M 8M 2

en 1en 8

u u 3

Mf Mt

De Q

ed uic

es ka

en dd

w ro

w

Lw&Lw&

p Hp H

M 8M 2

en 2en 8

u u 4

Mf Mt

M Q

ast uic

er ka

nw dd

col

Lw&Lw&

p Hp H

M 8M 2

en 3en 8

u u 5

Mf Mt

N De

wi leo

m tu

pe m

rt n

Lw&Lw&

p Hp H
M 8M 2
en 4en 8
u u 6
Mf Mt
M De
er Iro
ge w
vie
wp
rin
t

Lw&Lw&
p Hp H
M 8M 2
en 5en 8
u u 7
Mf Mt
La In
sto se
pe rtr
n1 ow

Lw&Lw&
p Hp H
M 8M 2
en 6en 8
u u 8
Mf Mt
La In
sto se
pe rte
n2 ol

Lw&Lw&
p Hp H
M 8M 2

en 7 en 8

u u 9

Mf Mt

La In

sto se

pe rta

n3 ee

el

Lw&Lw&

p Hp H

M 8M 2

en 8 en 8

u u A

Mf Mt

La De

sto let

pe ea

n4 ee

el

Lw&Lw&

p Hp H

M 8M 2

en 9 en 8

u u B

Mf Mt

La De

sto Ita

pe bl

n5 e

Lw&Lw&

p Hp H

M 8M 2

en A en 8

u u G

Mf Mt

Op Se
en lee
ne te
sty ol
le u
m
n

Lw&Lw&
p Hp H
M 8M 2
en Ben 8
u u D
Mf Mt
M Se
er lee
gel tro
ab w
els

Lw&Lw&
p Hp H
M 8M 2
en Gen 8
u u E
Mf Mt
Ot Se
he lee
rfl tta
ds bl
e

Lw&Lw&
p Hp H
M 8M 2
en Den 8
u u F
Mf Mt

lm Ch

pe gie

rtø øn

ict siz

ur e

e

Lw&Lw&

p Hø H

M 8M 2

en Een 9

u u 0

Mf Mt

Ex Di

pe se

rt øn

ne

et

Lw&Lw&

p Hø H

M 8M 2

en Fen 9

u u 1

Mf Mt

Att Sp

rib litc

nw ell

s

Lw&Lw&

p Hø H

M 9M 2

en 2en 9

u u 2

Mf Mt

Ga Sp

n litt

m ab
er le
ge
pri
nt

Lw&Lw&
p Hp H
M 9M 2
en 8en 9
u u 3
Mf Mt
lm Pc
pe oli
tt ns
ext col
re
w

Lw&Lw&
p Hp H
M 2M 2
en 3en 9
u 5u 4
Mf Mt
M Pc
er oli
ge ns
pri ert
nt re
op w
ts

Lw&Lw&
p Hp H
M 9M 2
en 0en 9
u u 5

Mf Mt
Gl Pe
es oli
e ns
ert
col

Lw&Lw&
p Hp H
M 2M 2
en Den 9
u Bu 6

Mf Mt
Gl Pe
es ol
eal de
t Ita
bl
e

Lw&Lw&
p Hp H
M 2M 2
en Den 9
u Gu 7

Mf Mt
Gl Pe
es ol
eal de
lre lco
pla lu
eel m
ast n

Lw&Lw&
p Hp H
M 9M 2
en 1en 9

u u 8
Mf Mt
Pa Pe
ss ol
wo de
rd lro
w

Lw&Lw&
p Hp H
M 1M 2
en 2en 9
u 2u 9
Mf Mt
Pa Pe
ss ole
wo on
rd ne
2 et

Lw&Lw&
p Hp H
M 2M 2
en Een 9
u 0u A
Mf Mt
M Pe
ast ol
er dis
de eo
eu nn
m ee
en t
t

Lw&Lw&
p Hp H
M 9M 2

en 3 en 9

u u B

Mf Mt

Gl Pe

os o/s

efil plit

e cel

ls

Lw&Lw&

p Hp H

M 9M 2

en 4 en 9

u u C

Mf Mt

Le Pe

ek o/s

el

ee

te

ol

u

m

n

Lw&Lw&

p Hp H

M 2M 2

en Den 9

u 7u D

Mf Mt

Ne Pe

xt o/s

m el

di ee

wi tre

nd w

ow
Lw&Lw&
p Hp H
M EM 2
en 5en 9
u u E
Mf Mt
M Pc
ail ols
el
ee
tta
bl
e

Lw&Lw&
p Hp H
M 2M 2
en Een 9
u 1u F
Mf Mt
M Pc
ail ol
Ne de
w lco
lro
w

Lw&Lw&
p Hp H
M EM 2
en 6en A
u u 0
Mf Mt
M Pc
air ol
ea he

d ad
in
g

Lw&Lw&

p Hp H

M EM 2

en 7en A

u u 1

Mf Mt

Qu Ge

ick llin

op fo

en be

x

Lw&Lw&

p Hp H

M EM 2

en Ben A

u u 2

Mf Mt

Fil Sh

es ow

op ta

bl

eli

ne

inf

ob

ox

Lw&Lw&

p Hp H

M 1M 2

en 2en A

u 4u 3

Mf Mt

Sa Sh
ve ow
eo ta
py bl
as esi
zei
nf
ob
ox

Lw&Lw&

p Hp H
M 1M 2
en 2en A
u 5u 4
Mf Mt
Up Gr
da ea
te tet
obj ab
ect le
wit
hg
rid

Lw&Lw&

p Hp H
M 2M 2
en Den A
u Du 5
Mf Mt
Ed Sh
ite ow
rgr pe
ee olli
t ne
inf

ob

ox

Lw&Lw&

p Hp H

M 2M 2

en Den A

u Eu 6

Mf Mt

Ne Sh

wd ow

ivi pe

sio ols

n ize

inf

ob

ox

Lw&Lw&

p Hp H

M 2M 2

en Den A

u Fu 7

Mf Mt

Op Sp

en litp

div col

isi

on

Lw&Lw&

p Hp H

M 2M 2

en Een A

u 2u 8

Mf Mt

Sa Pe

Ge ols

nti el

nu ec

eR te

ou nti

te ret

ab

le

Lw&Lw&

p Hp H

M 2M 2

en Een A

u 3u 9

Mf Mt

Sa Se

Re lec

ut te

e nti

ret

ab

le

Lw&Lw&

p Hp H

M 3M 2

en 0en A

u Gu A

Mf Mt

Se Se

lda lec

tafi te

le nti

re

eel

tra

ng

e

Lw&Lw&

p Hp H

M 3M 2

en 0en A

u Gu B

Mf Mt

Mr Se

gd lec

eli te

mit nti

re

pe

ole

ell

ra

ng

e

Lw&Lw&

p Hp H

M 3M 2

en 0en A

u Du G

Mf Mt

Mr Ta

gle bl

tte et

r op

ali

gn

Lw&Lw&

p Hp H

M 3M 2

en 1en A

u 2u D

Mf Mt

Mr Ta
ge bl
nv ee
elo en
pe ter
ali
gn

Lw&Lw&
p Hp H
M 3M 2
en 0en A
u Eu E

Mf Mt
Mr Ta
gg bl
ot eb
or ott
ee e
or m
d ali
gn

Lw&Lw&
p Hp H
M 3M 2
en 0en A
u Fu F

Mf Mt
Gr Co
ea nn
te ee
de tro
se w

Lw&Lw&
p Hp H
M 3M 2

en 1 en B

u 0 u 0

Mf Mt

Mr Pe

gs 0lc

ort on

ne

ctr

ow

Lw&Lw&

p Hp H

M 3M 2

en 1 en B

u 1 u 1

Mf Mt

Gr G

ea ot

te on

m ex

er tbl

ge oe

k

Lw&Lw&

p Hp H

M 3M 2

en 1 en B

u 3 u 2

Mf Mt

Ed Ce

itd lin

at fo

afil bo

e x

m

ou

se
Lw&Lw&
p Hp H
M 3M 2
en 1en B
u 4u 3
Mf Mt
Us Ta
ee bl
ur ei
as nf
mr ob
gd ox
ee m
ou
se

Lw&Lw&
p Hp H
M 3M 2
en 1en B
u 5u B
Mf M
Op xR
en un
an re
de xx
dit
da
tafi
le
Lw&Lw&
p Hp H
M 3M 2
en 1en B
u 6u G

Mf M

Op xR

en un

pr on

evi ti

ew m

e

Lw&Lw&

p Hp H

M 3M 2

en 1en D

u 7u 0

Mf M

Ne xH

wp ou

re rgl

vie as

w s

Lw&Lw&

p Hp H

M 3M 2

en 1en D

u 8u 1

Mx M

M x

er M

ge ae

m pa

ae us

re e

Lw&Lw&

p Hp H

M 3M 2

en 1en D

u 9u 2

Mf M
M x
er M
ge ae
pri re
nt su
all m
e

Lw&Lw&

p Hp H
M 3M 2
en 1en D
u Au 3

Mf M

M xT
er im
ge ed
op el
en ay

Lw&Lw&

p Hp H
M 3M 2
en 1en D
u Bu 4

Mf M

sR xT
un es
ser tfo
ipt re
an

eel

Lw&Lw&

p Hp H
M 3M 2
en 5en D

u Fu 5
Ml M
sG x
re M
at ac
es m
eri es
pt sa
ge
s

Lw&Lw&
p Hp H
M 3M 2
en 6en D
u 0u 6
Ml M
sG xP
re re
at an
edi sw
alo er
g m
sg
bo
x

Lw&Lw&
p Hp H
M 3M 2
en 6en D
u 5u 8
Ml M
sG x
e M
m ac
pil ca

e ne
el
Lw&Lw&
p Hp H
M 3M 2
en 6en D
u 9u 9
Ml M
sA x
ut M
oa ac
pp eo
nti
nu
e

Lw&Lw&
p Hp H
M 3M 2
en Gen D
u 0u A
Ml M
sld x
eS M
te ac
pl re
n va
ria
bl
es

Lw&Lw&
p Hp H
M 3M 3
en Gen 1
u 1u E
Ml M

sld xR
eS un
te au
pO to
ve ne
r w

Lw&Lw&
p Hp H
M 3M 3
en Gen 1
u 2u F
Ml M

sld xR
eS un
te au
pE to
xit op
en

Lw&Lw&
p Hp H
M 3M 2
en Gen E
u 3u E
Ml O

sld utli
eG ne
on m
tin en
ue u

Lw&Lw&
p Hp H
M 3M 2
en Gen E
u 4u F
Ml M

sl ol
eS Pr
to e
p m
ot
e

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u 5u 0
Ml M

sl ol
eS De
et m
Bp ot
e

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u 6u 1
Ml M

sl ol
eD M
is ov
Bp eu
p

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u 7u 2
Ml M

sl ol

eG M

lea ov

rAl ed

l ow

n

Lw&Lw&

p Hp H

M 3M 2

en Gen F

u 8u 3

Ml M

sld ol

eD Ex

is pa

All nd

Lw&Lw&

p Hp H

M 3M 2

en Gen F

u 9u 4

Ml M

sld ol

eB Co

re ntr

ak ae

poi t

nts

Lw&Lw&

p Hp H

M 3M 2

en Gen F

u Au 5

Ml M

sld ol

eB Sh

re ow
ws lev
er els
Lw&Lw&
p Hp H
M 3M 2
en Gen F
u Bu 6
Ml M
std ol
eO Ex
ut pa
pu nd
t sin
gl
e

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u Gu 7
Ml M
std ol
eV Co
ari ntr
abl ae
es tsi
ng
le

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u Du 8
Ml M

sld ol
eG Us
he eo
ek utli
Se ne
rip sty
t le

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u Eu 9
Ml M

sld ol
eN Se
ew ts
Su ho
b wl
ev
el
1

Lw&Lw&
p Hp H
M 3M 2
en Gen F
u Fu A
Ml M

sld ol
eN Se
ew ts
Fu ho
n wl
ev
el
2

Lw&Lw&

p Hp H

M 3M 2

en Den F

u Ou B

Ml M

slđ ol

eD Se

eb ts

ug hō

wl

ev

el

3

Lw&Lw&

p Hp H

M 3M 2

en Den F

u 1u G

Ml M

slđ ol

eR Se

ed ts

e hō

wl

ev

el

4

Lw&Lw&

p Hp H

M 3M 2

en Den F

u 2u D

Ml M

sS ol

he Se
wi ts
de he
wl
ev
el
5

Lw&Lw&
p Hp H
M 3M 2
en Den F
u 3u E
Ml M
sld el
el Se
m ts
pe he
rt wl
ev
el
6

Lw&Lw&
p Hp H
M 3M 2
en Den F
u 4u F
Ml M
sld el
eE Se
xp ts
ort he
wl
ev
el
7

Lw&Lw&

p Hp H

M 3M 3

en Den 0

u 5u 0

Ml M

sld ol

eS Se

eri ts

pt ho

Pr wl

ef ev

el

8

Lw&Lw&

p Hp H

M 3M 3

en Den 0

u 6u 1

Ml M

sld ol

eG Se

los ts

e ho

Wi wl

n ev

el

9

Lw&Lw&

p Hp H

M 3M 3

en Den 0

u 7u 2

Ml M

sld ol

eT Se
eg ts
gle hø
wl
ev
el
10

Lw&Lw&
p Hp H
M 3M 2
en Den E
u 8u D
Ml M
sld ol
eP Ø
re ut
vS sty
ub les
eq
ue
ne
e

Lw&Lw&
p Hp H
M 3M 3
en Den 2
u 9u Ø
Ml Mt
sld Ta
eN bk
ext ey
Su

b
Lw&Lw&
p Hp H

M 3M 3

en Den 2

u Au 4

Ml M

slđ eD

eT ef

og pa

gle th

bø sn

w

Lw&Lw&

p Hø H

M 3M 3

en Den 2

u Bu 5

Ml M

slđ vH

eG id

ed ep

e ict

Wi s

nd

ow

Lw&Lw&

p Hø H

M 3M 3

en Den 2

u Cu 6

Ml M

slđ vH

eG id

he e

ek m

All ar

ks

Lw&Lw&

p Hp H

M 3M 3

en Den 2

u Du 7

Ml M

slđ vH

eS id

av et

eL ab

so ra

ck

Lw&Lw&

p Hp H

M 3M 3

en Een 2

u Ou 8

Ml M

slđ vG

eS ol

ho gu

wi id

eo e

n

Lw&Lw&

p Hp H

M 3M 3

en Een 4

u 1u 8

Ml M

slđ vS

eR ho

un we

cu te

rs an

ub se
re
en
Lw&Lw&
p Hp H
M 3M 3
en Een 4
u 2u A
Ml M
slđ vF
eH ixē
elp dic
Ls on
ba
f

Lw&Lw&
p Hp H
M 3M 3
en Een 4
u 3u B
Ml M
slđ vV
eH ie
elp w!
W ev
p el

Lw&Lw&
p Hp H
M 3M 3
en Een 2
u 4u A
Ml Mt
slđ bG
eH ha
elp ng

Se e
Lw&Lw&
p Hp H
M 3M 3
en Een 2
u 5u B
Ml Mt
sld bT
eH ab
elp bo
Ab x
ou

t
Lw&Lw&
p Hp H
M 3M 3
en Een 2
u 6u E
Ml M
sld vV
el ert
ns rul
ert er
Ls

e
Lw&Lw&
p Hp H
M 9M 3
en 5en 3
u u 0
Ml M
Re vS
nu ho
m wn
no ot

tes es
Lw&Lw&
p Hp H
M 9M 3
en 6en 3
uE u 2
dit M
m vS
en ho
u wt
ab
sa
nd
ret
ur
ns

Lw&Lw&
p Hp H
M 9M 3
en 7en 3
u u 4
M M
eU vO
nd ut
e 2l
ay

Lw&Lw&
p Hp H
M 9M 3
en 9en 3
u u 5
M Mf
eG Im
ut pe
rd

ra
w
Lw&Lw&
p Hp H
M 9M 3
en Aen 3
u u 6
M Ml
eC M
op ae
y ro
en
dr
ee
or
d
Lw&Lw&
p Hp H
M 9M 3
en Ben 3
u u 7
M M
eP vL
ast ay
e 2o
ut
Lw&Lw&
p Hp H
M 9M 3
en Gen 3
u u 8
M M
eU vS
nd et
or pr

ed ef

e er

en

ee

s

Lw&Lw&

p Hp H

M 9M 3

en Den 3

u u 9

M Mr

eP Cr

ast ea

eli tel

nk as

s tfr

a

m

e

Lw&Lw&

p Hp H

M 9M 3

en Een 3

u u A

M MI

el Q

ns uic

va ke

ria nd

ble re

ee

rd

Lw&Lw&

p Hp H

M AM 3

en 0en 3

u u B

M Mr

eF Un

oo gr

tn ou

ot pfr

es a

m

es

Lw&Lw&

p Hp H

M AM 3

en 1en 3

u u G

M Ml

eN Re

ot v

es m

ar

kin

ge

ff

Lw&Lw&

p Hp H

M 1M 3

en 1en 3

u Au D

M Ml

eN Fi

ot el

es dh

20 id

e

Lw&Lw&

p Hp H
M AM 3
en 2en 4
u u 6
M M
eD eE
ef xtr
aul ae
ts to
n

Lw&Lw&
p Hp H
M AM 3
en 3en 4
u u 6
M M
eT div
yp Di
ee vis
ve io
r np
ro
pe
rti
es

Lw&Lw&
p Hp H
M AM 3
en 4en 4
u u D
M M
eN div
bD Ne
ef wd
aul ivi

ts sio
n
Lw&Lw&
p Hp H
M AM 3
en 5en 4
u u E
M M
el div
ns Co
ert m
da bi
te ne
div
isi
en
s

Lw&Lw&
p Hp H
M AM 3
en 6en 4
u u E
M M
eA div
dd De
m let
er ed
ge ivi
va sio
f n

Lw&Lw&
p Hp H
M AM 3
en 7en 5
u u 0

M M
eA div
ed Ex
de pa
ed nd
es div
ev isi
ar on

Lw&Lw&
p Hp H
M AM 3
en 8en 5
u u 6
Mx M
M div
dis Gr
wit ou
ch pd
ivi
sio
n

Lw&Lw&
p Hp H
M AM 3
en 9en 5
u u 1
M M
eS se
ecl eS
ev ee
act tio
ion np
re
pe
rti

es
Lw&Lw&
p Hp H
M AM 3
en Aen 5
u u 2
M M
eD se
ef cN
pa ew
ths se
cti
on

Lw&Lw&
p Hp H
M AM 3
en Ben 5
u u 3
M M
eF se
oo cG
to e
pts m
bi
ne
se
cti
on

s
Lw&Lw&
p Hp H
M AM 3
en Gen 5
u u 4
M M

eR se
ed eS
e el
ee
ts
ee
tio
n

Lw&Lw&

p Hp H

M AM 3

en Den 5

u u 5

M M

eD se

ee eE

op xp

tio an

ns ds

ee

tio

n

Lw&Lw&

p Hp H

M AM 3

en Een 5

u u 7

M Mt

eD ab

ef Se

pa ta

th bs

Lw&Lw&

p Hp H

M AM 3

en Fen 5

u u 8

M Mt

eD ab

efs Gl

ty ea

rta

bs

Lw&Lw&

p Hp H

M BM 3

en 0en 5

u u 9

M Mt

eD ab

ef Le

ba ftt

ek ab

Lw&Lw&

p Hp H

M BM 3

en 1en 5

u u A

M Mt

eF ab

oo Ri

tn gh

ot tta

ein b

se

rt

Lw&Lw&

p Hp H

M BM 3

en 2en 5

u u B
M Mt
el ab
ns Ge
ert nt
no ert
te ab

Lw&Lw&

p Hp H
M BM 3
en 3en 5

u u E
M Mt
eD ab
ele Nu
te m
eri
et
ab

Lw&Lw&

p Hp H
M BM 3
en 4en 6

u u 1
M Mt
el ab
nd De
ex fa
en ult
try ta
bs

Lw&Lw&

p Hp H
M BM 3
en 5en 6

u u 2

M Mt

eA ab

dd Re

no m

te ov

er

ul

er

Lw&Lw&

p Hp H

M BM 3

en 6en 6

u u 3

M Mt

e ab

M Hi

er de

ge rul

va er

r

Lw&Lw&

p Hp H

M BM 3

en 7en D

u u E

M Mt

eD ab

es Q

ev uic

ar kri

gh

t

Lw&Lw&

p Hp H

M BM 3
en 8en D
u u F
M Mt
eF ab
or Q
m uic
ate ke
he en
ck ter
ba

r
Lw&Lw&
p Hø H
M BM 3
en 9en 6
u u 6
M Mr
eP Se
ag tm
eu ar
p gi
nsi
nf
ob
ox

Lw&Lw&
p Hø H
M BM 3
en Aen 6
u u 7
M Mf
eP Te
ag gg
ed lei

ow nf

n ob

ox

Lw&Lw&

p Hp H

M BM 3

en Ben 6

u u 8

M Mf

eR Gr

evi ea

ns te

ert ve

ion rsi

on

Lw&Lw&

p Hp H

M BM 3

en Gen 8

u u 9

M Mf

eP Gr

rot ea

ect te

ed ve

rsi

on

Di

vis

ie

n

Lw&Lw&

p Hp H

M BM 3

en Den 7

u u ø
M M
eN vV
øh ie
yp wv
he er
n sio
n

Lw&Lw&
p Hø H
M CM 3
en 6en 8
u u 4
M M
eG afi
et dH
glø el
sn p
a
m
e

Lw&Lw&
p Hø H
M 1M 3
en 1en 8
u Eu 5
M M
e afi
M dN
ae øff
pø øu
th nd

Lw&Lw&
p Hø H
M 3M 3

en 4en 8

u 3u 6

M Mf

eT M

oe ail

en Ne

try xts

te

p

Lw&Lw&

p Hp H

M 3M 3

en 4en 8

u 4u 7

M Mf

eB M

ull ail

et Ed

itr

ou

te

Lw&Lw&

p Hp H

M 2M 3

en 3en 8

u 2u E

M M

e er

M ge

er Se

ge rip

da tN

tafi oti

le fy

Lw&Lw&

p Hp H
M 2M 3
en 3en 9
u 4uFD
M irs
e tp
M op
er up
ge id
ext
de
se

Lw&Lw&
p Hp H
M 2M 3
en 3en 5
u 6uRG
M en
e de
M rfo
er rm
ge at
va
rs
da
tafi
le

Lw&Lw&
p Hp H
M 2M 3
en 4en 5
u CuRD
M en
eP de
ast ral

es lfo
pe rm
cia at
l s
Lw&Lw&
p Hp H
M 2M 3
en 4en 7
u Du 9
M M
el eG
ns e
ert m
obj m
ect en
tte
ols
Ø
n

