

```

//*****
//*   MACRO: TCONVERT.WCM
//*   PURPOSE:   Converts templates that were created for use with AUTOFILL.WCM.
//*               Converted templates work with both the Template Fill feature and
*_AUTOFIL.WCM.
//*****
Application (A1; "WordPerfect"; Default; "UK")

NumDefFld:=9           // Holds number of default personal information fields
MaxPrompts:=12        // Maximum number of prompts that fit in a dialog

Declare(DefFld[NumDefFld])           // Holds default personal field names

PromptAbbrName:="Template Prompts"
BkmkAbbrName:="Template Bookmarks"
ObjectFilename:="_AUTOTMP.WPT"
ObjectName:="<dofiller>"
DelimiterChar:="|"
ConvertOldFields:=True
OldField:="Company"
NewField:="Organization"
// Set names of personal information fields
DefFld[1]:="<Name>"
DefFld[2]:="<Title>"
DefFld[3]:="<Company>"
DefFld[4]:="<Address>"
DefFld[5]:="<Town>"
DefFld[6]:="<County>"
DefFld[7]:="<Postcode>"
DefFld[8]:="<Telephone>"
DefFld[9]:="<Fax>"
HasBookmarks:=False

Call(CloseSubdocs@)

ObjectFile:=FindFile(ObjectFilename)
If(ObjectFile="")
    MessageBox(; ObjectFilename+" Not Found"; "The template named ^0 was not found in
either of your template directories. Make certain that ^0 is in your Default or Additional
Templates Directory, then play TCONVERT.WCM again."; IconStop! | HasParameters!;
ObjectFilename)
    Quit
EndIf

Filename:=?Name
If(ToUpper(SubStr(Filename; StrLen(Filename)-3; 4))<>".")+ToUpper(?

```

```

DefaultTemplateExtension))
  Filename:=""
  Label(Display@)
  DialogDefine("Dlg1"; 50; 50; 172; 90; 1+2+16; "Convert Template")
  DialogAddFileNameBox("Dlg1"; "Fb1"; 8; 8; 150; 14; 2; Filename; ?
  PathTemplate; "*.wpt")
  DialogAddText("Dlg1"; "T1"; 8; 28; 150; 20; 1; "Select a template file to convert.")
  DialogDisplay("Dlg1"; "Fb1")
  Button:=MacroDialogResult
  DialogDestroy("Dlg1")
  If(Button=2)
    Quit
  EndIf
  Result:=FindFile(Filename)
  If(Result="")
    MessageBox(; "Template Not Found"; "The template file ^0 was not found,
    please include the full path and filename of the template."; Ok! | HasParameters!;
    Filename)
    Go(Display@)
  Else
    Filename:=Result
  EndIf
  TemplateEdit(Filename)
  ExitWhenDone:=True
Else
  ExitWhenDone:=False
EndIf

If(?DocBlank)
  MessageBox(; "Template Is Blank"; "Template contains nothing to convert."; IconStop!)
  Quit
EndIf

If(GetAbbreviation(BkmkAbbrName)<>"")
  MessageBox(MsgBoxResult; "Template Already Converted"; "This template already
  appears to be converted. Do you wish to continue?"; IconQuestion! | YesNo!)
  If(MsgBoxResult<>6)
    Quit
  EndIf
  HasBookmarks:=True
EndIf

Display(On!)
WaitMessage:="Converting template fields"
Call(WaitMessage@)
Display(Off!)

```

OnCancel(End@)

TemplateCopyObject(ObjectFile; Macro!; ObjectName)  
TemplateSetAssociation(PostNew!; Macro!; ObjectName)

PosDocVeryTop()  
MatchSelection ()  
SearchWrap(No!)  
SearchFindWholeWordsOnly(No!)  
SearchCaseSensitive(No!)

If(ConvertOldFields)  
    OnNotFound(NextOld@)  
    SearchString("<"+OldField+">")  
    ReplaceString("<"+NewField+">")  
    ReplaceAll(Extended!)  
    Label(NextOld@)  
    OnNotFound(LastOld@)  
    PosDocVeryTop()  
    SearchString("[ "+OldField+" ]")  
    ReplaceString("[ "+NewField+" ]")  
    ReplaceAll(Extended!)  
    Label(LastOld@)  
    PosDocVeryTop()

EndIf

// Convert the Personal Information Fields  
BkmrkString:=""  
BkmrkCount:=1  
For(Count; 1; Count<=NumDefFld; Count+1)  
    OnNotFound(NextDefFld@)  
    SearchString(DefFld[Count])  
    Repeat  
        SearchNext(Extended!)  
        BookmarkCreate(BkmrkCount)  
        BkmrkString:=BkmrkString+Count+DelimiterChar  
        BkmrkCount:=BkmrkCount+1  
    Until(False)  
  
    Label(NextDefFld@)  
    Call(CloseSubdocs@)  
    PosDocVeryTop()

