

Application (A1; "WordPerfect"; UK)

Display(Off!)

Global(CommandFile; DoMerge)

```
*****  
*****
```

// Get name of INI file to read

CommandFile := GetValue ("Command Files"; "MYOB"; "swdrvs.ini")

MacroStatusPrompt(1;"Processing your template. Please Wait...")

DoMerge := 1

MergeFileName := GetValue("Settings";"MergeFile";CommandFile)

```
*****  
// SEARCH FOR FILL POINTS IN DOCUMENT AND FILL THEM
```

```
*****
```

OnError(Exit)

// If merge required, prepares for merge by converting to merge codes rather than actual data

// Leaves document open as current document

SearchAndFill(Domerge)

OnError(Exit)

OnNotFound(Exit)

If (DoMerge=1)

 // Complete the document merge

 MergeDoc(MergeFileName)

endif

OnError(Exit)

OnNotFound(Exit)

HiddenTextShowAll (State: Off!)

BookmarkFind("StartPoint")

Label(Exit)

WRITEDLL ("Settings"; "Run Flag" ; "0" ;CommandFile)

Quit

```
*****  
*****
```

Function GetValue (section;Tag;CommandFile)

If (?MajorVersion>6)

```

        DLLLoad(Kernel;"Kernel32.DLL")
else
        DLLLoad(Kernel;"Kernel.DLL")
endif

ReturnValue:=""
DLLCall(Kernel;"GetPrivateProfileString";itemnum:WORD;
{section;Tag;"";Address(ReturnValue);Loword(127);CommandFile})
DLLFree(Kernel)

Return(ReturnValue)

```

EndFunc

```

//*****
// *****
Procedure TConvert(MFile)

// opens a new document and converts the ascii delimited file to a WordPerfect merge file
// closes the document on completion and returns to the preceding document
OnError(ErrorMessage)
OnNotFound(ErrorMessage)
MacroStatusPrompt(1;"Converting your data file. Please Wait...")

FileNew()

ImportSetFileName (Filename: MFile)
ImportSetSource (SourceType: ASCII!)
ImportSetDestination (DestinationType: MergeData!)
ImportSetAsciiFieldDelimiter (FieldDelimiter: ",")
ImportSetAsciiRecordDelimiter (RecordDelimiter: "[SRt][HRt]")
ImportSetAsciiStrip (StripChars: "")
ImportSetAsciiEncap (EncapsulationChar: "''''")

ImportDoImport ()

PosDocTop ()
SearchString (StrgToLookFor: "[MRG: ENDRECORD]")
MatchPositionBefore ()
SearchNext (SearchMode: Extended!)
SearchString (StrgToLookFor: "[MRG: ENDFIELD]")
ReplaceString (RplcStrg: ";")
ReplaceBackward (SearchMode: Extended!)
SearchString (StrgToLookFor: "[MRG: ENDRECORD]")
MatchPositionBefore ()
SearchNext (SearchMode: Extended!)

```

```
SearchString (StrgToLookFor: "[HRt]")
ReplaceString (RplcStrg: "")
ReplaceBackward (SearchMode: Extended!)
SearchString (StrgToLookFor: "[MRG: ENDRECORD]")
MatchPositionBefore ()
SearchNext (SearchMode: Extended!)
DeleteCharPrevious ()
```

```
SelectDocTop ()
TFields := ?SelectedText
```

```
DeleteCharNext ()
DeleteCharNext ()
DeleteCharNext ()
```

```
MergeCode (Codes: FieldNames!; String: "temp")
PosDocTop ()
```

```
SearchString (StrgToLookFor: "temp")
MatchSelection ()
SearchNext (SearchMode: Extended!)
Type(TFields)
```

```
FileSave(MFile;4;1)
Close(No!)
```

```
MacroStatusPrompt(1;"Merging contact data. Please Wait...")
Go(TCExit)
```

```
Label(ErrorMessage)
MESSAGEBOX(x; "Error"; "An error occured in conversion of the comma delimited merge file."; IconStop!)
WRITEDLL ("Settings"; "Run Flag" ; "0" ;CommandFile)
Quit
```

```
Label(TCExit)
EndProc
```

```
Procedure SubExit()
While (?SubStructure)
    SubStructureExit
EndWhile
Type("x")
DeleteCharPrevious
EndProc
```

```
*****
```

```
//*****
```

Procedure SearchAndFill(DoMerge)

OnNotFound(NoMore)

OnError(NoMore)