Lw&Lw&
p Hp H
M 2M 3
en 4en 7
u Eu A
M M
eG nD
ha el
ng et
eli en
nk et
e

Lw&Lw&
p Hp H
M 2M 3
en 4en 7

u Fu B
M M
eU nG
se los
wo en
rk ot
pa e
th

Lw&Lw&
p Hp H
M 2M 3
en 5en 7
u 3u G
M M
eS nN
ele ex
et tn
wo ot
rd e

Lw&Lw&
p Hp H
M 2M 3
en 5en 7
u 4u D
M M
eS nP
ele re
ets vn
en ot
te e
ne
e

Lw&Lw&
p Hp H
M 2M 3

en 5en 7

u 5u E

M M

eS eG

ele e

etp m

ar m

ag en

ra tto

ph ols

Lw&Lw&

p Hp H

M 2M 3

en 5en 7

u 6u F

M M

eS eH

ele ig

etfi hn

le ot

e

Lw&Lw&

p Hp H

M 2M 3

en 5en 8

u 7u 0

M M

eR nG

evi los

sio ea

n H

m

ed

e

Lw&Lw&

p Hp H
M 2M 3
en 5en 8
u 8u 1
Ml M
De n
ee O
on pe
tro na
l ll

Lw&Lw&
p Hp H
M 2M 3
en 5en 8
u Au 2
Mf M
Sh nD
ar el
ed et
ee ea
ee ll
ntr
ol

Lw&Lw&
p Hp H
M 2M 3
en 5en 8
u Bu 3
Ml M
De nS
ee he
on wi
tro niti
lE als
dit

or

pa

nel

Lw&Lw&

p Hp H

M 2M 3

en 5en 8

u Gu 4

Ml M

Do nN

ee u

on m

tro be

IE rn

xte ot

rn es

al

Lw&Lw&

p Hp H

M 2M 3

en 5en B

u Du 2

Mv Mf

Vi Q

ew uic

ve kp

rsi rin

on t

Ex

ter

nal

Lw&Lw&

p Hp H

M 2M 3

en 5en E

u Eu 8

Mf M

Do dD

ed eb

es ug

eE ge

xte r

rn

al

Lw&Lw&

p Hp H

M 2M 3

en 5en E

u Fu 9

Mf M

Mr dD

gla eb

bel ug

Do

e

Lw&Lw&

p Hp H

M 2M 3

en 6en E

u Qu A

Mf M

Mr dD

gle eb

tte ug

rs Pr

et es

up

Lw&Lw&

p Hp H

M 2M 3

en 6en E

u 1u B

Mf M

Mr dD

ge eb

nv ug

elo Di

pe vin

set fo

up

Lw&Lw&

p Hp H

M 2M 3

en 6en E

u 4u G

Mf M

Mr dD

gvi eb

ew ug

pri Go

nt nt

Lw&Lw&

p Hp H

M BM 3

en Een E

uN u D

wo M

bje dD

ets eb

ug

Go

t

Lw&Lw&

p Hp H

M BM 3

en Fen E

u u E

M M

oO dD

pe eb

nn ug

w Li

ne

Lw&Lw&

p Hp H

M GM 3

en Oen E

u u F

M M

oG dD

re eb

at ug

en Ve

w bj

Lw&Lw&

p Hp H

M GM 3

en 1en F

u u O

M M

oLi dD

stn eb

w ug

He

ap

Lw&Lw&

p Hp H

M GM 3

en 2en F

u u 1

M M

oLi dD

nk eb

sn ug

w M

ai

n

Lw&Lw&

p Hp H

M GM 3

en 3en F

u u 2

M M

oA dD

ttri eb

bn ug

w Pr

of

Lw&Lw&

p Hp H

M GM 3

en 8en F

uV u 3

ie M

w dD

m eb

en ug

u Vc

nt

Lw&Lw&

p Hp H

M GM 3

en 9en F

u u 4

Mv M

Fit dD
eb
ug
Di
vis
ie
n

Lw&Lw&
p Hp H
M CM 3
en Aen F
u u 5
Mv M
W dD
er eb
kin ug
g La

y
Lw&Lw&
p Hp H
M 3M 3
en 1en F
u Gu 6
Mv M

75 dD
eb
ug
Ve
rm
gr

Lw&Lw&
p Hp H
M 3M 3
en 1en F
u Du 7

Mv M

15 dD

0 eb

ug

Ve

ret

fl

Lw&Lw&

p Hp H

M GM 3

en Ben F

u u 8

Mv M

10 dD

0 eb

ug

Do

ela

y

Lw&Lw&

p Hp H

M GM 3

en Gen F

u u A

Mv M

20 dD

0 eb

ug

Pa

ra

Lw&Lw&

p Hp H

M GM 3

en Den F

u u B

Mv M

S dD

wit eb

chi ug

co Inf

nst e

at

e

Lw&Lw&

p Hp H

M 3M 3

en 0en F

u 3u G

Mv M

Pa dD

ge eb

se ug

rt At

e

m

Lw&Lw&

p Hp H

M 3M 3

en 0en F

u 4u D

Mv M

Ze dD

e eb

m ug

ep Wr

tio da

ns rg

Lw&Lw&

p Hp H

M 3M 3

en 0en F
u 6u E
Mv M
Go dD
m eb
m ug
en D
t wr
da
rg

Lw&Lw&
p Hp H
M 3M 3
en 0en F
u 7u F
Mv M
Sh dD
ow eb
he ug
ad Str
fo ar
ot g

Lw&Lw&
p Hp H
M 3M 4
en 0en 0
u 5u 0
Mv M
Gu dD
sto eb
m ug
Sl
ne

Lw&Lw&
p Hp H

M 3M 4
en 0en 0
u 8u 1
Mv M
Fa dD
cin eb
gp ug
ag Ph
es int
Lw&Lw&
p Hp H
M 3M 4
en 0en 0
u 9u 2
Mv M
M dD
ulti eb
ple ug
pa Gh
ge int
s
Lw&Lw&
p Hp H
M 3M 4
en 0en 0
u Au 3
Mv M
Pa dD
ge eb
m ug
ar M
gin sg
tra
ee
Lw&Lw&

p Hp H
M 3M 4
en 0en 0
u Bu 4
Mv M
Pa dD
ge eb
wi ug
dt Fo
h un
dr
y

Lw&Lw&
p Hp H
M 3M 4
en 2en 0
u 3u 5
Mv M
Sp dD
eci eb
alv ug
ie Co
w un
ts

Lw&Lw&
p Hp H
M 3M 4
en 7en 0
u 1u 6
Mv M
Gl dD
ea eb
rs ug
pe Pr
cia esi

lvi nf
ew e
Lw&Lw&
p Hp H
M 3M 4
en 6en 0
u 4u 7
Mv M
Sy dD
ne eb
hr ug
oni M
ze et
vie er
ws
Lw&Lw&
p Hp H
M 3M 4
en 6en 0
u Au 8
Mv M
Gl dD
ea eb
rhi ug
ghi St
igh ae
t k
Lw&Lw&
p Hp H
M GM 4
en Een 0
u u 9
Mv M
Pr dD
ef eb

er ug
en Ptr
ee hø
s ld
er
Lw&Lw&
p Hø H
M GM 4
en Fen 0
u u A
Mv M
Ta dD
br eb
ae ug
k M
e
m
er
y
Lw&Lw&
p Hø H
M DM 4
en 0 en 0
u u B
Mv M
Fu dD
net eb
ion ug
ba M
f e
m
er
yd
u
m

p
Lw&Lw&
p Hp H
M DM 4
en 1 en 0
u u G
Mv M
Gh dD
gsi eb
de ug
ba Co
r nt
en

t
Lw&Lw&
p Hp H
M DM 4
en 2 en 0
u u D
Mv M
La dD
yo eb
ut ug
Sil
ve

r
Lw&Lw&
p Hp H
M DM 4
en 3 en 0
u u E
Mv M
Dr dD
aff eb
ug

Ve

rn

ed

e

Lw&Lw&

p Hp H

M DM 4

en 4en 0

u u F

Mv M

Fa dD

ein eb

g ug

Ch

ee

kp

oi

nt

Lw&Lw&

p Hp H

M DM 4

en 6en 1

u u 0

Mv M

Fa dD

ee eb

ea ug

ne So

el ek

et

Lw&Lw&

p Hp H

M DM 4

en 7en 1

u u 1

Mv M

Sh dD

ow eb

pic ug

ts Sf

b

Lw&Lw&

p Hp H

M DM 4

en 8en 1

u u 2

Mv M

Sh dD

ow eb

m ug

ar De

ks m

an

dl

oa

d

Lw&Lw&

p Hp H

M DM 4

en 9en 1

u u 3

Mv M

Pr dD

ef eb

er ug

de Nu

fa m

ult be

s rhi

nt

Lw&Lw&

p Hp H

M DM 4

en Aen 1

u u 4

Mv M

Ou dD

flin eb

e ug

Di

al

og

Lw&Lw&

p Hp H

M DM 4

en Ben 1

u u 5

Mv M

Fl dD

oa eb

tin ug

gic Ba

on ck

Lw&Lw&

p Hp H

M DM 4

en Gen 1

u u 6

Mv M

Le dD

ftie eb

on ug

He

ad

tail

he

ld

er

Lw&Lw&

p Hp H

M DM 4

en Den 1

u u 7

Mv M

Ri dD

gh eb

tie ug

on No

tify

pe

rsi

st

en

t

Lw&Lw&

p Hp H

M DM 4

en Een 1

u u 8

Mv M

To dD

pic eb

on ug

O

bj

ee

th

of

de

f

Lw&Lw&

p Hp H

M DM 4

en Fen 1

u u 9

Mv M

Be dD

tto eb

mi ug

co En

n u

m

ob

jh

ea

d

Lw&Lw&

p Hp H

M EM 4

en 8en 1

u u A

Mv M

S dD

m eb

art ug

ico No

ns tify

Lw&Lw&

p Hp H

M EM 4

en 9en 1

u u B

Mv M

Fu dD

lls eb

er ug
ee Na
n m
ed
ob
jec
th
ol
de
t

Lw&Lw&
p Hø H
M EM 4
en Aen 1
u u G
Mv M
Fu dD
lls eb
er ug
ee Se
no eti
pti on
on
s

Lw&Lw&
p Hø H
M EM 4
en Gen 1
u u D
Mv M
Ed dD
itie eb
on ug
Fe
nt

m

gr

Lw&Lw&

p Hp H

M EM 4

en Den 1

u u E

Mf M

Pri dD

nt eb

en ug

vel Sa

op nit

e y

Lw&Lw&

p Hp H

M EM 4

en Fen 1

u u F

Mv M

Ne dD

xti eb

co ug

ns At

et o

m

W

at

ch

Lw&Lw&

p Hp H

M FM 4

en 0en 2

u u 0

Mv M

Sa dD

vei eb

eo ug

n At

e

m

Co

un

t

Lw&Lw&

p Hp H

M FM 4

en 1 en 2

u u 1

Mv M

Gr dD

ea eb

tei ug

eo Pi

n ee

eG

ou

nt

Lw&Lw&

p Hp H

M FM 4

en 2 en 2

u u 2

Mv M

Fu dD

lls eb

er ug

ee O

nst bj

art ee

up tSi
ze
s
Lw&Lw&
p Hp H
M FM 4
en 4en 2
u u 3
Mv M
Se dD
eti eb
on ug
ta Te
bs xts
tyl
e

Lw&Lw&
p Hp H
M FM 4
en Aen 4
uG u B
ha M
rm dD
en eb
u ug
En
d

Lw&Lw&
p Hp H
M 1M 4
en 1en 4
u Bu G
Me M
Te kP
xti ag

nf eu
ob p
ox
Lw&Lw&
p Hp H
M 2M 4
en 6en 4
u 2u D
Mc M
Te kP
xti ag
nf ed
ob ow
ox n
m
ou
se
Lw&Lw&
p Hp H
M 2M 4
en 6en 4
u 3u E
Mc M
Dr eD
aw oo
inf le
ob dr
ox ag
m dr
ou op
se
Lw&Lw&
p Hp H
M FM 5
en Ben 1

u u 4

Me M

Fe s

nt M

pl

M

af

gi

ns

Lw&Lw&

p Hp H

M FM 5

en Den 1

u u 5

Me M

Ne s

rm M

al pl

Pa

ge

Lw&Lw&

p Hp H

M FM 5

en Een 1

u u 6

Me M

Be s

ld M

pl

Fe

ot

er

Lw&Lw&

p Hp H

M FM 5

en Fen 1

u u 7

Me M

Ita! s

ie M

pl

He

ad

er

Lw&Lw&

p Hp H

M 1M 5

en 0en 1

u 0u 8

Me M

Un s

de M

rlin pl

e Li

ne

s

Lw&Lw&

p Hp H

M 1M 5

en 0en 7

u 1u 8

Me Mr

W Mf

or ITy

du pe

nd

er

Lw&Lw&

p Hp H

M 1M 5

en 0en 7

u 3u 9

Me Mr

Le Mf

ft tSi

ze

pø

s

Lw&Lw&

p Hp H

M 1M 5

en 0en 7

u 4u A

Me Mr

Ge Mf

nt tLi

er ne

s

Lw&Lw&

p Hp H

M 1M 5

en 0en 7

u 5u B

Me Mr

Ri Mf

gh tG

t ol

u

m

ns

ta

bs

Lw&Lw&

p Hp H

M 1M 5

en 0en D

u 6u G

Me Ml

Ju Se

stif tm

y ae

rof

ile

Lw&Lw&

p Hp H

M 1M 5

en 0en D

u 7u D

Me Ml

Ali Co

gn m

m pil

en e

t m

ae

re

Lw&Lw&

p Hp H

M 1M 5

en 0en D

u 8u E

Me Ml

Ga M

ps ae

ror

un

Lw&Lw&

p Hp H

M FM 5

en 7en D

u u F
Me Ml
In M
de ae
nt re
all ap
pe
nd

Lw&Lw&
p Hp H
M FM 5
en 8en E

u u 0
Me Ml
In Pr
de ot
ntfi ot
rst yp
e

Lw&Lw&
p Hp H
M FM 5
en 9en E

u u 1
Me M
In eS
de m
ntr art
est co
rre
et

Lw&Lw&
p Hp H
M FM 5
en 6en E

u u 2

Me Mr

Ex Ch

tra art

et

Lw&Lw&

p Hp H

M EM 5

en Een E

u u 3

Me Mr

M Ch

ae art

rof to

ast dr

for aw

m

at

Lw&Lw&

p Hp H

M FM 6

en 3en 4

u u 0

Me M

Fo m

ntr Ca

ev se

ert ad

es

Lw&Lw&

p Hp H

M FM 6

en Gen 4

u uS0

Me ub

En m
lar en
ge ul
tex ns
t ert
Lw&Lw&
p Hp H
M FM 6
en 5en 4
u u 1
Mc M
Re m
du Fi
eet el
ext ds
Lw&Lw&
p Hp H
M 1M 6
en 0en 4
u 2uS 2
Mc ub
In m
de en
nt u
M
ar
kt
ex
ta
s
Lw&Lw&
p Hp H
M 1M 6
en 1en 4
u GuS 3

Me ub

Ou m

td en

en uA

t lig

n

m

en

t

Lw&Lw&

p Hp H

M 1M 6

en 1en 4

u Du 4

Me M

Bu m

Het Ca

pa ps

nel

Lw&Lw&

p Hp H

M 3M 6

en Ben 4

u 6u 7

Me M

Ne m

xtf M

ae ae

en re

a s

m

e

Lw&Lw&

p Hp H

M 3M 6

en Ben 4

u 7uS9

Mc ub

Ne m

xta en

lig ul

n ns

m ert

en s

t m

art

su

m

Lw&Lw&

p Hp H

M 3M 6

en Ben 4

u 8u A

Mc M

Ne M

xta ail

ttri m

bu e

te m

e

Lw&Lw&

p Hp H

M 3M 6

en Ben 4

u 9u B

Mc M

Ne Pri

xtb ntt

ull ofil

et e

Lw&Lw&

p Hp H

M 3M 6

en Ben 4

u AuSG

Mc ub

Ne m

xtn en

u uU

m se

be rs

rin et

g up

Lw&Lw&

p Hp H

M 3M 6

en Ben 4

u BuSD

Mc ub

Ne m

xts en

tyl uS

e cri

pt

s

Lw&Lw&

p Hp H

M 1M 6

en 0en 4

u Eu E

Mc M

Do m

ubl Sh

eu ow

nd ep

er tie
ns
Lw&Lw&
p Hp H
M 1M 6
en 0en 4
u FuSF
Me ub
Su m
pe en
rsc u
rip Q
t uie
kd
e
m
e

Lw&Lw&
p Hp H
M 1M 6
en 1en 5
u 0uS0
Me ub
Su m
bs en
eri u
pt Ot
he
rh
el
ps
tuf
f

Lw&Lw&
p Hp H

M 1M 6
en 1en 5
u 1u 1
Mc M
All m
ca Ot
ps he
rfr
a
m
e

Lw&Lw&
p Hp H
M 1M 6
en 1en 5
u 2u 2
Mf M
En m
vel Ot
op he
ep rta
rin bl
t e

Lw&Lw&
p Hp H
M 1M 6
en 1en 5
u FuS3
Mc ub
All m
ow en
er u
Ot
he
rte

xts

tuf

f

Lw&Lw&

p Hp H

M 1M 6

en 2en 5

u 0uS4

Mc ub

lnit m

ca en

ps uP

rie

rit

y

Lw&Lw&

p Hp H

M 1M 6

en 2en 5

u 1u 5

Mc M

S m

m Ot

all he

ca rst

ps uff

Lw&Lw&

p Hp H

M 1M 6

en 1en 5

u 6uS6

Mc ub

Ju m

stf en

on uS

t el
ee
t
Lw&Lw&
p Hp H
M 1M 6
en 1en 5
u 7uS7
Me ub
Ju m
stp en
oin u
tsi Ot
ze he
rd
ee
pa
rt
Lw&Lw&
p Hp H
M 1M 6
en 1en 5
u 8uS8
Me ub
Str m
ike en
thr uA
u ttri
bu
te
s
Lw&Lw&
p Hp H
M 1M 6
en 2en 5

u 3uS9

Me ub

Fa m

stf en

or u

m O

at utli

ne

Lw&Lw&

p Hp H

M 1M 6

en 2en 5

u 6uSA

Me ub

Hi m

ghl en

igh uG

t oll

ap

se

Lw&Lw&

p Hp H

M 1M 6

en 2en 5

uS CuSB

tyl ub

e m

m en

en uE

u dit

Lw&Lw&

p Hp H

M 1M 6

en 2en 5

u DuSG

Ms ub

Gr m

ea en

te uD

eb

ug

Lw&Lw&

p Hp H

M 1M 6

en 2en 5

u FuSD

Ms ub

St m

yle en

m uD

an ee

ag pr

e ep

s

Lw&Lw&

p Hp H

M 1M 6

en 3en 5

u OuSE

Ms ub

Gh m

gst en

yle uZ

ee

mt

e

Lw&Lw&

p Hp H

M 1M 6

en 3en 5

u 1uSF

Ms ub

Sa m

ve en

sty uS

le he

wh

id

e

Lw&Lw&

p Hp H

M 1M 6

en 3en 6

u 2uS0

Ms ub

Se m

lec en

tst u

yle Ot

he

rvi

ew

st

uff

Lw&Lw&

p Hp H

M 1M 6

en 4en 6

u 0uS1

Ms ub

Ou m

tlin en

est uN

yle a

m

ed

sty

les

Lw&Lw&

p Hp H

M 1M 6

en 4en 6

u 1uS2

Ms ub

St m

yle en

m uD

an el

ag et

ein e

it

Lw&Lw&

p Hp H

M 1M 6

en 4en 6

u 2uS3

Ms ub

St m

yle en

m uF

an ile

ag

ea

eti

en

Lw&Lw&

p Hp H

M 1M 6

en 4en 6

u 3uS4

Ms ub
Gh m
an en
ge uV
sty ie
len w
a
m
e

Lw&Lw&
p Hp H
M 1M 6
en 4en 6
u 4uS5

Ms ub
De m
fin en
est uT
yle ea
m
m
ail

Lw&Lw&
p Hp H
M 1M 6
en 4en 6
u 5uS6

Ms ub
M m
ed en
er uE
ea dit
tes m
tyl ed
e e

m
en
u
Lw&Lw&
p Hp H
M 1M 6
en 4en 6
u 6uS7
Ms ub
St m
yle en
m uD
an eb
ag ug
efi m
nis od
h e
m
en
u
Lw&Lw&
p Hp H
M 1M 6
en 5en 6
u 0uS8
Ms ub
Re m
de en
fin uH
est el
yle p
Lw&Lw&
p Hp H
M 1M 6
en 4en 7

u 8uG2

Ms re

M at

ar e

gin m

s en

u

Lw&Lw&

p Hp H

M 1M 6

en 4en 7

u 9uPA

Ms ar

Pa all

ge els

set tyl

tin e

gs m

en

u

Lw&Lw&

p Hp H

M 1M 6

en 4en 7

u AuED

Ms nv

Ap Ad

ply dr

es

s

Lw&Lw&

p Hp H

M 1M 6

en 4en 8

u BuE1

Ms nv

St Sa

yle ve

mr

u1

Lw&Lw&

p Hp H

M 1M 6

en 4en 8

u CuE2

Ms nv

St Pri

yle nt

mr

u2

Lw&Lw&

p Hp H

M 1M 6

en 4en 8

u Du 6

Ms Mf

St Gr

yle ea

mr tet

u3 ab

div

isi

on

Lw&Lw&

p Hp H

M 1M 6

en 4en 8

u Eu 3

Ms Mf

St Gr

yle ea
mr te
u4 div
isi
on

Lw&Lw&
p Hp H
M 1M 6
en 4en 8
u Fu 4
Mr M
M eT
an oe
ual as
ta sis
ble t

Lw&Lw&
p Hp H
M 1M 6
en 8en 8
u 7u 5
Mr M
Fr pN
a ew
m se
ea cti
ne on

he
rin
fo
be
x

Lw&Lw&
p Hp H
M 1M 6

en 8en 8

u 8u 7

Mr Mt

Fr In

a se

m rte

eli lic

nei kh

nf er

ob e

ox

Lw&Lw&

p Hp H

M 1M 6

en 8en 8

u 9u 8

Mr M

Fr eT

a ex

m tlin

eb ep

ac an

kin el

fo

bo

x

Lw&Lw&

p Hp H

M 1M 6

en 8en 8

u Au 9

Mr M

Fr vO

a th

m er

eg vie

fxi ws

nf tuf

ob f

ox

Lw&Lw&

p Hp H

M 1M 6

en 8en 8

u Bu A

Mr M

Fr oA

a pp

m lys

et tyl

op e

ali

gn

Lw&Lw&

p Hp H

M 1M 6

en 8en 8

u Gu B

Mr M

Fr sF

a ra

m m

ee ee

en re

ter at

ali e

gn

Lw&Lw&

p Hp H

M 1M 6

en 8en 8
u Du G
Mr M
Fr sF
a ra
m m
eb er
ott ed
e efi
m ne
ali sty
gn le
Lw&Lw&
p Hp H
M 1M 6
en 8en 8
u Fu D
Mr M
De sT
let ab
efr lec
a re
m at
e e
Lw&Lw&
p Hp H
M 1M 6
en 5en 8
uPEu E
ag M
e sT
m ab
en ler
u ed
efi

ne

sty

le

Lw&Lw&

p Hp H

M 1M 6

en 5en 8

u Fu F

M M

pG sT

re ab

at lec

ela ell

ye er

ut ea

te

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u Ou O

M M

p sT

M ab

odi lec

fyl ell

ay re

ou de

t fin

es

tyl

e

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u 3u 1

M M

pT vS

ab he

s w

mi

ss

pe

he

d

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u 4u 2

M Mt

pP De

ag tre

en wa

u ee

m

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u 6u 3

M Mt

pP Ct

ag rlr

eb et

re ae

ak e

Lw&Lw&

p Hp H

M 1M 6
en 6en 9
u 7u 4
M M
pLi sQ
ne uic
nu kr
m ed
be efi
r ne
sty
le

Lw&Lw&
p Hp H
M 1M 6
en 6en 9
u 8u 5
M Mt
pH In
ea se
df rtr
oo ow
t ae
e

Lw&Lw&
p Hp H
M 1M 6
en 6en 9
u 9u 6
M Mr
p Ol
M ee
edl on
ay ve
ou rt

tini

t

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u Au 7

M Mr

p Ol

M ee

odl on

ay ve

pa rta

ge nd

set ae

tin tiv

gs at

e

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u Bu 8

M Mr

p Fr

M a

odl m

ay el

rig eft

htf

ae

ep

ag

e

Lw&Lw&

p Hp H
M 1M 6
en 6en 9
u Gu 9
M Mr
p Fr
M a
odl m
ay ee
rig en
ht ter
he
ad
erf
ae
ep
ag
e

Lw&Lw&
p Hp H
M 1M 6
en 6en 9
u Du A
M Mr
p Fr
M a
odl m
ay eri
rig gh
htf t
ee
ter
fac
ep
ag

e

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u Eu B

M Mr

p Fr

M a

odl m

ay eh

rig or

htli zs

ne pa

s n

Lw&Lw&

p Hp H

M 1M 6

en 6en 9

u Fu G

M Mr

p Fr

M a

odl m

ayl ev

eft erf

fac sp

ep an

ag

e

Lw&Lw&

p Hp H

M 1M 6

en 7en 9

u Qu D

M Mf
p Ne
M wa
odl ny
ayl ta
eff b
he
ad
eff
ae
ep
ag
e
Lw&Lw&
p Hp H
M 1M 6
en 7en 9
u 1u E
M Mf
p Θ
M d
odl m
ayl ae
eff pe
fo n
ot
eff
ae
ep
ag
e
Lw&Lw&
p Hp H
M 1M 6
en 7en 9

u 2u F

M Mf

p Ø

M d

odl m

ayl ai

effl ns

ine ert

s

Lw&Lw&

p Hp H

M 1M 6

en 7en A

u 5u Ø

M Mf

pD Ø

ell d

ay m

ou ae

t on

ve

ft

Lw&Lw&

p Hp H

M 1M 6

en 7en A

u 6u 1

M Mf

pR Ø

ev d

ert m

lay as

ou av

t ea

s

Lw&Lw&

p Hp H

M 1M 6

en 7en A

u 7u 2

M Mf

pl O

ns! d

ay m

ou ae

t xt

er

na

liz

e

Lw&Lw&

p Hp H

M 1M 6

en 7en A

u 8u 3

M Mf

p O

M d

od! m

ay ae

ou xp

tfin ort

ish

Lw&Lw&

p Hp H

M 1M 6

en 7en A

u 9u 4

M Mr

pH O!