EndFor

If(StrLen(BkmrkString)=0)  
    BkmrkString:=BkmrkString+DelimiterChar

```

EndIf

// Convert the Template Prompts
SingleString:=GetAbbreviation(PromptAbbrName)
If(SingleString="")
    BkmrkString:=BkmrkString+DelimiterChar
EndIf
PullString:=SingleString
NewSingleString:=""
ArrayCount:=0
While(PullString<> "")
    ArrayCount:=ArrayCount+1
    For(Loop; 0; Loop<2; Loop+1)
        StrPos(NextDelimLoc; DelimiterChar; PullString)
        If(NextDelimLoc=0)
            AbbValue:=PullString
            PullString:=""
            Go(LastOne@)
        EndIf
        StrLen(PullStringLength; PullString)
        SubStr(AbbValue; 1; NextDelimLoc-1; PullString)
        SubStr(PullString; NextDelimLoc+1; PullStringLength; PullString)
        Label(LastOne@)
        If(ConvertOldFields)
            If(AbbValue=OldField)
                AbbValue:=NewField
            EndIf
            NewSingleString:=NewSingleString+AbbValue
            If(PullString<> "")
                NewSingleString:=NewSingleString+DelimiterChar
            EndIf
        EndIf
        If(Loop=0)
            OnNotFound(NextAddFld@)
            SearchString("[ "+AbbValue+" ]")
            Repeat
                SearchNext(Extended!)
                BookmarkCreate(BkmrkCount)
                BkmrkString:=BkmrkString+DelimiterChar+ArrayCount
                BkmrkCount:=BkmrkCount+1
            Until(False)

            Label(NextAddFld@)
            While(?Substructure)
                SubstructureExit()
            EndWhile
        EndIf
    EndFor
EndWhile

```

```

                PosDocVeryTop()
            EndIf
        EndFor
    EndWhile

    OnNotFound(ReadOnlyTemplate@)
    If(ConvertOldFields And (SingleString<>""))
        AbbreviationDelete(PromptAbbrName;CurrentDoc!)
        AbbreviationCreate(PromptAbbrName;CurrentDoc!;NewSingleString)
    EndIf
    If(HasBookmarks)
        AbbreviationDelete(BkmkAbbrName;CurrentDoc!)
    EndIf
    AbbreviationCreate(BkmkAbbrName;CurrentDoc!;BkmrkString)

    MsgBox(MsgBoxResult; "Conversion Complete"; "The template conversion is complete.
    Do you wish to overwrite the original template?"; YesNoCancel! | DefButton2!)
    Switch(MsgBoxResult)
        CaseOf 2:           // Cancel
            Go(End@)
        CaseOf 6:           // Yes
            FileSave(; WordPerfect_60!; Yes!)
        CaseOf 7:           // No
            FileSaveAsDlg()
    EndSwitch
    If(ExitWhenDone)
        Close()
    EndIf

    Label(End@)
    Call(KillWaitMessage@)
    Quit

    /*******
    /**    ROUTINE: ReadOnlyTemplate@
    /**    INPUT VARIABLES: None
    /**    OUTPUT VARIABLES: None
    /**    DESCRIPTION: Specifies when a 'not found' has occurred during the abbreviation
    tokens - probably due to a read-only template.
    /*******
    Label(ReadOnlyTemplate@)
    MsgBox(; "Read Only Template"; "An unexpected 'Not Found' condition occurred. The
    template file may be marked read only. You may be able to make a writable copy of the template
    and play TCONVERT.WCM again."; IconStop!)
    Go(End@)
    /*******

```

```
*****
/*  ROUTINE: WaitMessage@
/*  INPUT VARIABLES: WaitMessage
/*  OUTPUT VARIABLES: WaitMessageResult
/*  DESCRIPTION: Presents a Please Wait message to user. Call KillWaitMessage to turn
it off.
```

```
*****
Label(WaitMessage@)
WaitTitle:="WAIT PROMPT"
WaitMessageWidth:=89
DialogDefine(WaitTitle; 50; 50; WaitMessageWidth+22; 43; 16+256; WaitTitle)
DialogAddText(WaitTitle; "T1"; 8; 8; 46; 10; 1; "Please Wait...")
DialogAddText(WaitTitle; "T2"; 8; 19; WaitMessageWidth; 10; 1; WaitMessage)
DialogDisplay(WaitTitle; 1; WaitDlgCallBack@)
Return
```

```
Label(WaitDlgCallBack@)
Return
```

```
*****
```

```
*****
/*  ROUTINE: KillWaitMessage@
/*  INPUT VARIABLES: none
/*  OUTPUT VARIABLES: none
/*  DESCRIPTION: Turns off wait message.
```

```
*****
Label(KillWaitMessage@)
DialogDestroy(WaitTitle)
Return
```

```
*****
```

```
*****
/*  ROUTINE: CloseSubdocs@
/*  INPUT VARIABLES: none
/*  OUTPUT VARIABLES: none
/*  DESCRIPTION: Exits all Sub Documents (headers, text boxes, etc.).
```

```
*****
Label(CloseSubdocs@)
While(?Substructure)
    SubDoc=?CurrentSubDoc
    SubstructureExit()
    If((SubDoc=10) Or (SubDoc=11))
        BoxEnd()
    EndIf
EndWhile
```



```
        If(CurrAbbrName=AbbrName)
            GetData(SingleString;Abbreviation!;Data!;CurrentDoc!;CurrAbbrNum)
            Break
        EndIf
    EndFor
Return(SingleString)
EndFunc
//*****
```