PosDocVeryTop

Label (SearchLoop)

```
// Main search loop. Searches for tags with [..]
```

```
SearchString (StrgToLookFor: "[[")
```

```
MatchPositionBefore ()
```

```
SearchCaseSensitive (State: No!)
```

```
SearchFindWholeWordsOnly (State: No!)
```

```
MatchWithAttributes (State: No!)
```

```
MatchWithFontSize (State: No!)
```

```
MatchWithFont (State: No!)
```

```
SearchWrap (State: No!)
```

```
SearchNext (SearchMode: Extended!)
```

```
If (?SubStructure)
```

```
    CheckNextChar()
```

```
else
```

```
    SelectMode (State: On!)
```

```
    SearchString (StrgToLookFor: "]]")
```

```
    MatchExtendSelection ()
```

```
    SearchNext (SearchMode: Regular!)
```

```
endif
```

```
TemplateField := ?SelectedText
```

```
TemplateField=Substr(TemplateField;3;strlen(TemplateField)-4)
```

```
TagType := SubStr(TemplateField; 1; 3)
```

```
// If Driver returns 0, fill with value. If 1, convert to merge field
```

```
Switch (DoMerge)
```

```
    Caseof 0 :
```

```
        If (TagType = "My:")
```

```
            TemplateField := Substr(TemplateField; 4; strlen(TemplateField)-3)
```

```
            FieldValue := GetValue ("MyRecord"; TemplateField; CommandFile)
```

```
        else
```

```
            FieldValue := GetValue ("Fill"; TemplateField; CommandFile)
```

```
        endif
```

```
        Type(FieldValue)
```

```

    Caseof 1 :
        If (TagType = "My:")
            TemplateField := Substr(TemplateField; 4; strlen(TemplateField)-3)
            FieldValue := GetValue ("MyRecord"; TemplateField; CommandFile)
            Type(FieldValue)
        else
            Type("")
            //DeleteCharNext
            MergeCode (Codes: Field!; String: TemplateField)
        endif
    EndSwitch

```

Go(SearchLoop)

Label(NoMore)
SubExit()

EndProc

Procedure MergeDoc(MFile)

```

OnError(BadMerge)
ErrorMsg := "An error occured converting the merge file."
TConvert(MFile)

```

```

MergeSelect (SelectType: All!)
ErrorMsg := "An error occured executing the merge."

```

```

// Get number of current document so that it can be closed later
n1 := ?DocNumber

```

```

// Mark text and merge codes for deletion after merge
// Need to do this to maintain the Template Abbreviation that marks document as ACT! Document
SelectAll()
BookmarkCreate("PreMerge Text")

```

```

// Run the merge to current document and then delete codes later
MergeRun (FormFileType: Current!; DataFileType: DataFile!; DataFile: MFile;
OutputFileType: ToCurrentDoc!; )
PosPage (Volume: NoneSpecified!; Chapter: NoneSpecified!; Page: 1)
SearchString (StrgToLookFor: "ú")
ReplaceString (RplcStrg: "£")
ReplaceForward (SearchMode: Extended!)

```

```
// Select and delete the merge codes
```

```
BookMarkBlock("PreMerge Text")
```

```
DeleteCharNext
```

```
BookMarkDelete("PreMerge Text")
```

```
PageNumber (Page: 1)
```

```
Go(MExit)
```

```
Label(BadMerge)
```

```
MESSAGEBOX(x; "Error"; "An error ocured in conversion of the comma delimited merge file."; IconStop!)
```

```
WRITEDLL ("Settings"; "Run Flag" ; "0" ;CommandFile)
```

```
Quit
```

```
Label(MExit)
```

```
EndProc
```

```
//************************************************************************
```

```
Procedure WRITEDLL (Section;Tag;StrValue;IniFileName)
```

```
If (?MajorVersion>6)
```

```
    DLLLoad(Kernel;"Kernel32.DLL")
```

```
else
```

```
    DLLLoad(Kernel;"Kernel.DLL")
```

```
endif
```

```
DLLCall(Kernel;"WritePrivateProfileString";itemnum:WORD;{section;Tag;StrValue;IniFileName})
```

```
DLLFree(Kernel)
```

```
EndProc
```

```
//************************************************************************
```

```
Procedure CheckNextChar()
```

```
// This procedure is included due to a bug in WP8 with Search / Extend Selection
```

```
// operation inside of a text box
```

```
SelectMode (State: On!)
```

```
PosCharNext
```

```
PosCharNext
```

```
FoundTag := 0
```

```
Repeat
  If (?RightChar<>""]")
    PosCharNext
  else
    FoundTag := 1
  endif
Until (FoundTag=1)
```

```
PosCharNext
PosCharNext
```

```
EndProc
```