ea ev

de er

rfo b

ot

er

ge

to

Lw&Lw&

p Hp H

M 1M 7

en 7en 0

u Au 7

M Mr

p Ol

M ev

aci er

nsf b

ay m

ou ax

t

Lw&Lw&

p Hp H

M 1M 7

en 7en 0

u Bu 8

M Mr

pP Ol

ar ed

all ivt

el ab

ve

rb

Lw&Lw&

p Hp H

M 1M 7

en 7 en 6
u Gu B
M Mr
pP Ol
col ed
inf ivt
ob ab
ox ve
rb
m
ax

Lw&Lw&
p Hp H
M 1M 7
en 7 en 6
u Du G
M Mt
pP Se
ag lee
ep te
re ell
pe
rti
es

Lw&Lw&
p Hp H
M 1M 7
en 7 en 6
u Eu D
M Mt
pG Pe
re o/s
at el
e ee
te

ell
Lw&Lw&
p Hp H
M 1M 7
en 7en 6
u Fu E
M M
pN pC
ew ol
pa u
ge m
sty nb
le re
ak

Lw&Lw&
p Hp H
M 1M 7
en 8en 6
u Ou F
M M
pH eV
ea ie
de wt
rpr ab
op le
ert gri
ies d

Lw&Lw&
p Hp H
M 1M 7
en 8en 7
u 1u 0
M M
pF eV
oo ie

ter wt

pr ab

op le

ert he

ies ad

in

g

Lw&Lw&

p Hp H

M 1M 7

en 8en 7

u 2u 1

M M

pP eV

ag ie

ep wp

re col

pe gu

rti id

es es

m

ou

se

Lw&Lw&

p Hp H

M 1M 7

en 8en 7

u 3u 2

M M

pF eV

oo ie

ter w

pr m

op ar

ert gi

ies ng
m ui
ou de
se s
Lw&Lw&
p Hp H
M 1M 7
en 8en 7
u 4u 3
M M
pH eV
ea ie
de wp
rpr ag
op eg
ert au
ies ge
m
ou
se
Lw&Lw&
p Hp H
M 1M 7
en 8en 7
u 5u 4
M M
pP eV
col ie
inf ws
ob he
ox wa
m llm
ou ar
se ks
Lw&Lw&

p Hp H
M 1M 7
en 9en 7
uF 0u 5
ra M
m eV
e ie
m wh
en id
u ea
llm
ar
ks

Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 2u 6
Mr Mt
Fr Ta
a bl
m el
et eft
ext

Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 3u 7
Mr Mt
Se Ta
ale bl
ee
en
ter

Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 4u 8
Mr Mt
Gr Ta
ou bl
pfr eri
a gh
m t
es

Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 5u 9
Mr Mt
Fr Ta
a bl
m eh
ein or
fo zs
bo pa
x n

Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 7u A
Mr Mt
Bri G
ng et
tof oi
re nf

nt ob
ex
Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 8u B
Mr M
Se hT
nd ou
to r
ba
ek
Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u 9u D
Mr Mf
Ad Gr
d ea
te
ol
ed
ivi
sio
n
Lw&Lw&
p Hp H
M 1M 7
en 9en 7
u Au E
Mr Mf
Se To
nd gg

ba lee

ek lie

en kh

e er

ep

re

m

pt

s

Lw&Lw&

p Hp H

M 1M 7

en 9en 7

u Bu F

Mr M

Dr eT

aw eg

ing gl

ice es

n kip

bu

let

Lw&Lw&

p Hp H

M 1M 7

en 9en 8

u Gu 0

Mr Ml

Ch Sk

art ip

ing mi

ice ss

n pe

lle

dw

or
d
Lw&Lw&
p Hp H
M 1M 7
en 9en 8
u Du 1
Mr Ml
Gr Sk
ea ip
tef all
ra mi
m ss
e pe
lle
dw
or
d
Lw&Lw&
p Hp H
M 1M 7
en 9en 8
u Eu 2
Mr Ml
Ad Ad
dfr dt
a ou
m se
ew rdi
ith eti
dlg on
ar
y
Lw&Lw&
p Hp H

M 1M 7
en 9en 8
u Fu 3
Mr Ml
Fr To
a gg
m le
ed bu
ef bb
aul le
ts he
lp

Lw&Lw&
p Hp H
M 1M 7
en Aen 8
u Ou 4
Mr Mf
Fr Pl
a ai
m nn
eli ew
ne
s

Lw&Lw&
p Hp H
M 1M 7
en Aen 8
u 1u 5
Mr M
Fr vT
a eg
m gl
ety es
pe he

wa

ne

he

rs

Lw&Lw&

p Hp H

M 1M 7

en Aen 8

u 2u 6

Mr M

Fr vN

a ex

m tn

eb ew

or es

de tv

rs er

sio

n

Lw&Lw&

p Hp H

M 1M 7

en Aen 8

u 3u 7

Mr M

Fr v

a M

m os

ee tre

olu ee

m nt

ns ve

rsi

on

Lw&Lw&

p Hp H
M 1M 7
en Aen 8
u 4u 8
Mr M
M vO
od ld
fra es
m tv
er er
efl sio
ow n

Lw&Lw&
p Hp H
M 1M 7
en Aen 8
u 5u 9
Mr M
M eJ
od us
fra tify
m all
ein
it

Lw&Lw&
p Hp H
M 1M 7
en 8en 8
u 6u A
Mr M
Bri eG
ng los
fre eg
nt ra
on ph

e ico
le
Lw&Lw&
p Hp H
M 1M 7
en 8en 8
u Eu B
Mr Mf
Se Fr
lec a
tfr m
a er
m ev
e ert
Lw&Lw&
p Hp H
M 1M 7
en Aen 8
u 7u G
Mr Mt
Eq Ge
ua llr
tio ev
nsi ert
eo
n
Lw&Lw&
p Hp H
M 1M 7
en Aen 8
u 8u D
Mr Mf
lm O
ag pe
ep nt

re ab
ce div
ssi isi
ng on
Lw&Lw&
p Hp H
M 2M 7
en 6en 8
u Cu E
Mr Mf
Fr Ne
a wt
m ab
ein div
fo isi
bo on
xm
ou
se
Lw&Lw&
p Hp H
M 2M 7
en 6en 8
u Du F
Mr Mt
Fr In
a se
m rte
eli lie
nei kh
nf er
ob eli
ox nk
m
ou

se
Lw&Lw&
p Hp H
M 2M 7
en 6en 9
u Eu 0
Mr Mt
Fr Ed
a itel
m ick
eb he
ae rel
kin ink
fo
bo
xm
ou
se
Lw&Lw&
p Hp H
M 2M 7
en 6en 9
u Fu D
Mr M
Fr se
a eD
m el
eg et
fxi es
nf ee
ob tio
ox n
m
ou
se

Lw&Lw&

p Hp H

M 2M 7

en 7en 9

u Ou E

Mr Mf

Fr Ex

a ter

m na

ea fil

ne ee

ho xit

rin

fo

be

xm

ou

se

Lw&Lw&

p Hp H

M 3M 7

en 8en 9

u Fu F

Mr Mf

Ex Ex

pe ter

rt na

fil

es

av

ed

ivi

sio

n

Lw&Lw&

p Hp H
M 3M 7
en 9en A
u Gu 0
Mr Mf
Ex Ft
pe pø
rt pe
m n
ax

Lw&Lw&
p Hp H
M 1M 7
en 5en A
u Du 1
Mr Mf
W Ft
pp pi
ac ns
k ert

Lw&Lw&
p Hp H
M 1M 7
en Aen A
u 9u 2
Mr Mf
Afi Ft
d pe
on
ve
rt

Lw&Lw&
p Hp H
M 1M 7
en Gen A

u 1u 3

Mr Mf

Afi Ft

d ps

m av

ax ea

s

Lw&Lw&

p Hp H

M 1M 7

en Gen A

uT 2u 4

ool Mf

m Ft

en pe

u xt

er

na

liz

e

Lw&Lw&

p Hp H

M 1M 7

en Gen A

u 3uS5

Ml ub

Se m

ar en

ch ul

nt

er

ne

t

Lw&Lw&

p Hp H

M 1M 7
en Gen A
u 4u 6
Ml Mf
Ge Ft
to pe
pti
on
s

Lw&Lw&
p Hp H
M 1M 7
en Gen A
u 5u 7
Ml Mf
Sp O
ell d
op m
ts as
he
wa
ttri
bu
te
s

Lw&Lw&
p Hp H
M 1M 7
en Gen A
u 6u 8
Ml Mr
Us Ru
er by
die
t

Lw&Lw&
p Hp H
M 1M 7
en Gen A
u 7u 9
Ml Mf
Th Ft
es pe
au pe
ru nd
s ivi
sio
n

Lw&Lw&
p Hp H
M 1M 7
en Gen A
u 8u A
Ml Mf
So Ft
rt pe
pe
nu
tl

Lw&Lw&
p Hp H
M 1M 7
en Gen A
u 9ul B
Ml bS
Ta he
ble wf
s re
m
m

en
u
Lw&Lw&
p Hp H
M 1M 7
en Gen A
u Au G
Ml M
Go oH
tol id
ast ei
nf
ob
ex
Lw&Lw&
p Hp H
M 1M 7
en Gen A
u Bu D
Ml Mf
Sp W
ell etc
op e
ts2 m
e
Lw&Lw&
p Hp H
M 1M 7
en Gen A
u Du E
Ml W
Re p5
vm Ex
ar pe
ke rt

pti
on
s
Lw&Lw&
p Hp H
M 1M 7
en Gen A
u Eu F
Ml W
To p6
e Ex
pe
ft
Lw&Lw&
p Hp H
M 1M 7
en Gen B
u Fu 0
Ml W
As p
sig W
n etc
m 0
ae m
ro e
Lw&Lw&
p Hp H
M 1M 7
en Den B
u 0u 1
Ml M
M sQ
ae uie
ro kr
s ed

efi

ne

fra

m

es

tyl

e

Lw&Lw&

p Hp H

M 1M 7

en Aen B

u 6u 2

Ml M

M sQ

ae uie

re kr

ae ed

cel efi

ne

ta

bl

es

tyl

e

Lw&Lw&

p Hp H

M 1M 7

en Den B

u 1u 3

Ml M

Ge sQ

ne uie

rat kr

et ed

oe efi

ne

cel

lst

yle

Lw&Lw&

p Hp H

M 1M 7

en 1en B

u 9u 4

MH M

nd sQ

ex uic

ne kr

xtfi ed

le efi

ne

te

xts

tyl

e

Lw&Lw&

p Hp H

M 1M 7

en Den B

u 2u 5

Ml M

Bo el

ok ns

m eft

ar sy

k m

bø

l

Lw&Lw&

p Hp H

M 1M 7
en Den B
u 3u 6
Ml M
Gl e
os W
sa of
ry de
ou
nt

Lw&Lw&
p Hp H
M 1M 7
en Den B
u 4u 7
Ml M
Gl eD
os re
set pe
ap
s

Lw&Lw&
p Hp H
M 1M 7
en Den B
u 5u 8
Ml Mf
M O
ae d
pa m
us ai
e m
pe
rtp
iet

ur
e
Lw&Lw&
p Hp H
M 1M 7
en Den B
u 6u 9
Ml Mr
Ge Ol
ne eo
rat bj
ein ee
de tin
x fo
bo
x

Lw&Lw&
p Hp H
M 1M 7
en Den B
u 7u A
Ml M
Sp el
ell nt
ad er
ddi ne
et tto
ols
on

Lw&Lw&
p Hp H
M 1M 7
en Den B
u 8u B
Ml M

Sp el
ell nt
ski er
pal ne
l tto
ols

Lw&Lw&
p Hp H
M 1M 7
en Den B
u Au G
Ml M
Sp hL
ellr ot
epf us
ae he
e m
ep
ag
e

Lw&Lw&
p Hp H
M 1M 7
en Den B
u Bu D
Ml M
Sp hL
ell ot
us
su
pp
ort

Lw&Lw&
p Hp H
M 1M 7

en Den B

u Gu E

Ml M

Sp hL

elli ot

nit us

ftø

sit

e

Lw&Lw&

p Hp H

M 1M 7

en Den B

u Du F

Ml Mr

Sp Li

ell nk

ch fra

ee m

k es

Lw&Lw&

p Hp H

M 1M 7

en Den C

u Eu 0

Ml Mr

Sp Un

ell lin

ea kfr

ne a

el m

es

Lw&Lw&

p Hp H

M 1M 7

en Den G

u Fu 1

Ml Mf

Sp Mr

ell gw

ski etc

p e

m

e

Lw&Lw&

p Hp H

M 1M 7

en Een G

u Ou 2

Ml Mf

Re Ht

pla ml

ce op

tie

ns

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 1u 3

Ml M

Gh eU

gla pd

ng at

ei

nd

ex

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 2u 4

Ml M

Re eU

nu pd

ms at

eq et

oe

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 5u 5

Ml M

Sr vD

att esi

rs gn

m

ed

e

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 6u 6

Ml Mf

Re No

pe se

pts pe

n

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 7u 7

Ml Mf

Sr No

op ss

ts av

ea

s

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 8u 8

Ml Mf

Se M

ar ail

eh Se

ent nd

y m

es

sa

ge

Lw&Lw&

p Hp H

M 1M 7

en Een G

u 9u 9

Ml M

M cl

ae ns

pla ert

y de

fn

u

m

be

f

Lw&Lw&

p Hp H

M 1M 7

en Een G

u Au A

Ml M

M el

ae ns

as ert

sig de

n fb

ull

et

Lw&Lw&

p Hp H

M 1M 7

en Een G

u Bu B

Ml M

M eT

ae og

dd gl

e es

m

art

sel

ee

t

Lw&Lw&

p Hp H

M 1M 7

en Een G

u Gu G

Ml M

M eG

act re
yp at
e ee
bj
ee
tvi
ew
er

Lw&Lw&

p Hp H
M 1M 7
en Een G
u Du D
Ml M
Ga eG
ll re

at

ee

bj

ee

te

e

m

m

en

t

Lw&Lw&

p Hp H
M 1M 7
en Een G
u Eu E
Ml M
Sa eG
m re
my at

z ee
bj
ee
tdr
aw

Lw&Lw&

p Hp H

M 1M 7

en Een G

u Fu F

Ml M

Te eG

ee re

pts at

ee

bj

ee

tse

he

du

le

Lw&Lw&

p Hp H

M 1M 7

en Fen D

u Ou O

Ml M

Srf eG

ind re

m at

ate ee

h bj

ee

ts

pr

ea

ds

he

et

Lw&Lw&

p Hp H

M 1M 7

en Fen D

u 1u 1

Ml M

Srr eG

epI re

ae at

et ee

he bj

nfi ee

nd te

ha

rt

Lw&Lw&

p Hp H

M 1M 7

en Fen D

u 2u 2

Ml M

Sr hl

ea nt

ne er

el ne

ts

ea

re

h

Lw&Lw&

p Hp H

M 1M 7
en Fen D
u 3u 3
Ml M
Sri hL
nit ot
us
s
m
art
sui
te

Lw&Lw&
p Hp H
M 1M 7
en Fen D
u 4u 4
Ml M
De eG
ee lie
e kh
m er
pa ek
re ey
wo
rd

Lw&Lw&
p Hp H
M 1M 7
en Fen D
u 5u 5
Ml Mf
M Pu
ae bli
re sh

edi we
t b
Lw&Lw&
p Hp H
M 1M 7
en Fen D
u 6u 6
Ml Ml
M Gr
ae os
ro sr
pla ef
y er
en
ee

Lw&Lw&
p Hp H
M 1M 7
en Fen D
u 7u 7
Ml Mt
M ab
ae Hi
ror de
ee ve
or rtr
d ul
er

Lw&Lw&
p Hp H
M 1M 7
en Fen D
u 8u 8
Ml M
M pU

ae pd

re at

op eh

tie ea

ns de

rs

Lw&Lw&

p Hp H

M 1M 7

en Fen D

u 9u 9

Ml M

Qu pU

ick pd

re at

e ef

oo

ter

s

Lw&Lw&

p Hp H

M 1M B

en Fen B

u Au 8

Ml Mr

Qu Ru

ick by

pla Ab

y ov

e

Lw&Lw&

p Hp H

M 1M B

en Fen B

u Bu 9

Ml Mr
Gr Ru
a by
m Be
m to
ar w

Lw&Lw&
p Hp H
M 1M B
en Fen B
u Gu A

Ml Mr
Gr Ru
a by
m De
m let
ar e
op
ts

Lw&Lw&
p Hp H
M 1M 2
en Fen 3
u Du 2

Ml M 8
Gr x
a Mi
m n
m m
ar ae
sta re
ft m
en
u

Lw&Lw&

p Hp H
M 1M 2
en Fen 3
u Eu 8
Ml M B
Gr x
a M
m ax
m m
ari ae
nit ro
m
en
u

Lw&Lw&
p Hp H
M 1M 2
en Fen 7
u Fu 1
Ml Mr 0
M Afi
od d
er m
ea en
ter u
ule

Lw&Lw&
p Hp H
M 2M 3
en 0en E
u 0u 8
Ml Mr 0
Gr Afi
a d
m m

m en
ar u
ne m
xt ax
Lw&Lw&
p Hp H
M 2M 3
en 0en E
u 1u 8
Ml M 1
Gr aN
a ot
m es
m flo
ar w
ca
ne
el
Lw&Lw&
p Hp H
M 2M 3
en 0en F
u 2u 4
Ml M 8
Gr aN
a ot
m es
m flo
ar w
su m
gt ax
ext
Lw&Lw&
p Hp H
M 2M 0

en 0en

u 3uF

Ml es

Gr Fe

a xt

m m

m en

ar u

op

tie

nst

ext

Lw&Lw&

p Hp H

M 2M 1

en 0en

u 4uF

Ml es

Gr Fr

a a

m m

m e

ar m

ch en

an u

ge

Lw&Lw&

p Hp H

M 2M 2

en 0en

u 5uF

Ml es

Gr Fr

a a

m m

m et

ar ex

sta tm

ts en

u

Lw&Lw&

p Hp H

M 2M 3

en 0en

u 6uF

MH es

ee Ta

np bl

at et

h ex

tm

en

u

Lw&Lw&

p Hp H

M 2M 4

en 0en

u 7uF

MH es

Cy Fr

ele a

ke m

ys eg

et ra

up ph

ie

m

en

u

Lw&Lw&

p Hp H
M 2M 5
en 0en
u 8uF
Ml es
S Po
m we
art rfi
fill el
d
m
en
u

Lw&Lw&
p Hp H
M 2M 6
en 0en
u 9uF
Ml es
Ne Pa
ws ral
pel lel
l col
te
xt
m
en
u

Lw&Lw&
p Hp H
M 2M 7
en 0en
u AuF
Ml es
Re Di

vis vis

ion io

ba n

r m

en

u

Lw&Lw&

p Hp H

M 2M 8

en 0en

u BuF

Ml es

Te Se

a eti

en

m

en

u

Lw&Lw&

p Hp H

M 4M 9

en en

u uF

Wi es

nd Ta

ow bl

m e

en m

u en

u

Lw&Lw&

p Hp H

M 2M A

en 0en

u DuF

M es
w Pa
Ne ral
w tel
wi col
nd m
ow en
u

Lw&Lw&
p Hp H
M 2M B
en 0en
u EuF

M es
wT Ta
ile bl
wi eg
nd ra
ow ph
ie
m
en
u

Lw&Lw&
p Hp H
M 2M G
en 0en
u EuF

M es
w Pa
Ga ral
se tel
ad col
ew gr
ind ap

ow hie

m

en

u

Lw&Lw&

p Hp H

M 2M D

en 1en

u 0uF

M es

w He

St ad

art er

offi m

les en

u

Lw&Lw&

p Hp H

M 2M E

en 4en

u 6uF

M es

wT Fo

ile ot

wi er

nd m

ow en

ho u

rZ

Lw&Lw&

p Hp H

M 2M F

en 5en

u 1uF

M es

w Ru

Sp ler

litv m

ert en

wi u

nd

ow

Lw&Lw&

p Hp H

M 2M 1

en 5en 0

u 2uF

M es

w No

Sp te

lith m

or en

zw u

ind

ow

Lw&Lw&

p Hp H

M 2M 1

en 2en 1

u 4uF

M es

w O

En utli

de ne

ffil m

es en

u

Lw&Lw&

p Hp H

M 2M 1

en 2 en 2

u 5uF

M es

wF Ol

ile ed

ss ivi

ep sie

ar n

at m

or en

u

Lw&Lw&

p Hp H

M 2M 1

en 2 en 3

u AuF

Ml es

M Fr

ae a

re m

su ee

m ha

e ft

m

en

u

Lw&Lw&

p Hp H

M 2M 1

en 2 en 4

u DuF

Ml es

Re Ta

vie bl

wr ee

ev ha
s ft
m
en
u

Lw&Lw&
p Hp H
M 3M 1
en 6en 5
u BuF

Ml es
Re Pa
va ral
ee lel
ep col
tall ch
art
m
en
u

Lw&Lw&
p Hp H
M 3M 1
en 6en 6
u CuF

Ml es
Re Ta
ve bl
an ee
eel or
all ne
rm
en
u

Lw&Lw&

p Hp H
M 3M 1
en 6en 7
u DuF
Ml es
Re Ta
va bl
ee er
ep ow
t m
en
u

Lw&Lw&
p Hp H
M 3M 1
en 6en 8
u EuF
Ml es
Re Ta
ve bl
an ee
eel ol
u
m
n
m
en
u

Lw&Lw&
p Hp H
M 2M 1
en 3en 9
u 8uF
Ml es
Fi Ve

eld rtr

ul

er

m

en

u

Lw&Lw&

p Hp H

M 2M 1

en 3en A

u 9uF

Ml es

Fi Ru

eld by

ne m

xt en

u

Lw&Lw&

p Hp H

M 2M 2

en 3en 8

u AuF

Ml es

Fi Fo

eld ot

pr no

ev tet

ex

tm

en

u

Lw&Lw&

p Hp H

M 2M 2

en 3en 9

u BuF

Ml esl

Fi nd

eld ex

ev te

al xt

m

en

u

Lw&Lw&

p Hp H

M 2M 2

en 3en A

u CuF

Ml es

Fi To

eld et

ad ex

d tm

en

u

Lw&Lw&

p Hp H

M 2M 2

en 3en B

u DuF

Ml es

Fi Ta

eld bl

up ee

da or

te ne

rm

en

u

Lw&Lw&
p Hp H
M 1M 2
en 0en G
u AuF
Ml es
Fi Ta
eld bl
to et
g ex
tm
en
u

Lw&Lw&
p Hp H
M 1M 2
en 0en D
u BuF
Ml es
Fi Fr
eld a
up m
all et
ex
tm
en
ud
ro
pe
ap

Lw&Lw&
p Hp H
M 1M 2
en 0en E
u GuF

M es

Fi Fr

eld a

loe m

k e

m

en

ud

re

pe

ap

Lw&

p H

M 1

en 0

u D

M

Fi

eld

sa

ve

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:

S S
u u
b b
e e
bj bj
e e
ct ct
— Ty
p
e
" F
Me
y xt
s (p
u lai
m n)
m
er
v
a
e
at
ie
ri:
"
—

~~I I~~
~~A a~~
~~B b~~
~~" "~~
~~W e~~
~~h xt~~
~~at (p~~
~~a lai~~
~~tri n)~~
~~p!~~
~~"~~
~~E E~~
~~O O~~
~~P P~~
~~(e~~
~~r~~
~~E~~
~~n~~
~~d~~
~~O~~
~~f~~
~~P~~
~~ar~~
~~a~~
~~gr~~
~~a~~
~~p~~
~~h)~~

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:

S S
u u
b b
e e
bj bj
e e
ct ct
—Ty
p
e
:" I
Me
y" xt

(b
of
d)
" F
s e
u xt
m (p
m lai
ern)
v
a
e
at
ie
ri:
"
[[[
A a
B]b
" F
We
h xt
at (it
" ali
e)
" F
a e
tri xt
p! (p
" lai
n)
[[[[
O O
P]P
(e

r

E

n

d

Q

f

P

ar

a

gr

a

p

h)

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.

