

BB11



Four Useful VBA Utilities for SAS® Programmers

David Franklin
TheProgrammersCabin.com

Introduction

- SAS Programmers occasionally have to look beyond SAS to do tasks that can be more efficiently outside SAS
- Tasks discussed here are:
 - converting files from one format to another (this paper will look at ASCII Text to Word)
 - print the first two pages of all Word documents in a directory
 - take all Word documents in a directory and concatenate them into one file
 - review all SAS LOGs in a directory and put any issues found into a single Word file for review
- Code is in the paper for these four VBA macros

What is VBA?

- VBA is a programming language that is inside products like MS Word that allow for the building of user defined functions and automating processes
- Is closely related to Visual Basic and its predecessors, Quick Basic and WordBasic
- VBA programs can be fixed to a menu button, a keyboard shortcut, or a window icon outside the MS Word environment.
- Has its own IDE

Converting Files From one Format to Another, Part 1

- There are five distinct steps to this
 - Open directory to process and select all files that are in that directory that fit the selection, in this case *.TXT
 - Open each file
 - Format text
 - Save as a Word file

Converting Files From one Format to Another, Part 2

- Open a Directory to Process and select all *.TXT files

```
With Application.FileDialog(msoFileDialogFolderPicker)
.InitialFileName = "C:\\"
.Show
strPath = .SelectedItems(1)
End With
MsgBox "You selected " & strPath
strFileName = Dir(strPath + "\*.txt", vbNormal)
```

Converting Files From one Format to Another, Part 3

- Select all files that are in that directory that fit the selection, in this case *.TXT, set font, then save

```
Do While strFileName <> ""
    strOutFileName = Replace(strFileName, ".txt", ".doc")
    Documents.Open FileName:=strPath + "\" + strFileName, _
        ConfirmConversions:= False, Encoding:=1252
    Selection.WholeStory
    Selection.Font.Name = "Courier New"
    Selection.Font.Size = 8
    ActiveDocument.SaveAs
        FileName:=strPath + "\" + strOutFileName, _
        FileFormat:= wdFormatDocument
    ActiveWindow.Close
    strFileName = Dir
Loop
```

Printing The First Two Pages Of All DOC Files In A Directory

```
Sub PrintPgsWord()  
    Dim strFileName As String  
    Dim strPath As String  
    strPath = "D:\TEMP\DOCFiles" ' change to your work  
area  
    strFileName = Dir(strPath + "\*.doc", vbNormal)  
    ' can be .rtf or .docx  
    Do While strFileName <> ""  
        Documents.Open FileName:=strPath + strFileName  
        Documents(strFileName).PrintOut  
        Range:=wdPrintFromTo, From:="1", To:="2"  
        Documents.Close  
        strFileName = Dir  
    Loop  
End Sub
```

Concatenate All Word Docs In a Directory Into A Single Word File

```
Sub MergeWrdDocs()  
    Dim RngA As Range  
    Dim CombDoc As Document  
    Dim FileInDoc As String  
    Const FolderLoc = "D:\TEMP\RTF\" 'change to your site  
    Set CombDoc = Documents.Add  
    FileInDoc = Dir$(FolderLoc & "*.rtf") ' can be .doc/.  
docx  
    Do Until FileInDoc = ""  
        Set RngA = CombDoc.Range  
        RngA.Collapse wdCollapseEnd  
        RngA.InsertFile FolderLoc & FileInDoc  
        RngA.InsertBreak Type:=wdSectionBreakNextPage  
        FileInDoc = Dir$(  
    Loop  
    ActiveDocument.SaveAs (FolderLoc + "COMBINED.doc")  
End Sub
```


Review All SAS Logs In A Directory, Part 1

```
sFileName = Dir$(FolderLoc & "*.LOG") ' Load all LOG files
Do Until sFileName = ""

    Selection.InsertAfter ("=====") &
vbCrLf
    Selection.InsertAfter ("Log Name: ") & sFileName &
vbCrLf
    Selection.InsertAfter (" ") & vbCrLf
    Open sFileName For Input As #1
    k = 0
```

Review All SAS Logs In A Directory, Part 2

```
Do While Not EOF(1)
  Line Input #1, Fields
  lPA = InStr(1, Fields, "ERROR:") ' ERROR text in line
  lPB = InStr(1, Fields, "WARNING:") ' WARNING text
  If lPA > 0 Or lPB > 0 Then ' One of the messages found
    Selection.InsertAfter Fields & vbCrLf
    k = k + 1
  End If
Loop
If k = 0 Then ' No issues found, report
  Selection.InsertAfter ("***No Issues Found***") &
vbCrLf
End If
Selection.InsertAfter (" ") & vbCrLf
Close #1
sFileName = Dir$() ' Get next file
Loop
```

Conclusion

- This paper has looked into four VBA utilities that are useful for the SAS programmer
- It is possible to do all these tasks within SAS, but it is not always the best way to do these

Contact Information



David Franklin

TheProgrammersCabin.com

603-275-6809

dfranklin@TheProgrammersCabin.com

<http://www.TheProgrammersCabin.com>

@ThePgmrsCabin