AtDes

tr cript

b ion

ut

e

s

ki

p

p

e

d

m

is

s

p

el

le

d

hi

d

d

e

n

gr

a

m

m

ar

er

ro

r

pr

ot

e

et

e

d

n

e-

h

y

p

h

e

n

d

e

u

bl

e-

w

or

d

hi

g

hl

ig

ht

e

d

re

vi

si

e

n

la

n

g
u
a
g
e
fo
nt
si
z
e
fo
nt
fa
e
e
e
ol
or
e
h
ar
a
et
er
st
yl
e
at
tri
b
ut
e
s
e
e

d
e
p
a
g
e

The table below provides descriptions of each of these Text subobjects.

S Des
u cript
b ion
e
bj
e
ct
N
a
m
e
A A
n Fra
e me
h whi
orch
e is
d plac
F ed

ra"In
mText
e ":-
"In
Text
=
Vert
ical"
.or
"Wit
h
Par
agr
aph
Abe
ve"
B The
e Mar
e ker
k for
meith
arer
k the
Star
t or
End
of a
Boo
kma
rk
ran
ge:
G The
li Mar
e ker

k for
H eith
erer
e the
BlStar
o t or
e End
k of a
Glic
kHe
reBl
ock.
GA
ol har
u d
m colu
n mn
B bre
reak
a with
k in a
mult
i
colu
mn
pag
e
layo
ut.
This
is
ere
ate
d by
cho

osin

g-

Text

=

Inse

rt-

Oth

er-

and

the

n-

Col

um

n-

Bre

ak-

D The

D Mar

E ker-

for-

eith

er-

the-

Star

t or-

End

of a

DD

E-

link-

D A-

e Dec

e Info

V field

ar

ia
bl
e
E The
O mar
P ker
at
the
End
Of a
Par
agr
aph
ere
ate
d by
pre
ssin
g-
EN
TE
R.
F The
o Mar
ot ker
n for
oteith
e er
Mthe
arStar
k t or
End
of a
Fee
tnot

e:-
H A
arnon
d :-
S bre
p akin
a g
e spa
e ce
cre
ate
d by
pre
ssin
g
CT
RL+
Spa
ceb
ar
H A
K spe
atcial
a Asi
k an
a lang
n uag
a e
cha
ract
er
See
n in
Asi
an

lang
uag
e-
vers
ions
only

±

K A-
a spe
nj cial-
i Asi
an-
lang
uag
e-
cha
ract
er-
See
n in-
Asi
an-
lang
uag
e-
vers
ions
only

±

Li A-
n soft
e line-
B bre
reak-
a ere

k ate
d by
pre
ssin
g
SHI
FT+
EN
TE
R.

NA
otCo
e mm
ent
Not
e
cre
ate
d by
cho
osin
g
Gre
ate

=
Co
mm
ent
Not
e.

PA
a har
g d
e pag
Be

rebre
a ak
k ere
ate
d by
pre
ssin
g
GT
RL+
EN
TE
R.
PA
a pag
g e
e nu
N mb
u er
m cre
b ate
erd by
cho
osin
g
Text
=
Inse
rt
Pag
e
Nu
mb
er.
P The

aroutli

a ne-

N styl

u e-

m seq

b uen

er ce-

posi

tion-

nu

mb

er-

for-

a-

par

agr

aph

±

P The

e Mar

w ker-

erfor-

Fi eith

e l er-

d the-

Star

t or-

End

of a

Pe

wer

Fiel

d:

R Ava

u ifabl

b e
y only
F in
rathe
m Asi
e an
* lang
uag
e
vers
ion
of
Wor
d
Pro
R Ava
u itabl
b e
y only
Min
arthe
k Asi
eran
* lang
uag
e
vers
ion
of
Wor
d
Pro
S A
e Sec
et tion

io mar
n ker
whi
ch
defi
nes
the
sep
arat
ion
of
two
sect
ions
with
in a
Wor
d
Pro
doc
um
ent.

S A
of hyp
t hen
H whi
y ch
p doe
h s
e not
n app
ear
unle
ss
the

hyp

hen

ate

d-

wor

d-

flow

s to

the

end

of a

line

where

re it

wou

ld-

bre

ak

nat

urall

y:

S Spe

p cial

e Tab

ci s-

al cont

T ain-

a thei

b r-

own

tab-

stop

and

alig

nm

ent
info
rma
tion.
Nor
mal
Tab
mar
kers
deri
ve
thei
r
tab
stop
and
alig
nm
ent
info
rma
tion
fro
m
the
Rul
er.
Spe
cial
Tab
mar
kers
are
cre
ate

~~d in~~
~~two~~
~~way~~
~~s:~~
~~the~~
~~Inse~~
~~rtSp~~
~~ecia~~
~~lTab~~
~~met~~
~~hod~~
~~or~~
~~by~~
~~imp~~
~~orti~~
~~ng~~
~~a~~
~~doc~~
~~um~~
~~ent~~
~~fro~~
~~m a~~
~~file~~
~~for~~
~~mat~~
~~whi~~
~~ch~~
~~use~~
~~s~~
~~Spe~~
~~cial~~
~~Tab~~
~~s~~
~~(suc~~
~~h as~~

Wor

dPe

rfee

t):

I A-

a nor

b mal-

Tab-

mar

ker-

cre

ate

d by

pre

ssin

g-

TAB

=

I A-

a Tabl

ble-

e whi

ch-

is-

plac

ed-

"In-

Text

"-

"In-

Text

=

Vert

ical"

-or-

"Wit
h-
Par
agr
aph
Abo
ve".
I AS
e CH-
xt text
cha
ract
ers.
For
det
ails.
see-
The
Diff
ere
nt-
Typ
es-
of
AS
CH-
Text
belo
w.
I A-
e revi
m sion
b mar
st ker-
e whi

n ch-
e disp
lays
the-
initi
als-
of-
the-
aut
hor-
of a
part
icul
ar-
revi
sion

=

U A-
ni spe
G cial-
o fore
d ign-
e lang
uag
e-
type
of-
cha
ract
er-
use
d in-
mult
i-
lang

uag

e-

doc

um

ents

i

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:

Cl

S as

u s-

b Na

e m

e e-

bj of

e Co

ct rre

N sp

a on

m di

e -ng

Lo

tu

sS

—

eri
pt
Q
bj
ee
t
I Ta
a bl
bl eL
e ay
ou
t
Su
pe
rI
ab
le
La
ye
ut
A Fr
n a
e m
h eL
oray
e ou
d t
F
ra
m
e
F Fo
e ot
ot no
n te

ot La

e yø

Mut

ar

k

S Se

e eti

et on

ie

n

N No

otte

e La

yø

ut

B Bo

e øk

e M

k ar

mk

ar

k

D Dd

D eL

E ink

P Pø

e we

w rFi

erel

Fið

el

ð

G Gli

li ek

e He

k re
H
er
e
Bl
e
e
k
R Ru
u by
b M
y ar
Mke
arr
k (A
ersia
n-
la
ng
ua
ge
ve
rsi
on
s-
on
ly)
R Ru
u by
b La
y ye
F ut
ra(A
msia
e n-

la
ng
ua
ge
ve
rsi
on
s
on
ly

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:

DValS

au*e*:i

t*fra* z

afng e

-i*e*

t*x*

y

p

e

l*:-* 2

n%2*.b*

t 76 y

e8 t

g*to* e

e32*.s*

r 76

7*ln*

it*ia*

t

val

ue:

0

L*:-* 4

o&2*.1b*

n47*.y*

g48 t

3.6e

48*-s*

to

2.1

47.

48

3.6

471

niti

at

val

ue:

0

S- 4

i !3.4b

n02 y

g82 t

! 3E e

e+3 s

8

to

3.4

02

82

3E

+3

8In

itia

t

val

ue:

0

D- 8

o#1.7b

u97 y

b69 t

! 31 e

e34 s

86

23

15

8+

30

8-

to-

1.7

97

69

31

34

86

23

15

8+

30

8ln

itia

t

val

ue:

0

G- 8

u~~0~~ b

r 2.3y

r 37.t

e20 e

n3.6s

e 85.

y 47

7.5

80

7

to-

92

2.3

37.

20

3.6

85.

47

7.5

80

7In

itia

t

val

ue:

0

S(St(

t\$in 2

r g b

i leny

ngtht

gra e

ng s

es/

fro e

m h

0 a

to r

32 a

K e

ch t

ar e

act r)

ers

)Ini

tial

val
ue:
'''
(e
mp
ty
stri
ng
)

Besides these scalar data types, LotusScript supports the following additional data

types and data structures:

D : :
at D Si
a e z
ty s e
p er
e ip
orti
st o
run
ct
ur
e
ArA U
ran p
y a to
g 6
gr4
e K

g b
at yt
e e
s s
et
of
el
e
m
e
nt
s
h
a
vi
n
g
th
e
s
a
m
e
d
at
a
ty
p
e.
A
n
ar
ra
y
e

a
n
e
e
m
pr
is
e
u
p
to
g
di
m
e
n
si
e
n
s
w
h
e
s
e
s
u
b
s
er
ip
t
b
e
u

n
d
s
e
a
n
ra
n
g
e
fr
e
m
=
3
2
7
6
8
to
3
2
7
6
7.
In
iti
al
v
al
u
e:
E
a
e

h
e
e
m
e
nt
in
a
fi
x
e
d
ar
ra
y
h
a
s
a
n
in
iti
al
v
al
u
e
a
p
pr
e
pr
ia
te
to

it
s-
d
at
a-
ty
p
e.

Li A U

st o p-
n to
e-6
di 4
m K
e b
n yt
si e
e s
n
at
a
g
gr
e
g
at
e-
s
et
w
h
e
s
e-
el

e
m
e
nt
s
h
a
v
e
th
e
s
a
m
e
d
at
a
ty
p
e
a
n
d
ar
e
re
fe
rr
e
d
to
b
y
n

a
m
e
ra
th
er
th
a
n
b
y
s
u
b
s
er
ip
t

V A 1

ars 6

ia p b

nte yt

ci e

al s

d

at

a

ty

p

e

th

at

e

a

n

e
e
nt
ai
n-
a
n
y-
s
e
al
ar
v
al
u
e-
ar
ra
y-
lis
t-
or
e
bj
e
et
re
fe
re
n
e
e-
ln
iti
al

v

at

u

e:

E

M

P

I

Y

U A U

s n p

era to

- g 6

d gr4

efe K

in g b

e atyt

d e e

d s s

at et

a of

ty el

p e

e m

e

nt

s

of

p

e

s

si

bl

y

di

s
p
ar
at
e
d
at
a
ty
p
e
s.
G
e
m
p
ar
a
bl
e
to
a
re
e
or
d
in
P
a
s
e
at
or
a
st

ru
et
ur
e
in
G
=
In
iti
al
v
al
u
e:
M
e
m
b
er
v
ar
ia
bl
e
s
h
a
v
e
in
iti
al
v
al
u

e
s
a
p
pr
e
pr
ia
te
to
th
ei
r
d
at
a
ty
p
e.
GIA
a n
ssa
g
gr
e
g
at
e
s
et
of
el
e
m
e

nt
s
of
p
e
s
si
bl
y
di
s
p
ar
at
e
d
at
a
ty
p
e
s
to
g
et
h
er
wi
th
pr
e
e
e
d
ur

e
s
th
at
e
p
er
at
e
e
n
th
e
m
=
ln
iti
al
v
al
u
e:
W
h
e
n
y
e
u
er
e
at
e
a
n

in
st
a
n
e
e
of
a
cl
a
s
s.
L
et
u
s
S
er
ip
t
in
iti
al
iz
e
s
it
s
m
e
m
b
er
v
ar

ia
bl
e
s-
to
v
al
u
e
s-
a
p
pr
e
pr
ia
te
to
th
ei
r-
d
at
a-
ty
p
e
s.
a
n
d-
g
e
n
er

at
e
s
a
n
e
bj
e
ct
re
fe
re
n
e
e
to
it.

QA4

bj p b
e oi yt
ct nte
re ers
fe to
rea
n n
e O
e L

E
A
ut
e
m
at
io
n

e
bj
e
et
or
a
n-
in
st
a
n
e
e-
of
a-
pr
e
d
u
et
el
a
s
s-
or
u
s
er
-
d
ef
in
e
d-
el

a
s
s
ln
iti
al
v
al
u
e
N
Q
I
H
t
N
G
E

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_USIN

G_LOTUSSCRIPT_IN_WORD_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: DblUnderline 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.

.DblUnderline

Messagebox "Click OK undo double underline change.", MB_OK, "Example Script" _____

.DblUnderline

'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

fbutton ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGR
OUPLAYOUT_CLASS;H_GELLYAYOUT_CLASS;H_COLUMNNGROUPPLAYOUT_GLAS
S;H_CONNECTEDLAYOUT_CLASS;H_DIVISIONINFO_CLASS;H_DROPGAPGONTAI
NER_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOT
ERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS
;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT
_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTECONTAINER_GL
ASS;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELAYOUT_CLA
SS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWG
ROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_R

UBYLAYOUT_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER

R_CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGROUPLAYOUT_

CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEH

EADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;

H_TOCSUPERTABLELAYOUT_CLASS: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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_Basetable_Class;H_Content_Class;H_FootnoteTable_Cl

Ass;H_Formula_Class;H_Glossary_Class;H_Graphic_Class;H_Graphi

OLEObject_Class;H_OLEObject_Class;H_ParallelColumns_Class;H_

Supertable_Class;H_Table_Class;H_TableHeading_Class;H_Text_Cla

SS';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

ValEff
ue ect
\$L Indi
wp cat
Go es-
nte that
ntTthe-
yp con
eT tent
ext typ
{1 e is
61 a-
} Tex
t
obj
ect.
\$L Indi
wp cat
Go es-
nte that
ntTthe-
yp con
eT tent
abl typ
eo e is
nly a-
{1 Tab
62 leO
} nly-

obj
ect.
\$L Indi
wp cat
Go es-
nte that
ntTthe-
yp con
eG tent
ra typ
phie is
ε a
{1 Gra
63 phi
} ε

obj
ect.
\$L Indi
wp cat
Go es-
nte that
ntTthe-
yp con
eO tent
leo typ
bje e is
ct a
{1 Ole
64 Obj
} ect.

\$L Indi
wp cat
Go es-
nte that
ntTthe-

yp con

eP tent

ar typ

all e is

etc a-

olu Par

mnalle

s- lGo

{1 lum

65 ns-

} obj

ect.

\$L Indi

wp cat

Go es-

nte that

ntThe-

yp con

eF tent

ootyp

note is

etaa-

ble Fo

{1 otn

66 ote

} Tab

le-

obj

ect.

\$L Indi

wp cat

Go es-

ntt that

entthe-

Ty con

pe tent
Fe typ
rm e-is
ula a-
{1 For
67 mul
} a-
obj
ect.
\$L Indi
wp cat
Co es-
nte that
ntTthe-
yp con
eF tent
or typ
mue is
la a-
{2 For
05 mul
6) a-
obj
ect.

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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

fbutton ,AL('H_BAGCOLLECTION_CLASS;H_BASECOLLECTION_CLASS;H_BOOKM

ARKCOLLECTION_CLASS;H_GELLCOLLECTION_CLASS;H_GELLLAYOUTCOLLEG

TION_CLASS;H_CHARACTERSTYLECOLLECTION_CLASS;H_GLICKHERECOLLEG

TION_CLASS;H_CONNECTEDLAYOUTCOLLECTION_CLASS;H_CONTENTCOLLEG

TION_CLASS;H_DDELINKCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;

H_DOGINFOFIELDGCOLLECTION_CLASS;H_DOCUMENTS_CLASS;H_DOGWINDO

WCOLLECTION_CLASS;H_DROPCAPLAYOUTCOLLECTION_CLASS;H_EDITORCO

LLECTION_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOU

TCOLLECTION_CLASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOU

TCOLLECTION_CLASS;H_FRAMELAYOUTCOLLECTION_CLASS;H_GLOSSARYCO

LLECTION_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHICOLEBJECTCO

LLECTION_CLASS;H_GROUPLAYOUTCOLLECTION_CLASS;H_HEADERLAYOUTC

OLLECTION_CLASS;H_ICONBARCOLLECTION_CLASS;H_LAYOUTCOLLECTION_

CLASS;H_MARKERCOLLECTION_CLASS;H_MENUITEMCOLLECTION_CLASS;H_N

OTELAYOUTCOLLECTION_CLASS;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLI

NESEQCOLLECTION_CLASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGE

LAYOUTCOLLECTION_CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PAR

ALLELCOLSCOLLECTION_CLASS;H_POWERFIELDCOLLECTION_CLASS;H_ROWL

AYOUTCOLLECTION_CLASS;H_RUBYLAYOUTCOLLECTION_CLASS;H_SECTIONG

OLLECTION_CLASS;H_SILVERBULLETCOLLECTION_CLASS;H_SMARTCORRECT

COLLECTION_CLASS;H_SMARTFILLCOLLECTION_CLASS;H_STATUSBARBUTTO

NGCOLLECTION_CLASS;H_STRINGCOLLECTION_CLASS;H_SUPERTABLECOLLEC
TION_CLASS;H_SUPERTABLELAYOUTCOLLECTION_CLASS;H_TABLECOLLECTIO
N_CLASS;H_TABLEHEADINGCOLLECTION_CLASS;H_TABLEHEADINGLAYOUTCO
LLECTION_CLASS;H_TABLELAYOUTCOLLECTION_CLASS;H_TABLEMARKERCOL
LECTION_CLASS;H_TABLEONLYCOLLECTION_CLASS;H_TEXTCOLLECTION_CLA
SS;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_DATEREVALUEDVALUE_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_PREFEREN

GES_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: DefaultRightColumnNameProperty

{button ,AL('H_GLOSSARY_CLASS;H_PARALLELCOLUMNS_CLASS;H_PREFEREN

GES_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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_CHARACT
ERSTYLE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_C
LASS;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_PAGELAYOUT_CLASS;H_PARAGRA
PHSTYLE_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RU
BYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAY
OUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TO
CSUPERTABLELAYOUT_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_G

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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

WPAplication 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ValEff

ue ect
Lw Set
pL ting
ay this
outval
Dir ue
ect spe
ion cifi
Us es
eD that
ire a
cti lay
on out
(& obj
H1 ect
) will
adh
ere
to
the
oth
er
bit
val
ues
set
for
this
pro
per
ty:
Lw Set
pL ting
ay this
outval

Dir ue
ect spe
ion cifi
Au es
to that
Gr a
ow lay
{& out
H2 obj
) ect
aut
ogr
ow
s-in
a
do
wn
war
d-
dire
ctio
n:

Lw Set
pL ting
ay this
outval
Dir ue
ect spe
ion cifi
Au es
to that
Siza
e lay
{& out
H4 obj

~~ject~~
~~aut~~
~~osi~~
~~zes~~
~~in a~~
~~do~~
~~wn~~
~~war~~
~~d-~~
~~dire~~
~~ctio~~
~~n.~~

~~Lw Set~~
~~pl ting~~
~~ay this~~
~~outval~~
~~Dir ue-~~
~~ect spe~~
~~ion cifi~~
~~To es-~~
~~co that~~
~~ntaa-~~
~~ine lay~~
~~r out-~~
~~{& obj~~
~~H8 ect~~

~~ject~~
~~ogr~~
~~ow~~
~~s-or~~
~~aut~~
~~osi~~
~~zes~~
~~to-~~
~~the-~~

bott

om

edg

e of

its

con

tain

er.

Lw Set

pl ting

ay this

outval

Dir ue

ect sim

ion ulta

All neo

Dir usl

ect y

ion aut

s ogr

{& ow

HF s or

} aut

osi

zes

a

lay

out

obj

ect

in

all

dire

ctio

ns

~~(de
wn;
up;
left;
and
righ
t).~~

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

ValEff

ue ect

Lw Set

pL ting

ay this
outval
Dir ue-
ect spe
ion cifi
Us es-
eD that
ire a-
cti lay
on-out
{& obj
H1 ect
} will-
adh
ere
to-
the-
oth
er-
bit-
val
ues
set-
for-
this
pro
per
ty.

Lw Set
pL ting
ay this
outval
Dir ue-
ect spe
ion cifi

Au es-
to that
Gr a-
ow lay
{& out-
H2 obj
} ect
aut
ogr
ow
s-
tow
ard
the-
left.

Lw Set
pL ting
ay this
outval
Dir ue-
ectspe
ion cifi
Au es-
to that
Siza-
e lay
{& out-
H4 obj
} ect
aut
osi
zes
tow
ard
the-

left.
Lw Set
pL ting
ay this
outval
Dir ue
ectspe
ioncifi
To es
co that
ntaa
ine lay
r out
(& obj
H8 ect
) aut
ogr
ow
s-or
aut
osi
zes
to
the
left
edg
e-of
its
con
tain
er.
Lw Set
pL ting
ay this
outval

Dir ue-
ect sim
ion ulta
All neo
Dir usl
ect y-
ion aut
s: ogr
{& ow
HF s or
} aut
osi
zes
a-
lay
out
obj
ect
in-
all-
dire
ctio
ns-
{do
wn,
up,
left,
and
righ
t}.

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

ValEff

ue ect

Lw Set

pL ting

ay this
outval
Dir ue-
ect spe
ion cifi
Us es-
eD that
ire a-
cti lay
on-out
{& obj
H1 ect
} will-
adh
ere
to-
the-
oth
er-
bit-
val
ues
set-
for-
this
pro
per
ty.

Lw Set
pL ting
ay this
outval
Dir ue-
ect spe
ion cifi

Au es-
to that
Gr a-
ow lay
{& out
H2 obj
} ect
aut
ogr
ow
s-
tow
ard
the-
righ
t:

Lw Set
pL ting
ay this
outval
Dir ue-
ectspe
ion cifi
Au es-
to that
Siza-
e lay
{& out
H4 obj
} ect
aut
osi
zes
tow
ard

the
righ
t:
Lw Set
pL ting
ay this
outval
Dir ue
ectspe
ion cifi
To es
co that
ntaa
ine lay
r out
(& obj
H8 ect
) aut
ogr
ow
s-or
aut
osi
zes
to
the
righ
t
edg
e-of
its
con
tain
er:
Lw Set

~~pl~~ ~~ting~~
~~ay~~ ~~this~~
~~out~~ ~~val~~
~~Dir~~ ~~ue~~
~~ect~~ ~~sim~~
~~ion~~ ~~ulta~~
~~All~~ ~~neo~~
~~Dir~~ ~~usl~~
~~ect~~ ~~y~~
~~ion~~ ~~aut~~
~~s~~ ~~ogr~~
~~{~~ ~~&~~ ~~ow~~
~~HF~~ ~~s~~ ~~or~~
~~}~~ ~~aut~~
~~osi~~
~~zes~~
~~a~~
~~lay~~
~~out~~
~~obj~~
~~ect~~
~~in~~
~~all~~
~~dire~~
~~ctio~~
~~ns~~
~~{~~ ~~do~~
~~wn~~
~~up~~
~~left~~
~~and~~
~~right~~
~~t~~

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

ValEff

ue ect

Lw Set

pL ting

ay this
outval
Dir ue-
ect spe
ion cifi
Us es-
eD that
ire a-
cti lay
on-out
{& obj
H1 ect
) will-
adh
ere
to-
the-
oth
er-
bit-
val
ues
set-
for-
this
pro
per
ty.

Lw Set
pL ting
ay this
outval
Dir ue-
ect spe
ion cifi

Au es-
to that
Gr a-
ow lay
{& out
H2 obj
} ect
aut
ogr
ow
s-in
a-
up
war
d-
dire
ctio
n-

Lw Set
pL ting
ay this
outval
Dir ue-
ectspe
ioncifi

Au es-
to that
Siza-
e lay
{& out
H4 obj
} ect
aut
osi
zes

in a
up
war
d-
dire
ctio
n-
Lw Set
pL ting
ay this
outval
Dir ue-
ect spe
ion cifi
To es-
co that
ntaa-
ine lay
r- out
(& obj
H8 ect
) aut
ogr
ow
s-or
aut
osi
zes
to-
the-
top-
edg
e-of
its-
con

tain
er.
Lw Set
pL ting
ay this
outval
Dir ue
ect sim
ion ulta
All neo
Dir usl
ect y
ion aut
s ogr
{& ow
HF s or
) aut
osi
zes
a
lay
out
obj
ect
in
all
dire
ctio
ns
{de
wn.
up.
left.
and
righ

t):

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_CLASS;H_TOGSUPERTABLELAYOUTCONTAINER_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

N_CLASS:H_DIVISION_CLASS:H_TEXTDOCUMENT_CLASS:H_WPAPPLICATION_C

LASS';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_DRIVERNAME DRIVERNAME_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

*fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_
DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CL
ASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPER
TABLELAYOUT_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

ValEff
ue ect
\$L An
wp edit
Ed or
itormu
Ve st
rifyuse
Ty an
pe E
By mai
E t
malogi
ilL nto
ogiacc
n ess
(2 a
36 doc
l um
ent
\$L An
wp edit
Ed or
itormu
Ve st
rifyuse
Ty an
pe ope
By rati
Opng
sy sys
sL tem

~~ogi logi~~
~~n- n-to~~
~~{2 acc~~
~~35 ess~~
~~} a-~~
~~doc~~
~~um~~
~~ent.~~
~~\$L An-~~
~~wp edit~~
~~Ed or-~~
~~itormu~~
~~Ve st-~~
~~rify use~~
~~Ty a-~~
~~pe use~~
~~By r-~~
~~Us na~~
~~er me-~~
~~Na to-~~
~~meacc~~
~~{2 ess~~
~~34 a-~~
~~} doc~~
~~um~~
~~ent.~~

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 indentation 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

embedfontsvalue = [objectreference].EmbedFonts

[objectreference].EmbedFonts = embedfontsvalue

Legal values

The legal values for this property are determined by its data type. For 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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_GLOSSARY_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_GLA

SS;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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_
DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_GL
ASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPER
TABLELAYOUT_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_GLA

SS;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: Import/Inserted 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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_
DROPGAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_GL
ASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPER
TABLELAYOUT_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 WPAApplication

object, this argument returns the WPAApplication 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: WPAApplication, 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 (WPAApplication).

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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_
DROPGAPLAYOUT_CLASS;H_ENDNOTE_LAYOUT_CLASS;H_FOOTERLAYOUT_CL
ASS;H_FOOTNOTE_LAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGE_LAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROW_LAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLE_LAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLE_LAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPER
TABLE_LAYOUT_CLASS;H_WPAPPLICATION_CLASS';0)} See list of classes

{button .AL('H_MOUSEDOWN_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_
DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_GL
ASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPER
TABLELAYOUT_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_DOCWI

NDOW_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_G

LASS;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_G

LASS;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_G

LASS;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_STATUSBARBUTTONOVERRIDE TEXTANDGRAPHIC_EVENT_EXSG

RIPT',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_WPAPPLI

GATION_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 WPAApplication.

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:

∕

æ

!y

⓪

—

~~ut~~

~~o~~

~~bi~~

~~e~~

~~et~~

~~C~~

~~r~~

~~e~~

~~at~~

~~e~~

~~d~~

~~P~~

~~a~~

~~g~~

~~e~~

~~L~~

~~a~~

~~y~~

~~e~~

~~ut~~

~~N~~

~~ot~~

~~e~~

~~L~~

~~a~~

~~y~~

~~e~~

~~ut~~

~~F~~

~~G~~

~~z~~

~~m~~

~~e~~

~~L~~

~~a~~

Y
e
ut
R
2
S
Y
L
a
Y
e
ut

\$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:

16

17

18

19

20

21

22

j
e
e
t
G
f
e
a
t
e
d
P
a
g
e
F
f
a
m
e
Z
a
b
i
e
Z
S
H
P
S
f
a
g
f
a

ð
h
æ
h
a
f
a
e
t
e
f
H
e
z
d
e
f
F
e
t
e
f
e
f
æ
p
e
a
p

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_WPAPPLICATI

ON_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: DblUnderline 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 "Dbl Underline" from the

Attributes box.

Syntax

[objectreference].DblUnderline()

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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CLICKHE

RE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;

H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_

CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRA

MELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LA

YOUT_CLASS;H_MARKER_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CL

ASS;H_POWERFIELD_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_G

LASS;H_RUBYLAYOUT_CLASS;H_RUBYMARKER_CLASS;H_SUPERTABLEGROUP

LAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CL

ASS;H_TABLELAYOUT_CLASS;H_TABLEMARKER_CLASS;H_TEXTMARKER_GLAS

S;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_GRAPHICOLEOBJECT_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: DeleteItem 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].DeleteItem(MenuText)

Parameters

MenuText

The String name of the menu item you want to delete from a menu.

Return value

String

Usage

You call the DeleteItem 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 DeleteItem 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 acceletator 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_C

LASS'.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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S;H_WPAPPLICATION_CLASS';0)} See list of classes

{button ,AL('H_DELETETABLE_METHOD_EXSCRIPT',1)} See example

[WPAplication]

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

[WPApplication]

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.

ValEff
ue ect
\$L
wp Del
Table
ble s
De any

~~Hy row~~
~~pe s~~
~~Re whi~~
~~w ch~~
~~(1 are~~
~~86 incl~~
~~7) ude~~
~~d in~~
~~the~~
~~cur~~
~~ren~~
~~t~~
~~sel~~
~~ecti~~
~~on,~~
~~unt~~
~~ess~~
~~you~~
~~spe~~
~~cify~~
~~val~~
~~ues~~
~~for~~
~~the~~
~~Sta~~
~~rt~~
~~and~~
~~Co~~
~~unt~~
~~par~~
~~am~~
~~ete~~
~~rs.~~
~~\$L Del~~
~~wp etc~~

Ta s-
ble any
De col
ITy um
pe ns-
Go whi
lu ch-
mn are
(1 incl
86 ude
8) d-in
the-
cur
ren
t
sel
ecti
on.
unl
ess
you
spe
cify
val
ues
for-
the-
Sta
rt
and
Go
unt
par
am
ete

rs:
\$L Del
wp ete
Ta s-
ble the-
De enti
ITy re-
pe tabl
W e-
hol
eta
ble
(1
86
9)

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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S;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_DROPCA
PCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_GLAS
S;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWG
ONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS
;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TAB
LECONTAINER_CLASS;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

fbutton ,AL('H_BASSETABLE_CLASS;H_GELLGROUPLAYOUT_CLASS;H_GELLLAYO

UT_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;

H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_

CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNOTETABLE_CLASS;H_FRAMEGR

OUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GLOSSARY_CLASS;H_GROUP

LAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT

_CLASS;H_PAGELAYOUT_CLASS;H_PARALLELCOLUMNS_CLASS;H_ROWGROUP

LAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTAB

LEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLE_CLASS;H_

TABLEHEADING_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_GL

ASS;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_GL

ASS;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: WPAplication, _

Division, TextDocument, DocWindow.

Syntax

[objectreference].WPAplication.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_DROP

PCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_GLAS

S;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWG

ONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TAB

LECONTAINER_CLASS;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: DefaultLeftColumnName 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 "Its"

'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: GetEnumerator method

' This example prints the numeric equivalent for the \$LwpMergeActionNewfile

' contant to the Lotus Script Output panel.

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

Dim Value As Long

Value = .GetEnumerator("\$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

MessageBox "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: GetMisspelledWord 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.GetMisspelledWord

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

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: 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(\$LwpGetObjectTyp

eParagraph, 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.

ValEff

~~ue ect~~
~~\$L Th~~
~~wp e-~~
~~Do na~~
~~cV me~~
~~ar s of~~
~~All all-~~
~~verthe-~~
~~sio ver~~
~~nn sio~~
~~amns-~~
~~es of a~~
~~(2 doc~~
~~10 um~~
~~) ent.~~
~~\$L Th~~
~~wp e-~~
~~Do We~~
~~cV rd-~~
~~ar Pro~~
~~Gr use~~
~~eatr~~
~~ed na~~
~~by me-~~
~~(2 of~~
~~04 the-~~
~~6) per~~
~~son~~
~~wh~~
~~e-~~
~~first~~
~~sav~~
~~ed-~~
~~the-~~
~~doc~~

um
ent.
\$L Th
wpe
Do dat
eV e
ar and
Da tim
tece
re the
atedoc
d um
(1 ent
96 wa
) s
first
cre
ate
d.
\$L Th
wpe
Do last
eV tim
ar e
Da the
tel doc
astum
revent
isi wa
on s
(1 ope
97 ned
) =
edit
ed.

and
sav
ed.
\$L Dis
wp pla
Do ys
eV a
ar des
De crip
scrtion
ipti of
on the
{1 doc
95 um
} ent.
\$L Th
wp e
Do na
eV me
ar of a
Di divi
vis sio
ion n
na wit
mehin
{2 the
03 doc
} um
ent.
\$L Th
wp e
De ass
eV ign
ar ed
De cat

~~ee ego~~
~~atery~~
~~go of~~
~~ry the~~
~~(2 divi~~
~~12 sio~~
~~) n-~~
~~or~~
~~doc~~
~~um~~
~~ent.~~
~~\$L Th~~
~~wp e-~~
~~Do siz~~
~~eV e-of~~
~~ar the~~
~~Do doc~~
~~esi um~~
~~ze ent~~
~~(2 in-~~
~~02 kilo~~
~~) byt~~
~~es.~~
~~\$L Sp~~
~~wp ecif~~
~~Do ies-~~
~~eV the~~
~~ar Do~~
~~Fieclnf~~
~~ld e-~~
~~(1 fiel~~
~~91 d-~~
~~) you~~
~~wa~~
~~nt~~

~~to~~
~~exp~~
~~ort~~
~~as~~
~~Not~~
~~es~~
~~FX.~~
~~\$L Th~~
~~wp e~~
~~Do file~~
~~cV na~~
~~ar me~~
~~File you~~
~~en spe~~
~~amcifi~~
~~e ed~~
~~(1 in~~
~~92 the~~
~~) Sa~~
~~ve~~
~~As~~
~~dial~~
~~og~~
~~box~~
~~wh~~
~~en~~
~~you~~
~~sav~~
~~ed~~
~~the~~
~~doc~~
~~um~~
~~ent.~~
~~\$L Dis~~
~~wp pla~~

~~De ys-~~
~~eV the-~~
~~ar ass~~
~~Ke ign~~
~~yw ed-~~
~~or key~~
~~ds wor~~
~~(2 ds-~~
~~15 for-~~
~~) the-~~
~~doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wpe-~~
~~De initi~~
~~eV als-~~
~~ar of~~
~~La the-~~
~~ste edit~~
~~dit or-~~
~~or wh~~
~~(2 e-~~
~~04 last~~
~~7) sav~~
~~ed-~~
~~the-~~
~~doc~~
~~um~~
~~ent.~~

~~\$L No-~~
~~wpe De~~
~~De clnf~~
~~eV e-~~
~~ar fiel~~

~~No ds-~~
~~ne are~~
~~(1 set~~
~~90 or~~
~~) res~~
~~et~~
~~as~~
~~Not~~
~~es~~
~~FX.~~

~~\$L Th~~
~~wpe-~~
~~Do nu~~
~~eV mb~~
~~ar er-~~
~~Nu of~~
~~me cha~~
~~ha ract~~
~~rs- ers-~~
~~(2 in-~~
~~01 the-~~
~~) doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wpe-~~
~~Do nu~~
~~eV mb~~
~~ar er-~~
~~Nu of~~
~~mppag~~
~~ag es-~~
~~es in-~~
~~(1 the-~~
~~99 doc~~

um
ent.

\$L Th

wpe

Do nu

cV mb

ar er

Nu of

mv ver

erssio

ion ns

s in

{2 the

09 doc

um

ent.

\$L Th

wpe

Do nu

cV mb

ar er

Nu of

m wor

wo ds

rdsin

{2 the

00 doc

um

ent.

\$L Th

wpe

De We

cV rd

ar Pro

Of use

he r
re na
dit me
orss of
(2 any
04 oth
8) er
peo
ple
wh
e
sav
ed
the
doc
um
ent.
\$L Th
wp e
Do Wo
cV rd
ar Pro
Ot use
he r
rvena
rsi me
on s of
ed any
edi oth
for er
s peo
(2 ple
07 wh
) e
sav

~~ed~~
~~a~~
~~ver~~
~~sio~~
~~n of~~
~~the~~
~~doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wpe~~
~~Do driv~~
~~eV e~~
~~ar and~~
~~Pa fold~~
~~th er~~
~~(1 wh~~
~~93 ere~~
~~) the~~
~~doc~~
~~um~~
~~ent~~
~~is~~
~~loc~~
~~ate~~
~~d.~~

~~\$L Th~~
~~wpe~~
~~Do na~~
~~eV me~~
~~ar of a~~
~~Se sec~~
~~cti tion~~
~~on in~~
~~na the~~

medoc
(2 um
04 ent.
)}
\$L Th
wpe-
Do styl
cV e-
ar she
St et
yle use
sh d in
eetthe-
(1 doc
94 um
) ent.
\$L Th
wpe-
Do tota
cV t-
ar nu
Tot mb
ale er-
ditt of
im min
e- ute
(1 s-
98 the-
) doc
um
ent-
wa
s-
ope
n-

~~\$L Th~~
~~wp e-~~
~~Do dat~~
~~cV e-~~
~~ar and~~
~~Ve tim~~
~~rsi e-~~
~~on the-~~
~~crever~~
~~atesio~~
~~datn-~~
~~e wa~~
~~(2 s-~~
~~06 first~~
~~)} cre~~
~~ate~~
~~d.~~

~~\$L Th~~
~~wp e-~~
~~Do Wo~~
~~cV rd-~~
~~ar Pro~~
~~Ve use~~
~~rsi r-~~
~~on na~~
~~creme-~~
~~ateof~~
~~db the-~~
~~y per~~
~~(2 son~~
~~05 wh~~
~~)} e-~~
~~first~~
~~sav~~
~~ed-~~

the
ver
sio
n of
the
doc
um
ent.

\$L Th
wpe
Do dat
eV e
ar and
Ve tim
rsi e
onl that
ast the
ediver
tdasio
te n
(2 wa
13 s
) last
sav
ed.

\$L Th
wpe
Do initi
eV als
ar of
Ve the
rsi edit
onl or
ast wh
edi e

~~ted last~~
~~by sav~~
~~(2 ed~~
~~14 the~~
~~) ver~~
~~sio~~
~~n.~~
~~\$L Th~~
~~wpe~~
~~Do na~~
~~cV me~~
~~ar of~~
~~Ve the~~
~~rsi cur~~
~~on ren~~
~~na t~~
~~mever~~
~~(2 sio~~
~~08 n.~~
~~)~~
~~\$L Th~~
~~wpe~~
~~Do nu~~
~~cV mb~~
~~ar er~~
~~Ve of~~
~~rsi tim~~
~~on es~~
~~nu the~~
~~mr ver~~
~~evi sio~~
~~sio n.~~
~~ns wa~~
~~(2 s~~
~~04 ope~~

9) ned

=

edit

ed.

and

sav

ed.

\$L An

wp y

Do re

cV ma

ar rks

Ve writ

rsi ten

on by

re an

maedit

rks or

(21to

1) be

revi

ew

ed

by

any

oth

er

ass

ign

ed

edit

ors.

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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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.

ValEff
ue ect
\$L Indi
wp cat
Pa es
re wh
ntTeth
yp er
eG the
ell na
(1 me
63 of
0) the
par
ent
con

tain
er
lay
out
you
wa
nt
to
ret
urn
is a
cell
con
tain
er.

\$L Indi
wp cat
Pa es
re wh
ntTeth
yp er
eF the
ra na
meme
(1 of
62 the
9) par
ent
con
tain
er
lay
out
you
wa

nt
to
ret
urn
is a
fra
me
con
tain
er.

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_CL

ASS;H_TEXTMARKER_CLASS;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:

[WPAApplication]

Executes a Find based on the values set in the FindAndReplace object. No parameters

are required when calling this method from WPAApplication.

[BookmarkManager]

Finds a specific bookmark by looking for the user-defined bookmark name or the

internal (hexidecimal) 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].WPAApplication.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

WPAApplication 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_FORGEDOCTOLOAD_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

fbutton ,AL('H_BASECONTAINER_CLASS;H_CELLCONTAINER_CLASS;H_CELLGR
OUPLAYOUT_CLASS;H_GELLLAYOUT_CLASS;H_CLICKHERE_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPCONTAIN
R_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTER
LAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMECONTAINER_CLASS;H_
FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_GL
ASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTECONTAINER_CLASS
;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;H_PAGELAYOUT_CLASS;
H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_ROWGRO
UPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYCONTAINER_CLASS;H_RUB

YLAYOUT_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_
CLASS;H_SUPERTABLECONTAINER_CLASS;H_SUPERTABLEGROUPLAYOUT_GL
ASS;H_SUPERTABLELAYOUT_CLASS;H_TABLECONTAINER_CLASS;H_TABLEHEA
DINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABLEONLYCONT_CLASS;H_T
EXT_CLASS;H_TEXTMARKER_CLASS;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_GRAPHICOLEOBJECT_

CLASS;H_OLEOBJECT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPA

PLICATION_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: GetEnumerator 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].GetEnumerator(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_GETLINKSOURCE_NAME_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

fbutton ,AL('H_BASSETABLE_CLASS;H_GELLGROUPLAYOUT_CLASS;H_GELLLAYO

UT_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;

H_DDELINKMANAGER_CLASS;H_DROPGLAYOUT_CLASS;H_ENDNOTELAYOU

T_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNO

TETABLE_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H

_GLOSSARY_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_L

AYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARALLELC

OLUMNNS_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RU

BYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAY

OUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAY

~~OUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACT

ERSTYLE_CLASS;H_CLICKHERE_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_GROUP

LAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_MARKER_GLA

SS;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_SUPERTAB

LAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;

H_TABLEMARKER_CLASS;H_TEXTDOCUMENT_CLASS;H_TEXTMARKER_CLASS;

H_TOGSUPERTABLELAYOUT_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_GETPARANUMBER_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_CLICKHE

RE_CLASS;H_DROPCAPCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_G

RAPHIC_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PA

RALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAIN

ER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;

H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEONL

YCONT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_WPAPPLICATION_C

LASS';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:

RDef

einiti

t on

u

f

n

=

√

a

!

u

e

-Th

-e

Zins

erti

on

poi

nt

is

not

loc

ate

d in

the

sa

me

text

stre

~~am-~~
~~as-~~
~~the-~~
~~spe~~
~~cifi~~
~~ed-~~
~~Ma~~
~~rke~~
~~r-~~
~~obj~~
~~ect.~~
~~-Th~~
~~-e-~~
~~ins~~
~~erti~~
~~on-~~
~~poi~~
~~nt-~~
~~is-~~
~~loc~~
~~ate~~
~~d-~~
~~bef~~
~~ore~~
~~the-~~
~~Ma~~
~~rke~~
~~r-~~
~~obj~~
~~ect.~~
~~-Th~~
~~De-~~
~~ins~~
~~erti~~
~~on-~~

~~poi~~
~~nt~~
~~is~~
~~loc~~
~~ate~~
~~d~~
~~wit~~
~~hin~~
~~the~~
Ma
rke
r
obj
ect.
~~Th~~
~~le~~
ins
erti
on
poi
nt
is
loc
ate
d
afte
r
the
Ma
rke
r
obj
ect.

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 "\$LwplniCustomFile" 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 "\$LwplniUserPrefs."

\$LwplniUserPrefs (2101)

Default for this parameter. This is the .INI file used to store Word Pro's user preference information (lwpuser.ini).

\$LwplniConfigPrefs (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: GetRightMouseMenu method

{button ,AL('H_GRAPHIC_CLASS';0)} See list of classes

{button ,AL('H_GETRIGHTMOUSEMENU_METHOD_EXSCRIPT';1)} See example

Syntax

[objectreference].GetRightMouseMenu()

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

ButtonType

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)

p!Storage

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.

\$LwpGetObjectTypeInfo (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)

~~\$LwpTOGScopeDestStyle (1854)~~

~~\$LwpTOGScopeLevelNumber (1851)~~

~~\$LwpTOGScopeRtAlignPgNum (1850)~~

~~\$LwpTOGScopeSearchStyle (1852)~~

~~\$LwpTOGScopeTypeOfLeader (1846)~~

~~\$LwpTOGScopeUseLeader (1847)~~

~~\$LwpTOGScopeUsePageNumber (1849)~~

~~\$LwpTOGScopeUseText (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:

D L M

G o t S

A u W

/ s or

R M d

F a fo

I n r

u W

scin

ri d

pt o

2. w

x s

1.

o

D L M

~~I ot S~~

~~F u W~~

~~s or~~

~~Q d~~

~~rgfo~~

~~a r~~

~~ni W~~

~~z in~~

~~erd~~

~~1. e~~

~~x w~~

~~s~~

~~2.~~

~~0~~

~~D L M~~

~~is ot S~~

~~plu W~~

~~a s or~~

~~y Wd~~

~~Worfo~~

~~ritd r~~

~~e PrW~~

~~e in~~

~~d~~

~~e~~

~~w~~

~~s~~

~~6.~~

~~0~~

~~H L M~~

~~I ot S~~

~~Mu W~~

~~L s or~~

~~Wd~~

~~orfo~~

~~d~~ r

PrW

~~e~~ in

S d

m ~~e~~

~~a~~ r w

t s

M 9

a 5

st 7.

~~e~~ r 0

L M M

~~ot~~ S S

u E W

~~s~~ ~~x~~ ~~e~~ ~~o~~ ~~r~~

1 ~~e~~ ~~d~~

= P

2 a

= ~~d~~

3 1.

0

L M 0

~~ot~~ S ffi

u E e

~~s~~ ~~x~~ e

1 ~~e~~ W

= 3 rit

2 0 ~~e~~ r

= 4.

3 5.

~~f~~ 0 6

r

0

S

/2

L M Ri

otS-e

u E h-

s-xcT

1 el e

- 4. xt

2 0 F

- or

3 m

R at

3 (

R

I

E)

L M S

otS-A

u E M

s-xcN

1 el A-

- 5. W

2 0 or

- d

3-

R

4.

5

L M W

otS-or

u E d

s-xcP

1 el er

- 7. fe

2 0 et

- 5.

3 0

R

6

L MW

otS or

u Wd

s in P

A d er

mo fe

i w et

P s 5.

r W1

e rit

e

3.

x

L MW

otS or

u Wd

s or P

A d er

mfo fe

i r et

P D 6.

r O x

e S

3.3.

x 4.

M5.

a 6

er

e

L MW

otS or

u Wd

s or St

~~A d ar
mfo 2
i r 0
P O 0
r S/O
e-2 R
3. 3
x
S
ty
le
s~~

Password

Use this parameter to provide a 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_DROPCA

PCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_GLAS

S;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWG

ONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TAB

LECONTAINER_CLASS;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.

ValEff
ue ect
\$L Th
wp e
Go met
To hod
Lo will
cat mo
ion ve
Pa the
ge ins
{3 erti
72 on
} poi
nt
to a
spe
cifi
e
pag
e
con

~~tain~~
~~er~~
~~wit~~
~~hin~~
~~the~~
~~cur~~
~~ren~~
~~t~~
~~con~~
~~text~~
~~of~~
~~the~~
~~doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wp e~~
~~Go met~~
~~To hod~~
~~Lo will~~
~~eat mo~~
~~ion ve~~
~~He the~~
~~ad ins~~
~~er erti~~
~~{3 on~~
~~73 poi~~
~~} nt~~
~~to a~~
~~spe~~
~~cifi~~
~~e~~
~~hea~~
~~der~~
~~con~~

tain
er
wit
hin
the
cur
ren
t
con
text
of
the
doc
um
ent.

\$L Th
wp e
Go met
To hod
Lo will
eat mo
ion ve
Fo the
oteins
r erti
{3 on
74 poi
} nt
to a
spe
cifi
e
foot
er
con

tain
er
wit
hin
the
cur
ren
t
con
text
of
the
doc
um
ent.
\$L Th
wp e
Go met
To hod
Lo will
eat mo
ion ve
Fr the
amins
e erti
{3 on
75 poi
} nt
to a
spe
cifi
e
fra
me
con

tain
er
wit
hin
the
cur
ren
t
con
text
of
the
doc
um
ent.

\$L Th
wp e
Go met
To hod
Lo will
eat mo
ion ve
Di the
vis ins
ion erti
{3 on
76 poi
} nt
to-a
spe
cifi
e
divi
sio
fr

con

tain

er

wit

hin

the

cur

ren

t

con

text

of

the

doc

um

ent.

\$L Th

wp e

Go met

To hod

Lo will

cat mo

ion ve

Exithe

t ins

{3 erti

77 on

} poi

nt

out

of a

spe

cifi

e

con

tain
er
wit
hin
the
cur
ren
t
con
text
of
the
doc
um
ent.

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_GLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 WPAplication 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:

AHP

li aa

gf g

nde

SN

pu

am

eb

ee

r

Al P

nno

edw

hee

onr

rt F

e i

d e

F l

r d

a

m

e

Bl R

onu

odi

kee

~~M~~r

a M

r a

k r

k

GLS

oae

lyft

uoH

muy

nt p

B h

r e

e n

a

k

DLS

Di p

Ena

ee

Bi

rn

eg

a

k

DNT

oaa

et b

Vel

a e

ri

a

b

l

e

FPT

oaO

ogG

te

nB

of

te

ea

Mk

a

r

k

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_BASETABLE_CLASS;H_FOOTNOTETABLE_CLASS;H_GLOSSARY_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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.

ValEff

ue ect

\$L Th

wp e

Ce Cel

#O IOb

bje ject

ctT Typ

yp e

e par

(11 am

7) ete

r

mu

st

be

this

val

ue.

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.

GGGR
uuue
ffffs
sssu!
ooo!
ffffo

loa lof
e t e t
a t a h
t h t e
e e e G
d e d o
inn inT
t d lao
h o stT
e f e a
lat elbi
sth L e
e e o G
elt f el
L e t L
o xth m
f w e e
a i t t
r hia h
o n blo
wt e d
h :
e
e
el
l
NDDG
oo o u
e e r
s s s
n'n'o
t t r
mmm
a a o
t t v

e e e

f r s

t

o

t

h

e

n

e

x t

e

e l

t

Y N D G

e o o u

s e r

s s

n' o

t r

m m

a o

t t y

e e

f s

t

o

t

h

e

e

n

d

o

f

t

h
e
t
e
xt
w
it
hi
n
t
h
e
e
el
t

YYNG

e e o u

s s f

s
o
r
m
o
v
e
s
t
o
t
h
e
n
e
xt
e

e/

t

Y Y Y N

e e e e

s s s w

f

e

w

i s

a

p

p

e

n

d

e

d

t

e

t

h

e

t

a

b/

e

=

a

n

d

t

h

e

e

u

r
s
e
r
m
e
y
e
s
t
e
t
h
e
fi
r
st
e
el
t
e
f
t
h
e
n
e
w
r
e
w
i

Word Pro: GoTo method

{button ,AL('H_CLICKHERE_CLASS;H_FOOTNOTE_CLASS;H_MARKER_CLASS;H_

POWERFIELD_CLASS;H_RUBYMARKER_CLASS;H_TABLEMARKER_CLASS;H_TE

XMARKER_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CLICKHE

RE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;

H_DIVISION_CLASS;H_DROPGLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;

H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAY

OUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLA

YOUT_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_S

UPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEH

EADINGLAYOUT_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_DOCWIN

DOW_CLASS;H_DROPDOWNCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_

NOTECONTAINER_CLASS;H_NOTELAYOUT_CLASS;H_PAGECONTAINER_CLASS;

H_PARALLELCOLSCONTAINER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCON

TAINER_CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_GL

ASS;H_SUPERTABLECONTAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLE

ONLYCONT_CLASS;H_WPAPPLICATION_CLASS'.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 lblPasswordHelp.



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 developed 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\SYSTEMWINMM.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 HELPBTN.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 Frame1.

Set MyFrame = Bind("!Body:Frame1")

' 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 of 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 EmbedOGX

' 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\MPLAYER.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\MPLAYER 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 -l

' specifies that WinHelp search for the specified context ID.

HelpReturnValue = Shell("WINHLP32.EXE -l 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 "OGX"

' * _____ 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 Replacelt 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.

—Replacelt = ReplaceTextString(MyInput, "two", "Three")

—' Direct the results of the function call to the IDE output panel.

—Print Replacelt

—' 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 colums 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 colums 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: SetTOGProperties 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 redisplay 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 IconPallet As String

Dim MsgStr As String

Dim IconMgr As IconBarManager

IconPallet = "Comment Tools"

Set IconMgr = .ApplicationWindow.IconBarManager

'Set icon pallet to show in its context

IconMgr.IconBars(IconPallet).ShowInContext = True

'This will force a redraw of IconBars

IconMgr.ShowIconBars

MsgStr = "|" & IconPallet

~~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~~

~~IconMgr.IconBars(IconPallet).ShowInContext = False~~

~~IconMgr.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 IconPallet As String

Dim MsgStr As String

Dim IconMgr As IconBarManager

IconPallet = "Comment Tools"

Set IconMgr = .ApplicationWindow.IconBarManager

' Set the context and show the bar

IconMgr.IconBars(IconPallet).ShowInContext = True

IconMgr.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

IconMgr.IconBars(IcnPallet).ShowInContext = False

IconMgr.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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_G
LASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_CLASS;H_FOOTERLAYOU
T_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FR
AMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_L
AYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPH
HBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_R
UBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELA
YOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_T
OC SUPERTABLELAYOUT_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOU

T_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H

_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_GLA

SS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUP

LAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADE

RLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_

CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW

LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_GLAS

S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL

AYOUT_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_CLAS

S',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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_KEEPPWITHNEXT_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.

L L
a a
n n
g g
u u
a a
g g
e e
G :
e G
d e
e d
s e
B s
Y :
N b
a y
m :
e N
u
m
b
-

e

r

A 10S

F 0Y

R 7S

IK 8T

A E

A M

N

S

A 11A

L 00R

B 52A

A 24B!

N 1C

IA S

N Y

RI

A

A 11A

M 00R

E 32A

R 35B!

I C

G

A

N

A 31S

R 00P

A 72A

B 35N!

G 0S

E H

G P

Y E

~~P R~~

~~T U~~

~~A 61G~~

~~R 10Z~~

~~A 42E~~

~~B159C~~

~~G H~~

~~M~~

~~O~~

~~R~~

~~O~~

~~G~~

~~G~~

~~O~~

~~A 11H~~

~~R 00E~~

~~A 23B~~

~~B157R~~

~~G E~~

~~W~~

~~A 11H~~

~~R 50U~~

~~A 33N~~

~~B168G~~

~~G 1A~~

~~B RI~~

~~A A~~

~~H N~~

~~R~~

~~AI~~

~~N~~

~~A 11D~~

~~R 10A~~

~~A 23N~~

~~B168S~~

G 5 H

J

Q

R

D

A

N

A 11G

R 66E

A 33L

B189A

G 5 N

Q DI

A G

F

A

R

A 211F

R 00A

A 44L

B190A

G N

I

R

A

Q

A 11B

R 00U

A 22L

B146G

G 1 A

S RI

Y A

R N

IA

A 71G

R 10E

A 63R

BI91M

G A

I N

U

N

IS

IA

A 11G

R 20R

A 23E

BI82E

G 9 K

L

E

B

A

N

O

N

A 41C

R 00A

A 92T

BI77A

G L

LI A

B N

Y

A

A 81A

R 10M

A 98E

BI33R1

G G

O A

M N

A

N

A 11S

R 30P

A 33A

BI 14NI

G 3S

K H

U

W

AI

I

A 51G

R 10HI

A 22N

BI 18E

G S

A E

L I

G R

E A

R DI

IA TI

O

N

A

E

A 91FI

R 20N

A 13NI

BI 7S

G H

Y
E
M
E
N
A 11F
R 40R
A 33E
B136N
G 7 G
U H
A
E
A 31J
U 00A
S 84P
I 11A
R N
A E
L1 S
A E
N
B 11K
A 000
S 64R
Q 92E
U A
E N
B 11D
R 00U
A 44F
Z163G
L1 H
A
N

B 21N

R 000

IT 51R

IS 74W

H E

G

IA

N

B 31P

R 500

IT 84L

IS 45S

H 9 H

M

E

D

IZ

E

B 31B

R 40R

M 84A

E 26Z

D 5 L

I A

G N

A

L

B 11R

U 00H

L 24A

G 67E

A F

R O

IA R

N O

M
A
N
B 11R
Y 000
E 51M
L 98A
O NI
R A
U N
S
SI
A
N
G 11R
A 00U
I 21S
A 79SI
L A
A N
N
G 21S
H 00W
I 56E
N 23DI
E S
S H
E
P
R
G
H
I
N
A

G 31G

H 00R

I 75Q

N 00A

E 7I

S A

E N

H

Q

N

G

K

Q

N

G

G 41S

H 10L

I 05Q

N 01V

E A

S K

E

S/

N

G

A

P

Q

R

E

G 11A

H 00L

I 25B

N 02A

E NI

S A
E N
I
R
A
D
H
I
O
N
A
L
G 21F
R 00H
O 75A1
A 44
H
A
N
G
Y
R
IL
LI
G
G 31F
R 00U
O 95R
A 85K1
H S
A H
N
S
E
R

~~BI~~
A
N
G 11U
R 00R
O 55D
A 06U
H
A
N
G 11IN
Z 00D
E 25O
G 97N
H E
SI
A
N
D 11U
A 00K
N 36R
IS08AI
H NI
A
N
D 21B
U 00Y
I 66E
G 79L
H O
B R
E U
L S
G SI
IA A

~~N N~~
~~D 11S~~
~~U 00L~~
~~F 46O~~
~~G 30V~~
~~H E~~
~~N~~
~~E~~
~~E 41E~~
~~N 10S~~
~~G 06T~~
~~LI 51O~~
~~S NI~~
~~H A~~
~~G N~~
~~A~~
~~N~~
~~A~~
~~D~~
~~IA~~
~~N~~
~~E 51L~~
~~N 10A~~
~~G 26T~~
~~LI 92VI~~
~~S A~~
~~H N~~
~~N~~
~~E~~
~~W~~
~~Z~~
~~E~~
~~A~~
~~L~~
~~A~~

N
D
E 61LI
N 10T
G 56H
LI 33U
S A
H NI
I A
R N
E
L
A
N
D
E 71F
N 10A
G 76R
LI 75SI
S
H
S
A
F
R
I
G
A
E 81B
N 20A
G 06S
LI 19Q
S U
H E
J

A
M
A/
G
A
E 91S
N 200
G 27R
LI 50BI
S A
H N
G
A
R
R
IB
E
A
N
E 11A
S 00F
I 67RI
O 18K
N A
IA A
N N
S
F 11F
A 00A
E 88E
R 00R
O O
E E
S S
E E

F 11A

A 01R

R 02A

S 150B1

5C

J

O

R

D

A

N

F 11S

N 01P

N 32A

I S 57N1

H 4S

H

A

R

G

E

N

H

N

A

F 21A

R 02R

E 02A

N 08B1

G 9C

H 1

B E

E B

L A

G N

IA O
N N
F 31S
R 02P
E 82A
N 49N1
G 8S
H H
G E
A G
N U
A A
D D
IA O
N R
F 41A
R 13R
E 08A
N 81B1
G 3C
H K
S U
W W
IS AI
S T
F 51S
R 13P
E 38A
N 22N1
G 2S
H H
L G
U HI
X L
E E

M
B
O
U
R
G
F 11A
R 01R
E 33A
N 63B1
G 7C
H U
A
E
G 21S
E 06P
R 53A
M 59N1
A 4S
N H
S B
W O
IS H
S VI
A
G 31S
E 01P
R 73A
M 94N1
A 6S
N H
A U
U R
S U
I G

R U
IA A
N Y
G 11A
E 06R
R 33A
M 16B1
A 1C
N B
A
H
R
AI
N
G 51S
E 15P
R 23A
M 77N1
A 0S
N H
LI P
E A
G R
H A
I G
E U
N A
S Y
I
E1
N
G 41A
E 16R
R 03A
M 38B1

~~A 5C~~
~~N Q~~
~~L A~~
~~U I~~
~~X A~~
~~E R~~
M
B
O
U
R
G
G ~~12A~~
R ~~00R~~
E ~~34A~~
E ~~29B1~~
K ~~G~~
IR
A
Q
H ~~12G~~
E ~~00H1~~
B ~~36N~~
R ~~72E~~
E S
W ~~E~~
P
R
G
H
N
A
H ~~12G~~
U ~~00E~~
N ~~36R~~

G 85M

A A

R N

IA S

N W

IS

S

I 12B

G 00RI

E 36TI

L 97S

A H

N

D

I

G

I 12S

N 00P

D 55A

O 78NI

N S

E H

SI M

A E

N XI

G

A

N

IT 22F

A 00R

LI 66E

A 40N

N G

S H

W B

IS E
S E
G
IA
N
IT 12T
A 00A
LI 46L
A 01A
N N
S
W
IS
S
J 12K
A 000
P 46R
A 16E
N A
E N
S J
E O
H
A
B
K 12D
O 00U
R 46T
E 27C
A H
N B
E
E
G
IA

N
K ~~2N~~
O ~~00Y~~
R ~~00N~~
E ~~080~~
A R
N S
J K
O
H
A
B
L 12P
A ~~000~~
I ~~67R~~
V ~~120F~~
A U
N G
U
E
S
E
L 12C
I ~~00R~~
H ~~67O~~
U ~~34A~~
A I
N A
I N
N G
Y
R
E
H
G

~~M 33A~~

~~E 30R~~

~~D 87A~~

~~I 08B1~~

~~G 1 G~~

~~A E~~

~~L G~~

~~Y~~

~~P~~

~~I~~

~~N 13C~~

~~O 00H1~~

~~R 47N~~

~~W 46E~~

~~E S~~

~~G E~~

~~IA H~~

~~N O~~

~~N~~

~~G~~

~~K~~

~~O~~

~~N~~

~~G~~

~~N 23G~~

~~Y 00E~~

~~N 67R~~

~~O 89M~~

~~R A~~

~~S N~~

~~K A~~

~~U~~

~~S~~

~~I~~

~~RI~~

A
N
P 13A
Q 00U
LH 48S
S 51T
H R
A
LH
A
N
P 23S
Q 00P
R 78A
T 00N
U S
G H
U M
E O
S D
E E
R
N
R 13F
H 00R
A 48E
E 74N
T G
Q H
R G
Q A
M N
A A
N DI
A

N
R 13V
Q 000
M 490
A 81R
N K
IA E
N U
R
R 13C
U 00R
S 490
SI98A
A TI
N A
N
S
E
R
BI
A
N
R 33M
U 33E
S 88DI
SI10C
A 71A
N E
I
Q
S 13R
L 00U
Q 58S
V 11SI
A 7A

~~K~~ ~~NH~~
~~O~~
~~S 13B~~
~~L 04R~~
~~O 68M~~
~~V 02E~~
~~E 5DI~~
~~N C~~
~~E A~~
~~L~~
~~S 13B~~
~~O 05R1~~
~~R 78H~~
~~B104S~~
~~A 9H~~
~~N M~~
~~E~~
~~DI~~
~~Z~~
~~E~~
~~S 14C~~
~~P 21H1~~
~~A 20N~~
~~N 90E~~
~~IS8 S~~
~~H E~~
~~E SI~~
~~G N~~
~~U G~~
~~A A~~
~~D P~~
~~O O~~
~~R R~~
~~E~~
~~S 76S~~

P 11P
A 73A
N 80N
IS S
H H
D G
O O
M S
I I
N A
I RI
G G
A A
N
S 34F
P 01R
A 80E
N 28N
IS G
H H
M S
O W
D IS
E S
R
N
S 16U
P 15N
A 25V
N 73E
IS 46R
H S
A A
R E
G

E
N
H
N
A
S ~~6~~S
P ~~1~~P
A ~~5~~A
N ~~4~~N
IS S
H H
P G
A U
N A
A I
M E
A M
A
E
A
S ~~1~~S
P ~~0~~P
A ~~2~~A
N ~~5~~N
IS S
H H
P P
E A
R N
U A
M
A
S ~~8~~S
P ~~2~~P
A ~~0~~A

~~N 23M~~

~~IS A~~

~~H N~~

~~V E~~

~~E U~~

~~N X~~

~~E E~~

~~Z M~~

~~U B~~

~~E O~~

~~L U~~

~~A R~~

~~G~~

~~S 27A~~

~~P 01R~~

~~A 56A~~

~~N 88B1~~

~~IS C~~

~~H I~~

~~M U~~

~~E N1~~

~~XI SI~~

~~G A~~

~~A~~

~~N~~

~~S 14E~~

~~P 31N~~

~~A 30G~~

~~N 251~~

~~IS2 S~~

~~H H~~

~~G G~~

~~H A~~

~~IL N~~

~~E A~~

DI
A
N
S 96A
P 21R
A 24A
N 66BI
IS C
H M
G O
O R
L O
O C
M C
BI O
A
S 15A
P 61R
A 32A
N 91BI
IS4 C
H A
B L
O G
LI E
VI RI
A A
S 14A
P 40R
A 39A
N 47BI
IS6 C
H LI
U B
R Y

U A
G
U
A
Y
S ~~5G~~
P ~~1E~~
A ~~3R~~
N ~~07M~~
IS A
H N
G H
O E
S G
F H
A F
R E
I N
G S
A F
E
N
S ~~15E~~
P ~~01N~~
A ~~3G~~
N ~~4L~~
IS S
H H
N
E
W
Z
E
A
E

A
N
D
S ~~4F~~
P ~~11R~~
A ~~08E~~
N ~~62N~~
IS C
H H
G L
U U
A X
I E
E M
M B
A O
L U
A R
G
S ~~16E~~
P ~~51N~~
A ~~36G~~
N ~~73L~~
ISO S
H H
P IR
A E
R L
A A
G N
U D
A
Y
S ~~17E~~
WGIN

~~E 57G~~

~~D 37L~~

~~IS S~~

~~H H~~

~~S~~

~~A~~

~~F~~

~~R~~

~~G~~

~~A~~

~~S 07S~~

~~Y 1P~~

~~S 7A~~

~~I 8N~~

~~E S~~

~~M H~~

~~D~~

~~O~~

~~M~~

~~IN~~

~~IG~~

~~A~~

~~N~~

~~I 18A~~

~~H 01R~~

~~A 50A~~

~~43B~~

~~G~~

~~O~~

~~M~~

~~A~~

~~N~~

~~I 18E~~

~~U 02N~~

~~R 50G~~

K151L1

S S

H H

J

A

M

AI

G

A

U 18S

K 02P

R 50A

AI 82NI

N S

IA H

N V

E

N

E

Z

U

E

L

A

U 60A

N 52R

IV51A

E 37BI

R 5C

S Y

A E

L M

E

N

U 19E

~~R 02N~~

~~D 52G~~

~~U 65L~~

~~S~~

~~H~~

~~G~~

~~A~~

~~R~~

~~Rl~~

~~B~~

~~E~~

~~A~~

~~N~~

~~V 30S~~

~~O 02P~~

~~O 02A~~

~~R 16Nl~~

~~K S~~

~~E H~~

~~U G~~

~~R O~~

~~L~~

~~O~~

~~M~~

~~Bl~~

~~A~~

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

keywordsvalue = [objectreference].Keywords

[objectreference].Keywords = keywordsvalue

Legal values

The legal values for this property are determined by its data type. For 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

lastgraphictypevalue = [objectreference].LastGraphicType

[objectreference].LastGraphicType = lastgraphictypevalue

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_LASTUSEDDEDATEFORMULA_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_CLA

SS;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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ValEff

ue ect

\$L Set

wp ting

Le this

ad val

er ue

~~De pre~~
~~tTy ven~~
~~pe ts~~
~~No lea~~
~~ne der~~
~~{5 dot~~
~~26 s~~
} fro
m
bei
ng
ins
ert
ed
in a
lay
out
obj
ect.
\$L Set
wp ting
Le this
ad val
er ue
Do ins
tTy erts
pe dot
Do s
ts as
{5 the
27 typ
} e of
lea
der
dot

use

d in

a

lay

out

obj

ect.

\$L Set

wp ting

Le this

ad val

er ue

Do ins

tTyerts

pe das

Da hes

sh as

es the

(5 typ

28 e of

) lea

der

dot

use

d in

a

lay

out

obj

ect.

\$L Set

wp ting

Le this

ad val

er ue

~~De ins
tTyerts
pe und
Un ers
de cor
rsc es
or as
es the
{5 typ
29 e of
} lea
der
dot
use
d in
a
lay
out
obj
ect.~~

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_ENDNOTEDIVISIONNU

M_CLASS:H_ENDNOTEDOCNUM_CLASS:H_FOOTNOTENUMBERING_CLASS:H_F

OOTNOTENUMOPT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_STATUS

BAR_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ValEff
ue ect
0 Set
ting
this
val
ue
pla
ces
a

line
are
und
the
ma
rgin
s of
a
lay
out
obj
ect.

1 Set
ting
this
val
ue
pla
ces
a
line
in
the
mid
dle
of a
lay
out
obj
ect.

2 Set
ting
this
val
ue

pla
ces
a
line
are
und
the
edg
e of
a
lay
out
obj
ect.

3 Set
ting
this
val
ue
allo
ws
you
to
spe
cify
the
dist
anc
e of
a
line
fro
m
the
edg

e of
a
lay
out
obj
ect.

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

ValEff

ue ect

\$L Set

wp ting

Ta this

ble val

Mi ue

xAlallo

lmi ws

xe you

d to

(1 ass

89 ign

3) a

diff

ere

nt

line

styl

e to

eac

h

line

of a

tabl

e

cell

z

\$L Set

wp ting

Ta this

ble val

Mi ue

xB allo

ott ws

omyou

~~mi to~~
~~xe ass~~
~~d ign~~
~~(1 a~~
~~89 line~~
~~0) styl~~
~~e to~~
~~the~~
~~bott~~
~~om~~
~~line~~
~~of a~~
~~tabl~~
~~e~~
~~cell~~
~~that~~
~~is~~
~~diff~~
~~ere~~
~~nt~~
~~fro~~
~~m~~
~~the~~
~~line~~
~~styl~~
~~e of~~
~~the~~
~~top.~~
~~righ~~
~~t.~~
~~and~~
~~left~~
~~line~~
~~of a~~
~~tabl~~

e-

cell

z

\$L Set

wp ting

Ta this

ble val

Mi ue-

xL allo

eft ws-

mi you

xe to-

d ass

(1 ign-

89 a-

1) line

styl

e to

the-

left-

line

of a

tabl

e-

cell

that

is-

diff

ere

nt

fro

m-

the-

line

styl

e of
the
top
right
t
and
bott
om
line
of a
tabl
e
cell

z

\$L Set
wp ting
Ta this
ble val
Mi ue
xR allo
igh ws
tmi you
xe to
d ass
(1 ign
89 a
2) line
styl
e to
the
right
t
line
of a
tabl

~~e-~~
~~cell~~
~~that~~
~~is~~
~~diff~~
~~ere~~
~~nt~~
~~fro~~
~~m~~
~~the~~
~~line~~
~~styl~~
~~e-of~~
~~the~~
~~top~~
~~bott~~
~~om~~
~~and~~
~~left~~
~~line~~
~~of a~~
~~tabl~~
~~e-~~
~~cell~~

~~z~~

~~\$L Set~~
~~wp ting~~
~~Ta this~~
~~ble val~~
~~Mi ue-~~
~~xT allo~~
~~op ws-~~
~~mi you~~
~~xe to-~~
~~d- ass~~

~~(1 ign~~
~~88 a-~~
9) ~~line~~
~~styl~~
~~e to~~
~~the~~
~~top~~
~~line~~
~~of a~~
~~tabl~~
~~e~~
~~cell~~
~~that~~
~~is~~
~~diff~~
~~ere~~
~~nt~~
~~fro~~
~~m~~
~~the~~
~~line~~
~~styl~~
~~e of~~
~~the~~
~~bott~~
~~om,~~
~~righ~~
~~t,~~
~~and~~
~~left~~
~~line~~
~~of a~~
~~tabl~~
~~e~~
~~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_LINESSPACEDEVERYNTHUNIT_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

linevalidvalue = [objectreference].LineValid

[objectreference].LineValid = linevalidvalue

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

~~ValEffe~~
~~ue ct~~

~~Lw Pre~~
~~pE vent~~
~~dit s-an~~
~~Lo edit~~
~~ck or~~
~~sN fro~~
~~oG m-~~
~~op cop~~
~~yA ying~~
~~nd any~~
~~No part~~
~~Sa of~~
~~ve the-~~
~~as doc~~
~~{& ume~~
~~H4 nt to~~
~~} the-~~
~~Clip~~
~~boa~~
~~rd-~~
~~usin~~
~~g-~~
~~dra~~
~~g-~~
~~and~~
~~dro~~
~~p-~~
~~on-~~
~~any~~
~~part~~

~~of~~
~~the~~
~~doc~~
~~ume~~
~~nt.~~
~~or~~
~~usin~~
~~g~~
~~File~~

=
~~Sav~~
~~e~~
~~As~~
~~to~~
~~sav~~
~~e~~
~~the~~
~~doc~~
~~ume~~
~~nt~~
~~with~~
~~a~~
~~new~~
~~nam~~
~~e.~~

~~Lw Pre~~
~~pE vent~~
~~dit s-an~~
~~Lo edit~~
~~ck or~~
~~sN fro~~
~~oE m~~
~~dit mod~~
~~Na ifyin~~
~~meg~~

~~dS any-
tyl nam
es ed
& styl
H2 es
) in-
the
doc
ume
nt.~~

~~Lw Allo
pE ws
dit an-
Lo edit
ek or-
sN to-
oL mak
oc e-
ks any-
& cha
H0 nge
) s-to-
a-
doc
ume
nt.~~

~~Lw Pre
pE vent
dit s-an
Lo edit
ek or-
sN fro
oP m-
rintprint
ing ing-~~

~~(& the~~
~~H8 doc~~
~~) ume~~
~~nt.~~

~~Lw Pre~~
~~pE vent~~
~~dit s-an~~
~~Lo edit~~
~~ck or~~
~~sN fro~~
~~oV m-~~
~~erscrea~~
~~ion ting-~~

~~Or new~~
~~Re vers~~
~~vie ions~~

~~w or~~
~~(& seei~~
~~H1 ng-~~

~~) any-~~
~~oth~~
~~er-~~
~~vers~~
~~ions~~

~~==~~

~~Lw Allo~~
~~pE ws-~~
~~dit an-~~
~~Lo edit~~
~~ck or~~
~~sR to-~~
~~ev edit-~~
~~mathe-~~
~~rk doc~~
~~On ume~~

~~ly nt.~~
~~(& Ho~~
~~H1 wev~~
~~0) er.~~
~~all~~
~~edit~~
~~s~~
~~app~~
~~ear~~
~~as~~
~~mar~~
~~kup~~
~~s.~~
~~The~~
~~edit~~
~~or~~
~~can~~
~~not~~
~~acc~~
~~ept~~
~~or~~
~~reje~~
~~ct~~
~~edit~~
~~s~~
~~usin~~
~~g~~
~~the~~
~~Rev~~
~~iew~~
~~bar.~~
~~For~~
~~exa~~
~~mpl~~
~~e.~~

the

n-

the

edit

or

dele

tes-

text.

it

doe

s-

not

disa

ppe

ar

fro

m-

the

doc

ume

nt.

but

app

ears

with

the

edit

or's-

dele

ted-

text

attri

but

es:

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_APPLIC

ATIONWINDOW_CLASS;H_DOCWINDOW_CLASS';0)} See list of classes

{button ,AL('H_LWPBASECTRL_LWPCUSTOMDIALOG_HWND_PROPERTY_EXSCR

IPT';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_MACRONAME_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_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

$\text{maintainaspectratiovalue} = [\text{objectreference}].\text{MaintainAspectRatio}$

$[\text{objectreference}].\text{MaintainAspectRatio} = \text{maintainaspectratiovalue}$

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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACT

ERBORDER_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT

_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERL

AYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;

H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_GLAS

S;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARA

GRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS

;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTAB

LELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;

H_TOCSUPERTABLELAYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACT

ERBORDER_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT

_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERL

AYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;

H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_GLAS

S;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARA

GRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS

;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTAB

LELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;

H_TOCSUPERTABLELAYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACT

ERBORDER_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT

_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERL

AYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;

H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_GLAS

S;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARA

GRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS

;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTAB

LELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;

H_TOCSUPERTABLELAYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CHARACT

ERBORDER_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT

_CLASS;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERL

AYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;

H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_GLAS

S;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARA

GRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS

;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTAB

LELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;

H_TOCSUPERTABLELAYOUT_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

margin_topvalue = [objectreference].MarginTop

[objectreference].MarginTop = margin_topvalue

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_MAXHORZPANEDISTANCE_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_NUMBEREVERYNTHLINE_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_TEXT

MARKER_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

numcharsinpsaragraphvalue = [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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

fbutton ,AL('H_BASETABLE_CLASS;H_GELLGROUPLAYOUT_CLASS;H_GELLAYO
UT_CLASS;H_CLICKHERE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNE
CTEDLAYOUT_CLASS;H_DOCWINDOW_CLASS;H_DROPGLAYOUT_CLASS;H_
ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_C
LASS;H_FOOTNOTETABLE_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMEL
AYOUT_CLASS;H_GLOSSARY_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAY
OUT_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_CLA
SS;H_RUBYLAYOUT_CLASS;H_RUBYSMARKER_CLASS;H_SUPERTABLEGROUPLA

YOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADI

NG_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TABL

EMARKER_CLASS;H_TEXTMARKER_CLASS;H_TOCSUPERTABLELAYOUT_CLAS

S;H_WINVIEWPREFS_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_DROPDOWNCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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\FootnoteSepOpt

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\FootnoteSepOpt

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

WPAApplication 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

WPAApplication 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.

WPApplication.TempFoundry

TempFoundry is a property on WPApplication. 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 WPAApplication object.

Base Classes

BaseObject\BaseContainer

Derived Classes

NoteContainer

RubyContainer

Contained by

WPAApplication 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

BaseObjectLayout

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

BaseObjectContent

Derived Classes

Graphic

OleObject

Contained by

WPAApplication 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 indention 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 Join Type

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

TOGSuperTableLayout

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

WPAApplication in the Layout Property

WPAApplication 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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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

DefaultLeftColumnName AS String

DefaultRightColumnName 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 WPAApplication

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 WPAApplication

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

DoneWithRightMouseMenu

EnumerateChartLinks

GetCopyFormatCategories

GetLinkDisplayNameFileLength

GetLinkName

GetLinkSourceName

GetPasteFormatCategories

GetRightMouseMenu

GetSource

GetUserClassNameFull

GetUserClassNameShort

ImportPicture

Mark

ProcessAccelKey

SaveData

SaveSnapshot

SetLinkSource

TheoreticalScaledSize

Undo

UpdateLink

Events

None

Word Pro: GroupLayoutCollection class members

Properties

Application AS WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 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: 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 WPAApplication

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 WPAApplication

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 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: 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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 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: MenuItemCollection class members

Properties

Application AS WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPAApplication

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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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

glossarydatapathvalue = [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

glossaryvalue = [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 WPApplication 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 WPApplication 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 WPApplication 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_GRAMMARTOPTIONS_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 WPAApplication 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 WPAApplication 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 WPAApplication 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 WPApplication 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 WPApplication 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 WPApplication 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_IDENTIFIER_COLOR_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_C

LASS;H_TEXT_CLASS;H_TEXTMARKER_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-

ferences 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_CLAS

S;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_CLICKHER

E_CLASS;H_DIVISIONOPTIONS_CLASS;H_FORMATPREFERENCES_CLASS;H_LA

NGUAGE_CLASS;H_PARAGRAPHSTYLE_CLASS;H_TEXT_CLASS;H_TEXTMARKE

R_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

layoutvalue = [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 WPAApplication 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 WPAApplication 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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_CLASS;H_FOOTERLAYOU
T_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FR
AMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_L
AYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_PARAGRAPH
HBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_R
UBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELA
YOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_T
OC SUPERTABLELAYOUT_CLASS;H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOU

T_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H

_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FONTMETRICS_GLA

SS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUP

LAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADE

RLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_

CLASS;H_PARAGRAPHBORDER_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW

LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_GLAS

S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL

AYOUT_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_BASETABLE_CLASS;H_CELLCONTAIN
ER_CLASS;H_CLICKHERE_CLASS;H_DROPCAPCONTAINER_CLASS;H_FOOTNO
TETABLE_CLASS;H_FRAMECONTAINER_CLASS;H_GLOSSARY_CLASS;H_MARK
R_CLASS;H_NOTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLE
LCOLSCONTAINER_CLASS;H_PARALLELOLUMNS_CLASS;H_POWERFIELD_GL
ASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_RUBYSMARKER_
CLASS;H_SUBPAGECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_S
UPERTABLECONTAINER_CLASS;H_TABLE_CLASS;H_TABLECONTAINER_CLASS;
H_TABLEHEADING_CLASS;H_TABLEMARKER_CLASS;H_TABLEONLYCONT_GLAS
S;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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

macropathsvalue = [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 WPApplication 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 WPApplication 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 WPApplication 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_CLAS

S;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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 WPAplication 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 WPAplication 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 WPAplication 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 WPAplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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_PAPER_NAMES_PROPERTY_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_CLAS

S;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 WPAApplication 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 WPAApplication 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 WPAApplication 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_TEXT

MARKER_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_DROP

PCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_GLAS

S;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWG

ONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TAB

LECONTAINER_CLASS;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_CLAS

S: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: RightMouseMenus 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

rightmousemenusvalue = [objectreference].RightMouseMenus

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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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

rowLayoutsvalue = [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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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 WPApplication 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

snapshotsaveoptionsvalue = [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: HideIconBars 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].HideIconBars()

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 Us
yo e-
u this
areval
im ue-
perfor
tin the
g Fil
thi eF
s or
typ ma
e t
of par

graam

phi ete

e: r:

W .w

or mf

dP

re

W

M

F

W .ep

or s

dP

re

E

P

S

W .b

or mp

dP

re

Pa

int

W .tif

or

dP

rol

m

ag

e

W .sd

or w

dP

re

Dr

aw

Bo :bo

xe x

s

W :te

or x

dP

re

Eq

n

W :pe

or x

dP

re

P

G

X

W :gif

or

dP

re

G!

F

G :eg

G m

M

H :plt

P

G

L

P :pe

G d

D

D :dr

R w

W
G .cd
D f
R
Pl .pi
G e
Im .w
p pg
W
P
G
Ex .sd
pS w
D
W
z
W
P
G
Ex .w
p mf
W
M
F2
W
P
G
Ex .b
pB mp
M
P2
W
P
G
Im .w

p p2

W

P2

Ex .sd

pS w

D

W

2

W

P2

Ex .w

p mf

W

M

F2

W

P2

Ex .b

pB mp

M

P2

W

P2

Im .jp

pJ g

P

G

Ex .b

pJ mp

P

G

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_INSERTGLICKHERE_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_INSERTGLICKHERE_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.

Thi ...
s yi
for el
mul d
a... s
th
is
re
s
ut
t
"Cr 1/
eat 8/
eD 9
ate 7
%D
a"
"Cr 0
eat 9:
eD 0
ate 1
%T A

3" M
"Gr S
eat at
eD ur
ate d
%F ay
LW :
eek 1/
day.8/
M/ 9
DAY 7.
Y. 0
HH: 9:
mm 0
am 1
pm" A
M
"Gr S
eat A
eD T
ate U
%F R
GWD
eek A
day.Y.
M/ 1/
DAY 8/
Y. 9
HH: 7.
mm 0
am 9:
pm" 0
1
A
M

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.

F ...
hi yi
s el
d ds
at thi
e. s:
...
N Th
e e-
w eu
t rr
en
t
sy
st
e
m-
da
te-
an
d
ti
m
e.
G Th

ree-

at da

e te-

D an

at d-

e ti

m

e-

th

e-

do

cu

m

en

t-

w

as

fir

st-

sa

ve

d-

E Th

di e-

t da

D te-

at an

e d-

ti

m

e-

th

e-

do

cu

m
en
t
w
as
la
st
sa
ve
d.

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.

PE S

r q a

eui m

- v pl

s a l e

ee

t nt

FC

ou

r st

me

um

l F

a or

m

ul

a

% 8 / 1 /

DF 8 /

a L 97

M

l

D /

Y

Y

% M Ja

De nu

b nt ar

~~h y~~

~~D 8~~

~~Y 19~~

~~Y 97~~

~~Y~~

~~Y~~

~~%M JA~~

~~De N~~

~~Bnt U~~

~~h A~~

~~D R~~

~~Y Y~~

~~Y 8~~

~~Y 19~~

~~Y 97~~

~~%D 8~~

~~DM Ja~~

~~ee nu~~

~~nt ar~~

~~h y~~

~~Y 19~~

~~Y 97~~

~~Y~~

~~Y~~

~~%D 8~~

~~DM JA~~

~~Ge N~~

~~nt U~~

~~h A~~

~~Y R~~

~~Y Y~~

~~Y 19~~

~~Y 97~~

~~%WS~~

~~De at~~

de ur
k da
d y:
a Ja
y: nu
Mar
e y
nt 8:
h 19
D,97
Y
Y
Y
Y
%WS
De A
De F
k U
d R
a D
y: A
M Y:
e JA
nt N
h U
D: A
Y R
Y Y
Y 8:
Y 19
97
%M Ja
De nu
ent ar
h y

D 8
~~%M~~ JA
De N
Ent U
h A
D R
Y
8
~~%WS~~
De at
f e ur
k da
d y
a Ja
y nu
Mar
e y
nt 8
h
D
~~%WS~~
De A
Fe I
k U
d R
a D
y A
M Y
e JA
nt N
h U
D A
R
Y
8

~~%M 1/~~
~~D/D8~~

g

~~%M 1/~~
~~D/ 8/~~
~~hD/19~~

~~Y 97~~

~~Y~~

~~Y~~

~~Y~~

~~%D.8.~~

~~DM Ja~~

~~i e nu~~

~~nt ar~~

~~h y~~

~~%D.8.~~

~~DM JA~~

~~l e N~~

~~nt U~~

~~h A~~

~~R~~

~~Y~~

~~%D.8.~~

~~DM Ja~~

~~j e nu~~

~~nt ar~~

~~h y~~

~~Y 19~~

~~Y 97~~

~~Y~~

~~Y~~

~~%D.8.~~

~~DM JA~~

~~Je N~~

~~nt U~~

h A
Y R
Y Y
Y 19
Y 97
~~%Y~~ 19
DY 97
k Y Ja
Y nu
Ma r
e y
nt 8
h
D
~~%Y~~ 19
DY 97
KY JA
Y N
M U
e A
nt R
h Y
D 8
~~%M~~ Ja
De nu
! nt ar
h. y.
Y 19
Y 97
Y
Y
~~%M~~ JA
De N
L nt U
h. A

Y R
Y Y-
Y 19
Y 97
%D 08
DD//0
mM 1/
M 19
/ 97
Y
Y
Y
Y
%H 09
TH::0
1m 1
m (le
ad
in
g-
ze
ro-
on
ho
ur
}
%H:9:
Im 01
2m A
a M
m
p
m
%H 09
TH::0

3m 1

m A

a M

m (le

p ad

m in

g

ze

ro

on

ho

ur

}

%n 9:

Te 01

4n A

e}

%n 09

Te :0

5n 1

e)A

(le

ad

in

g

ze

ro

on

ho

ur

}

%H:9:

Im 01

6m a

a m

m
p
m
%H 09
TH::0
7m 1a
m m-
a (le
mad
p in
m g-
ze
re-
on
he
ur
}
%n 9:
Te 01
8n a
e)
%n 09
Te :0
9n 1a
e) (le
ad
in
g-
ze
re-
on
he
ur
}

G
u D S
s e a
t sem
e ri pl
m pte
- ie
f n
e
r
m
u
l
a
e
l
e
m
e
n
t
%B S
Fe at
Lgi ur
n da
s y
a Ja
m nu
ix ar
e y
d 1
e 19
a 97
s
e

d
at
e
fo
r
m
ul
a
(a
ft
er
a
s
p
e
cif
ic
d
at
e
s
u
e
h
a
s
C
re
at
e
D
at
e)
:
%B S

Fe A

Ggi I

n U

s R

a D

n A

u Y

p JA

p N

er U

e A

a R

s Y

e 1

d 19

at 97

e

fo

f

m

ul

a

fa

ft

er

a

s

p

e

cif

ic

d

at

e

s

u
e
h
a
s
G
re
at
e
D
at
e)

±
SR (S
y et ee
s urth
t n e
es D
nth at
Se e
hd pa
oat ne
rte t
Din in
ath yo
t e ur
eS R
h eg
orio
t na
D t
at S
e ett
fo in
f gs

m e
a n
w e
h p
e n
h t
i
s
p
e
e
e
i
e
d
i
n
t
e
D
a
e
p
a
n
n
e
l
o
f
t
h
e
R
e
g
i
e
n
a
t
S
e
t
t

n

g

s

e

e

nt

ro

t

p

a

n

el

z

SR ~~(S~~

y et ee

s ur th

t n e

es D

nth at

Le e

od pa

nat ne

ge t

Din in

ath yo

t e ur

e L R

e eg

n io

g na

D t

at S

e ett

fo in

f gs

m ce
at ntr
w ol
hi pa
e ne
h h
is
s
p
e
eif
ie
d
in
th
e
D
at
e
p
a
n
el
of
th
e
R
e
gi
e
n
at
S
et
ti

n
g
s
e
e
nt
re
t
p
a
n
el

i
R 19
Set 97
Our 0
Dn 1/
as 08

t th
ee
1d
at
e
in
th
e
IS
O
D
at
e
1
fe
t
m

at
:
l R 19
Set 97
Our /0
Dn 1/
as 08
t th 09
ee :0
2d 1:
at 05
e
in
th
e
IS
Q
D
at
e
2
fo
t
m
at
:
MR :
oet Ja
nur nu
t n ar
hs y
th
e
e
nt

ir

e

m

e

nt

h

n

a

m

e.

MR :

netJa

urn

n

s

a

n

a

b

br

e

vi

at

e

d

m

e

nt

h

n

a

m

e.

WR :

eetS

eurat

kn ur

ds da

ath y

ye

e

nt

ir

e

w

e

e

k

d

a

y

n

a

m

e.

WR -

det S

aurat

yn

s

a

n

a

b

br

e

vi

at

e

d

w
e
e
k
d
a
y
n
a
m
e.

DR 8

e
t
u
r
n
s
t
e
d
a
y
d
i
g
t.

DR :

Det 08

u
r
n
s
t
e
d
a
y
d
i

gi

t

wi

th

le

a

di

n

g

z

er

e.

MR -1

et

ur

n

s

th

e

m

e

nt

h

di

gi

t.

MR :

Met 01

ur

n

s

th

e

m

e

nt

h

di

gi

t

wi

th

le

a

di

n

g

z

er

e

YR 97

Yet

ur

n

s

th

e

a

b

br

e

vi

at

e

d

y

e

ar

i

YR 19

Yet 97

Yur

Yn

s

th

e

e

nt

ir

e

y

e

ar

=

HR 9

et

ur

n

s

th

e

h

e

ur

=

HR :

Het 09

ur

n

s

th

e

h

e

ur

wi
th
le
a
di
n
g
z
er
o.

mR -1

et
ur
n
s
th
e
m
in
ut
e
s.

mR :

met 01

ur
n
s
th
e
m
in
ut
e
s
wi

th

le

a

di

n

g

z

er

o.

SR 5

et

ur

n

s

th

e

s

e

e

o

n

d

s.

SR :

Set 05

ur

n

s

th

e

s

e

e

o

n

d
s
wi
th
le
a
di
n
g
z
er
e
aU -
ms A
pe M
ms
th
e
1
2-
h
e
ur
el
e
ek
a
n
d
a
p
p
e
n
d

s
th
e
A
M
AP
M
in
di
e
at
of
z

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.

\$LwpDocVarDoctype (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.~~

~~\$LwpDocVarOtherversionededitors (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 Add DdeLink 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-8C12-02608G454BFF}

ProgID = FLW3Presentation

Launches Freelance 96 and prompts the user with the New Presentation dialog box.

Lotus Freelance 96 Drawing

ClassID = {CF746001-94FB-101B-8C12-02608G454BFF}

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 = ScreenCamMoviev2

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.

\$LwpStartTypeNextoddpag (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.

LwpPageNumberFlagsIncludeseaname (&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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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.

ValEff

ue ect

\$L Indi

wp cat

Ta es-

ble that

Ins the-

Ty met

pe hod

Ro sho

w uld-

(1 ins

87 ert

5) row

s-

\$L Indi

wp cat

Ta es-

ble that

Ins the-

Ty met

pe hod

Co sho

lu uld-

mnins

(1 ert

87 col

6) um

ns-

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.

\$LwpStartTypeNextoddpag (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: InsertTOG method

{button ,AL('H_WPAPPLICATION_CLASS':0)} See list of classes

{button ,AL('H_INSERTTOG_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].InsertTOG([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

WPApplication. 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].WPAApplication.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

WPAApplication 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

WPAApplication as the source for the paste.

When you call InternalPaste from WPAApplication, 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

fbutton ,AL('H_BAGCOLLECTION_CLASS;H_BASECOLLECTION_CLASS;H_BASET
ABLE_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_GELLCOLLECTION_CLASS;
H_GELLLAYOUTCOLLECTION_CLASS;H_CHARACTERSTYLECOLLECTION_GLAS
S;H_CLICKHERE_CLASS;H_CLICKHERECOLLECTION_CLASS;H_CONNECTEDLA
YOUTCOLLECTION_CLASS;H_CONTENT_CLASS;H_CONTENTCOLLECTION_GLA
SS;H_DDELINKCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCIN
FOFIELDCOLLECTION_CLASS;H_DOCUMENTS_CLASS;H_DOGWINDOWCOLLEC
TION_CLASS;H_DROPCAPLAYOUTCOLLECTION_CLASS;H_EDITORCOLLECTION
_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECT
ION_CLASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTI

ON_CLASS;H_FOOTNOTETABLE_CLASS;H_FORMULA_CLASS;H_FRAMELAYOUT
COLLECTION_CLASS;H_GLOSSARY_CLASS;H_GLOSSARYCOLLECTION_CLASS;
H_GRAPHIC_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHICOLEBJECT_
CLASS;H_GRAPHICOLEBJECTCOLLECTION_CLASS;H_GROUPLAYOUTCOLLEC
TION_CLASS;H_HEADERLAYOUTCOLLECTION_CLASS;H_ICONBARCOLLECTION
_CLASS;H_LAYOUTCOLLECTION_CLASS;H_MARKERCOLLECTION_CLASS;H_ME
NUEMITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLASS;H_OLEOBJE
CT_CLASS;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLINESEQCOLLECTION_C
LASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELAYOUTCOLLECTION_
CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARALLELCOLSCOLLECTI
ON_CLASS;H_PARALLELCOLUMNS_CLASS;H_POWERFIELDCOLLECTION_GLAS

S;H_ROW LAYOUT COLLECTION_CLASS;H_RUBY LAYOUT COLLECTION_CLASS;H_SECTION COLLECTION_CLASS;H_SILVER BULLET COLLECTION_CLASS;H_SMART CORRECT COLLECTION_CLASS;H_SMART FILL COLLECTION_CLASS;H_STATUS BAR BUTTON COLLECTION_CLASS;H_STRING COLLECTION_CLASS;H_SUPER TABLE_COLLECTION_CLASS;H_SUPER TABLE COLLECTION_CLASS;H_SUPER TABLE LAYOUT COLLECTION_CLASS;H_TABLE_COLLECTION_CLASS;H_TABLE HEADING_COLLECTION_CLASS;H_TABLE HEADING COLLECTION_CLASS;H_TABLE HEADING LAYOUT COLLECTION_CLASS;H_TABLE LAYOUT COLLECTION_CLASS;H_TABLE MARKER COLLECTION_CLASS;H_TABLE ONLY COLLECTION_CLASS;H_TEXT_COLLECTION_CLASS;H_TEXT COLLECTION_CLASS;H_TEXT MARKER COLLECTION_CLASS;H_TEXT STYLE COLLECTION_CLASS;H_VERSION COLLECTION_CLASS;H_WP DATASET COLLECTION_CLASS

ASS;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.

ValEff

~~ue ect~~
~~\$L Th~~
~~wp e-~~
~~Do na~~
~~cV me~~
~~ar s of~~
~~All all-~~
~~verver~~
~~sio sio~~
~~nn ns-~~
~~am of a~~
~~es doc~~
~~(2 um~~
~~10 ent.~~
~~l~~
~~\$L Th~~
~~wp e-~~
~~Do We~~
~~cV rd-~~
~~ar Pro~~
~~Gr use~~
~~eatr~~
~~ed na~~
~~by me-~~
~~(2 of~~
~~04 the-~~
~~6) per~~
~~son~~
~~wh~~
~~e-~~
~~first~~
~~sav~~
~~ed-~~
~~the-~~
~~doc~~

um
ent.
\$L Th
wpe
Do dat
eV e
ar and
Da tim
tece
re the
atedoc
d um
(1 ent
96 wa
) s
first
cre
ate
d.
\$L Th
wpe
Do last
eV tim
ar e
Da the
tel doc
astum
revent
isi wa
on s
(1 ope
97 ned
) =
edit
ed.

and
sav
ed.
\$L Dis
wp pla
Do ys
eV a
ar des
De crip
scrtion
ipti of
on the
{1 doc
95 um
} ent.
\$L Th
wp e
Do na
eV me
ar of a
Di divi
vis sio
ion n
na wit
mehin
{2 the
03 doc
} um
ent.
\$L Th
wp e
De ass
eV ign
ar ed
De cat

~~ee ego~~
~~atery~~
~~go of~~
~~ry the~~
~~(2 divi~~
~~12 sio~~
~~) n-~~
~~or~~
~~doc~~
~~um~~
~~ent.~~
~~\$L Th~~
~~wp e-~~
~~Do siz~~
~~eV e-of~~
~~ar the~~
~~Do doc~~
~~esi um~~
~~ze ent~~
~~(2 in-~~
~~02 kilo~~
~~) byt~~
~~es.~~
~~\$L Sp~~
~~wp ecif~~
~~Do ies-~~
~~eV the~~
~~ar Do~~
~~Fieclnf~~
~~ld e-~~
~~(1 fiel~~
~~91 d-~~
~~) you~~
~~wa~~
~~nt~~

~~to~~
~~exp~~
~~ort~~
~~as~~
~~Not~~
~~es~~
~~FX.~~
~~\$L Th~~
~~wp e~~
~~Do file~~
~~cV na~~
~~ar me~~
~~File you~~
~~en spe~~
~~amcifi~~
~~e ed~~
~~(1 in~~
~~92 the~~
~~) Sa~~
~~ve~~
~~As~~
~~dial~~
~~og~~
~~box~~
~~wh~~
~~en~~
~~you~~
~~sav~~
~~ed~~
~~the~~
~~doc~~
~~um~~
~~ent.~~
~~\$L Dis~~
~~wp pla~~

~~De ys-~~
~~eV the-~~
~~ar ass~~
~~Ke ign~~
~~yw ed-~~
~~or key~~
~~ds wor~~
~~(2 ds-~~
~~15 for-~~
~~) the-~~
~~doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wpe-~~
~~De initi~~
~~eV als-~~
~~ar of~~
~~La the-~~
~~ste edit~~
~~dit or-~~
~~or wh~~
~~(2 e-~~
~~04 last~~
~~7) sav~~
~~ed-~~
~~the-~~
~~doc~~
~~um~~
~~ent.~~

~~\$L No-~~
~~wpe De~~
~~De clnf~~
~~eV e-~~
~~ar fiel~~

~~No ds-~~
~~ne are~~
~~(1 set~~
~~90 or~~
~~) res~~
~~et~~
~~as~~
~~Not~~
~~es~~
~~FX.~~

~~\$L Th~~
~~wpe-~~
~~Do nu~~
~~eV mb~~
~~ar er-~~
~~Nu of~~
~~me cha~~
~~ha ract~~
~~rs- ers-~~
~~(2 in-~~
~~01 the-~~
~~) doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wpe-~~
~~Do nu~~
~~eV mb~~
~~ar er-~~
~~Nu of~~
~~mppag~~
~~ag es-~~
~~es in-~~
~~(1 the-~~
~~99 doc~~

um
ent.

\$L Th

wpe

Do nu

cV mb

ar er

Nu of

mv ver

erssio

ion ns

s in

{2 the

09 doc

um

ent.

\$L Th

wpe

Do nu

cV mb

ar er

Nu of

m wor

wo ds

rdsin

{2 the

00 doc

um

ent.

\$L Th

wpe

De We

cV rd

ar Pro

Of use

he r
re na
dit me
orss of
(2 oth
04 er
8) peo
ple
wh
e
sav
ed
the
doc
um
ent.

\$L Th
wp e
Do Wo
cV rd
ar Pro
Of use
he r
rvena
rsi me
on s of
ed oth
edier
for peo
s ple
(2 wh
07 e
) sav
ed
a

ver
sio
n-of
the
doc
um
ent.

\$L Th
wpe
Do driv
eV e
ar and
Pa fold
th er
{1 wh
93 ere
} the
doc
um
ent
is
loc
ate
d.

\$L Th
wpe
Do na
eV me
ar of a
Se sec
cti tion
on in
na the
medoc
{2 um

04 ent.

}

\$L Th

wpe

Do styl

eV e

ar she

St et

yle use

sh d-in

eetthe-

{1 doc

94 um

} ent.

\$L Th

wpe

Do tota

eV t

ar nu

Totmb

ale er

ditt of

im min

e ute

{1 s

98 the-

} doc

um

ent

wa

s

ope

n.

\$L Th

wpe

De dat
eV e-
ar and
Ve tim
rsi e-
on the-
crever
atesio
datn-
e wa
{2 s-
06 first
} cre
ate
d.
\$L Th
wp e-
De Wo
eV rd-
ar Pro
Ve use
rsi r-
on na
creme-
ateof
db the-
y per
{2 son
05 wh
} e-
first
sav
ed-
the-
ver

~~sio~~
~~n of~~
~~the~~
~~doc~~
~~um~~
~~ent.~~

~~\$L Th~~
~~wp e~~
~~Do dat~~
~~cV e~~
~~ar and~~
~~Ve tim~~
~~rsi e~~
~~onl that~~
~~ast the~~
~~ediver~~
~~tdasio~~
~~te n~~
~~(2 wa~~
~~13 s~~
~~) last~~
~~sav~~
~~ed.~~

~~\$L Th~~
~~wp e~~
~~Do initi~~
~~cV als~~
~~ar of~~
~~Ve the~~
~~rsi edit~~
~~onl or~~
~~ast wh~~
~~edi e~~
~~ted last~~
~~by sav~~

~~(2 ed~~
~~14 the~~
~~) ver~~
~~sio~~
~~n.~~
~~\$L Th~~
~~wp e~~
~~Do na~~
~~cV me~~
~~ar of~~
~~Ve the~~
~~rsi cur~~
~~on ren~~
~~na t~~
~~mever~~
~~(2 sio~~
~~08 n.~~
~~)~~
~~\$L Th~~
~~wp e~~
~~Do nu~~
~~cV mb~~
~~ar er~~
~~Ve of~~
~~rsi tim~~
~~on es~~
~~nu the~~
~~mf ver~~
~~evi sio~~
~~sio n~~
~~ns wa~~
~~(2 s~~
~~04 ope~~
~~9) ned~~
~~=~~

~~edit~~
~~ed.~~
~~and~~
~~sav~~
~~ed.~~
~~\$L An~~
~~wp y~~
~~Do re~~
~~cV ma~~
~~ar rks~~
~~Ve writ~~
~~rsi ten~~
~~on by~~
~~re an~~
~~maedit~~
~~rks or~~
~~(21 to~~
~~1) be~~
~~revi~~
~~ew~~
~~ed~~
~~by~~
~~oth~~
~~er~~
~~ass~~
~~ign~~
~~ed~~
~~edit~~
~~ors.~~

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_CLICKHE

RE_CLASS;H_DROPDOWNCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_N

OTECONTAINER_CLASS;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAI

NER_CLASS;H_ROWCONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPA

GECONTAINER_CLASS;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECON

TAINER_CLASS;H_TABLECONTAINER_CLASS;H_TABLEONLYCONT_CLASS;H_TE

XT_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_CLAS

S';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

fbutton ,AL('H_BAGCOLLECTION_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_
GELLCOLLECTION_CLASS;H_GELLLAYOUTCOLLECTION_CLASS;H_CHARACTER
STYLECOLLECTION_CLASS;H_GLICKHERECOLLECTION_CLASS;H_CONNECTED
LAYOUTCOLLECTION_CLASS;H_CONTENTCOLLECTION_CLASS;H_DDELINKCOL
LECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCINFOFIELDCOLLECTI
ON_CLASS;H_DOCUMENTS_CLASS;H_DOGWINDOWCOLLECTION_CLASS;H_DR
OPCAPLAYOUTCOLLECTION_CLASS;H_EDITORCOLLECTION_CLASS;H_ENDNO
TELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECTION_CLASS;H_FO
OTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTION_CLASS;H_FR
AMELAYOUTCOLLECTION_CLASS;H_GLOSSARYCOLLECTION_CLASS;H_GRAPH

IGCOLLECTION_CLASS;H_GRAPHICOLEBJECTCOLLECTION_CLASS;H_GROUP

LAYOUTCOLLECTION_CLASS;H_HEADERLAYOUTCOLLECTION_CLASS;H_ICONB

ARGCOLLECTION_CLASS;H_LAYOUTCOLLECTION_CLASS;H_MARKERCOLLECTIO

N_CLASS;H_MENUITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLA

SS;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLINESEQCOLLECTION_CLASS;H_

OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELAYOUTCOLLECTION_CLASS;H

_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARALLELCOLSCOLLECTION_CLA

SS;H_POWERFIELDCOLLECTION_CLASS;H_ROWLAYOUTCOLLECTION_CLASS;H

_RUBYLAYOUTCOLLECTION_CLASS;H_SECTIONCOLLECTION_CLASS;H_SILVER

BULLETCOLLECTION_CLASS;H_SMARTCORRECTCOLLECTION_CLASS;H_SMAR

TFILLCOLLECTION_CLASS;H_STATUSBARBUTTONCOLLECTION_CLASS;H_STRI

NGCOLLECTION_CLASS;H_SUPERTABLECOLLECTION_CLASS;H_SUPERTABLEL

AYOUTCOLLECTION_CLASS;H_TABLECOLLECTION_CLASS;H_TABLEHEADINGG

OLLECTION_CLASS;H_TABLEHEADINGLAYOUTCOLLECTION_CLASS;H_TABLELA

YOUTCOLLECTION_CLASS;H_TABLEMARKERCOLLECTION_CLASS;H_TABLEONL

YCOLLECTION_CLASS;H_TEXTCOLLECTION_CLASS;H_TEXTMARKERCOLLECTI

ON_CLASS;H_TEXTSTYLECOLLECTION_CLASS;H_VERSIONCOLLECTION_GLAS

S;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

[WPApplication]

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

WPApplication.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

fbutton ,AL('H_BASETABLE_CLASS;H_GELLGROUPLAYOUT_CLASS;H_GELLAYO
UT_CLASS;H_CLICKHERE_CLASS;H_COLUMNNGROUPLAYOUT_CLASS;H_CONNE
CTEDLAYOUT_CLASS;H_DROPGAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLAS
S;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FOOTNOTETABL
E_CLASS;H_FORMULA_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAY
OUT_CLASS;H_GLOSSARY_CLASS;H_GRAPHIC_CLASS;H_GRAPHICOLEOBJECT
_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_GLA
SS;H_NOTELAYOUT_CLASS;H_OLEOBJECT_CLASS;H_PAGELAYOUT_CLASS;H_
PARALLELCOLUMNS_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_C
LASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPE

RTABLELAYOUT_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLASS;H_TOGSUPERTABLELAYOUT_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.

ValEff

ue ect

\$L Ins

wp erts

Maa

rkeboo

rTykm

pe ark

Bo ma

ok rker

ma :

rk

{5

90

}

\$L Ins

wp erts

Maa

rkeGlic

rTykH

pe ere

Gli ma

ek rker

he :

re

{5

93

}

\$L Ins

wp erts

Maa

rkeDD

~~rTyE~~
~~pe ma~~
~~Dd rker~~
~~e~~
~~(5~~
~~91~~
~~}~~
~~\$L Ins~~
~~wp erts~~
~~Maa~~
~~rkepos~~
~~rTyitio~~
~~pe nal~~
~~De ma~~
~~faurke~~
~~It r~~
~~(5 not~~
~~89 of~~
~~}~~ ~~any~~
~~oth~~
~~er~~
~~typ~~
~~e~~
~~and~~
~~is~~
~~onl~~
~~y~~
~~use~~
~~d to~~
~~ma~~
~~rk a~~
~~spo~~
~~t or~~
~~pos~~
~~itio~~

~~fr.~~
~~\$L Ins~~
~~wp erts~~
~~Maa~~
~~rkefiel~~
~~rTyd~~
~~pe ma~~
~~Fierker~~
~~ld :~~
~~(5~~
~~92~~
~~)}
~~\$L Ins~~
~~wp erts~~
~~Maa~~
~~rkeRu~~
~~rTyby~~
~~pe ma~~
~~Ru rker~~
~~by :~~
~~(5 Us~~
~~94 ed~~
~~)} onl
~~y in~~
~~the~~
~~Asi~~
~~an~~
~~lan~~
~~gua~~
~~ge~~
~~ver~~
~~sio~~
~~ns~~
~~of~~
~~We~~~~~~

~~rd~~

~~Pro~~

~~z~~

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.

ValEff
ue ect
\$L Mo
wp difi
Ra es
ng the
eP star
art ting
Be poi
th nt
(1 and
66 end
3) ing
poi
nt
of

~~the~~
~~na~~
~~me~~
~~d~~
~~ma~~
~~rke~~
~~r to~~
~~mat~~
~~ch~~
~~the~~
~~pos~~
~~itio~~
~~n of~~
~~the~~
~~ins~~
~~erti~~
~~on~~
~~poi~~
~~nt.~~
~~If~~
~~so~~
~~met~~
~~hin~~
~~g is~~
~~sel~~
~~ect~~
~~ed.~~
~~the~~
~~star~~
~~ting~~
~~and~~
~~end~~
~~ing~~
~~poi~~
~~nts~~

~~mat~~
~~ch~~
~~the~~
~~star~~
~~ting~~
~~and~~
~~end~~
~~ing~~
~~poi~~
~~nts~~
~~of~~
~~the~~
~~sel~~
~~ecti~~
~~on.~~
~~\$L Mo~~
~~wp difi~~
~~Ra es~~
~~ng the~~
~~eP end~~
~~art ing~~
~~En poi~~
~~d nt~~
~~(1 of~~
~~66 the~~
~~5) na~~
~~me~~
~~d~~
~~ma~~
~~rke~~
~~r to~~
~~mat~~
~~ch~~
~~the~~
~~pos~~

~~ition of the insertion point. If something is selected, the ending point matches the ending point of the selection.~~

wp difi
Ra es-
ng the-
eP star
art ting
St poi
art nt
(1 of
66 the-
4) ma
rke
r to
mat
ch-
the-
pos
itio
n of
the-
ins
erti
on-
poi
nt-
If
so
met
hin
g is
sel
ect
ed-
the-
star
ting

poi
nt
mat
che
s
the
star
ting
poi
nt
of
the
sel
ecti
on.

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 followings 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 MSStep~~

~~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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_GLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 work.

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", "I"

.Text.SplitParagraph

.InsertBullet "Wingdings", "O"

.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 successfully 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: IsAsciiGRLF 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

LwplIncludeListAllText (&H1) Sets the option to search all text in the document.

LwplIncludeListFootnotes (&H20) Sets the option to search all text in footnotes.

LwplIncludeListFrames (&H10) Sets the option to search all text in frames.

LwplIncludeListHeadersFooters (&H4) Sets the option to search all text in headers and

footers.

LwplIncludeListMainDocText (&H2) Sets the option to search text only in the main part of

the document.

LwplIncludeListTables (&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

indexindenttypevalue = [objectreference].IndexIndentType

[objectreference].IndexIndentType = indexindenttypevalue

Legal values

The legal values for this property are -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_LBUTTONDOWN 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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: IsAsciiGRLF property

{button ,AL('H_FILTER_CLASS;H_TEXTDOCUMENT_CLASS':0)} See list of classes

{button ,AL('H_ISASCIIGRLF_PROPERTY_EXSCRIPT',1)} See example

(Read-write)

Data Type

Integer

Syntax

isasciicrlfvalue = [objectreference].IsAsciiGRLF

[objectreference].IsAsciiGRLF = 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_ISASCIIKEEPSTYLE_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_C

LASS;H_ParallelColumns_Class;H_Table_Class;H_TableHeading_Clas

S'.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_GCELLLAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_ISCHANGEDSINGETIMESAVE_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_PO

WERFIELD_CLASS;H_RUBYMARKER_CLASS;H_TABLEMARKER_CLASS;H_TEXT

DOCUMENT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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

fbutton ,AL('H_BAGCOLLECTION_CLASS;H_BASECOLLECTION_CLASS;H_BASET
ABLE_CLASS;H_BOOKMARKCOLLECTION_CLASS;H_GELLCOLLECTION_CLASS;
H_GELLLAYOUTCOLLECTION_CLASS;H_CHARACTERSTYLECOLLECTION_GLAS
S;H_CLICKHERE_CLASS;H_CLICKHERECOLLECTION_CLASS;H_CONNECTEDLA
YOUTCOLLECTION_CLASS;H_CONTENT_CLASS;H_CONTENTCOLLECTION_GLA
SS;H_DDELINKCOLLECTION_CLASS;H_DIVISIONCOLLECTION_CLASS;H_DOCIN
FOFIELDCOLLECTION_CLASS;H_DOCUMENTS_CLASS;H_DOGWINDOWCOLLEC
TION_CLASS;H_DROPPLAYAYOUTCOLLECTION_CLASS;H_EDITORCOLLECTION
_CLASS;H_ENDNOTELAYOUTCOLLECTION_CLASS;H_FOOTERLAYOUTCOLLECT
ION_CLASS;H_FOOTNOTECOLLECTION_CLASS;H_FOOTNOTELAYOUTCOLLECTI

ON_CLASS;H_FOOTNOTETABLE_CLASS;H_FORMULA_CLASS;H_FRAMELAYOUT
COLLECTION_CLASS;H_GLOSSARY_CLASS;H_GLOSSARYCOLLECTION_CLASS;
H_GRAPHIC_CLASS;H_GRAPHICCOLLECTION_CLASS;H_GRAPHICOLEBJECT_
CLASS;H_GRAPHICOLEBJECTCOLLECTION_CLASS;H_GROUPLAYOUTCOLLEC
TION_CLASS;H_HEADERLAYOUTCOLLECTION_CLASS;H_ICONBARCOLLECTION
_CLASS;H_LAYOUTCOLLECTION_CLASS;H_MARKERCOLLECTION_CLASS;H_ME
NUIITEMCOLLECTION_CLASS;H_NOTELAYOUTCOLLECTION_CLASS;H_OLEOBJE
CT_CLASS;H_OLEOBJECTCOLLECTION_CLASS;H_OUTLINESEQCOLLECTION_C
LASS;H_OUTLINESEQITEMCOLLECTION_CLASS;H_PAGELAYOUTCOLLECTION_
CLASS;H_PARAGRAPHSTYLECOLLECTION_CLASS;H_PARALLELCOLSCOLLECTI
ON_CLASS;H_PARALLELCOLUMNS_CLASS;H_POWERFIELDCOLLECTION_GLAS

S;H_ROW LAYOUT COLLECTION_CLASS;H_RUBY LAYOUT COLLECTION_CLASS;H_SECTION COLLECTION_CLASS;H_SILVER BULLET COLLECTION_CLASS;H_SMART CORRECT COLLECTION_CLASS;H_SMART FILL COLLECTION_CLASS;H_STATUS BAR BUTTON COLLECTION_CLASS;H_STRING COLLECTION_CLASS;H_SUPER TABLE_COLLECTION_CLASS;H_SUPER TABLE COLLECTION_CLASS;H_SUPER TABLE LAYOUT COLLECTION_CLASS;H_TABLE_COLLECTION_CLASS;H_TABLE HEADING_COLLECTION_CLASS;H_TABLE HEADING COLLECTION_CLASS;H_TABLE HEADING LAYOUT COLLECTION_CLASS;H_TABLE LAYOUT COLLECTION_CLASS;H_TABLE MARKER COLLECTION_CLASS;H_TABLE ONLY COLLECTION_CLASS;H_TEXT_COLLECTION_CLASS;H_TEXT COLLECTION_CLASS;H_TEXT MARKER COLLECTION_CLASS;H_TEXT STYLE COLLECTION_CLASS;H_VERSION COLLECTION_CLASS;H_WP DATASET COLLECTION_CLASS

ASS;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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISIONINFO_GLAS
S;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT
_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRA
MELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LA
YOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUP
LAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTAB
LEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLA
YOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_DROPCA

PCONTAINER_CLASS;H_FRAMECONTAINER_CLASS;H_NOTECONTAINER_GLAS

S;H_PAGECONTAINER_CLASS;H_PARALLELCOLSCONTAINER_CLASS;H_ROWG

ONTAINER_CLASS;H_RUBYCONTAINER_CLASS;H_SUBPAGECONTAINER_CLASS

;H_SUPERPAGECONTAINER_CLASS;H_SUPERTABLECONTAINER_CLASS;H_TAB

LECONTAINER_CLASS;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_DROPCONTAINER_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_CLASS;H_TABLEONLYCONTAINER_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

ishoverhelpvalue = [objectreference].IsHoverHelp

[objectreference].IsHoverHelp = ishoverhelpvalue

Legal values

The legal values for this property are -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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_CHARACT
ERSTYLE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_C
LASS;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_PAGELAYOUT_CLASS;H_PARAGRA
PHSTYLE_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RU
BYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLASS;H_SUPERTABLELAY
OUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLELAYOUT_CLASS;H_TO
CSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_ISMISSPELLED_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_CLICKHE
RE_CLASS;H_COLUMNGROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;
H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_
CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRA
MELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LA
YOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUP
LAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTAB
LEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLA
YOUT_CLASS;H_TABLELAYOUT_CLASS;H_TEXT_CLASS;H_TEXTMARKER_GLAS
S;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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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 WPApplication 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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_PARAGRAPH

STYLE_CLASS;H_SILVERBULLET_CLASS;H_TEXT_CLASS;H_TEXTMARKER_CLA

SS',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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISION_CLASS;H_
DROPCAPLAYOUT_CLASS;H_ENDNOTE_LAYOUT_CLASS;H_FOOTERLAYOUT_CL
ASS;H_FOOTNOTE_LAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAME
LAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYO
UT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAY
OUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBY_LAYOUT_CLASS;H_SUPERTABLEG
ROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYO
UT_CLASS;H_TABLELAYOUT_CLASS;H_TEXTDOCUMENT_CLASS;H_TOCSUPER
TABLELAYOUT_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 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_Cl

Ass;H_Formula_Class;H_Glossary_Class;H_Graphic_Class;H_Graphi

COLEOBJECT_Class;H_OLEOBJECT_Class;H_ParallelColumns_Class;H_

SUPERTable_Class;H_Table_Class;H_TableHeading_Class;H_Text_Cla

SS';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_Basetable_Class;H_Footnotetable_Class;H_Glossary_C

LASS;H_ParallelColumns_Class;H_Table_Class;H_TableHeading_Clas

S'.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_TEXT

MARKER_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_CLASS;H_TOCSUPERTABLELAYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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

fbutton ,AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DIVISIONINFO_GLAS
S;H_DROPCAPLAYOUT_CLASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT
_CLASS;H_FOOTNOTELAYOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRA
MELAYOUT_CLASS;H_GROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LA
YOUT_CLASS;H_NOTELAYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUP
LAYOUT_CLASS;H_ROWLAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTAB
LEGROUPLAYOUT_CLASS;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLA
YOUT_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_CLAS

S;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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_C

LASS;H_PARALLELCOLUMNS_CLASS;H_TABLE_CLASS;H_TABLEHEADING_CLAS

S'.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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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: IsTOG property

{button .AL('H_CELLGROUPLAYOUT_CLASS;H_CELLLAYOUT_CLASS;H_COLUMN
GROUPLAYOUT_CLASS;H_CONNECTEDLAYOUT_CLASS;H_DROPCAPLAYOUT_C
LASS;H_ENDNOTELAYOUT_CLASS;H_FOOTERLAYOUT_CLASS;H_FOOTNOTELA
YOUT_CLASS;H_FRAMEGROUPLAYOUT_CLASS;H_FRAMELAYOUT_CLASS;H_G
ROUPLAYOUT_CLASS;H_HEADERLAYOUT_CLASS;H_LAYOUT_CLASS;H_NOTEL
AYOUT_CLASS;H_PAGELAYOUT_CLASS;H_ROWGROUPLAYOUT_CLASS;H_ROW
LAYOUT_CLASS;H_RUBYLAYOUT_CLASS;H_SUPERTABLEGROUPLAYOUT_CLAS
S;H_SUPERTABLELAYOUT_CLASS;H_TABLEHEADINGLAYOUT_CLASS;H_TABLEL
AYOUT_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 TOG table.

Data Type

Integer

Syntax

istocvalue = [objectreference].IsTOG

Legal values

The legal values for this property are -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_CLAS

S';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: IsUpdateTOG 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].IsUpdateTOG

[objectreference].IsUpdateTOG = 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

invalidvalue = [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_ISVIEWMENUCLEANSKRN_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_ISVIEWRETURNICONGLEANSGRN_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_ISVIEWSMARTICONSGLAANSCRN_PROPERTY_EXSCRIPT',1)} See

example

(Read-write)

Data Type

Integer

Syntax

isviewsmarticonscleanscrnvalue = [objectreference].IsViewSmartIconsCleanScrn

[objectreference].IsViewSmartIconsCleanScrn = isviewsmarticonscleanscrnvalue

Legal values

The legal values for this property are -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: IsTOG 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: IsUpdateTOG 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.