

Batch Extracted, Categorized and Saved Files Using Standard File Names

Overview of what the code does below:

1. Extract and download FNOL's URLs from columns
2. Categorizes and save files into 4 categories – CLAIMS_SUMMARY_MEMO, EXCLUSION, FNOL, OTHERS base on workflow types and category
3. The saved files should follow the standards of the target file name –
loss_no_original_source_file_name
 - a. Example:
 - a. Loss_no = CLP0102011
 - b. Original source file name = 2051010_OriginalCorrespondence.msg
 - b. Target file name = CLP0102011_2051010_OriginalCorrespondence.msg

```
1. Private Declare Function URLDownloadToFile Lib "urlmon" Alias "URLDownloadToFileA" (ByVal pCaller As Long,
ByVal szURL As String, ByVal szFileName As String, ByVal dwReserved As Long, ByVal lpfnCB As Long) As Long
2.
3. Dim WORK_FLOW_TYPE, Category, FolderName
4.
5. Sub GetFolderName()
6.
7. Dim x As Integer
8. Dim WF, CAT As String
9. Dim LastRow As Long
10. FolderName = "Others" 'set as Default Folder
11.
12. LastRow =
ThisWorkbook.Worksheets("KEY").Range("A1").Offset(ThisWorkbook.Worksheets("KEY").Rows.Count - 1,
0).End(xlUp).Row 'take column "A" to read the first to last row from "KEY" tab
13.
14. For x = 1 To LastRow 'loop from 1st row to the last row
15. WF = UCase(ThisWorkbook.Worksheets("KEY").Cells(x, "B").Value) 'set column "B" as WF for work flow
type and turn the whole column into all capital letters in "KEY" tab
16. CAT = UCase(ThisWorkbook.Worksheets("KEY").Cells(x, "A").Value) 'set column "A" as CAT for category and
turn the whole column into all capital letters in "KEY" tab
17. If WORK_FLOW_TYPE = WF And Category = CAT Then 'if WORK_FLOW_TYPE = WF AND Category = CAT
from "KEY" tab match with "attachmentURL_vba" tab
18. FolderName = ThisWorkbook.Worksheets("KEY").Cells(x, "C").Value 'Then, files will directly reference
column "C" from "KEY" tab to categorize into 4 folders
19. x = LastRow + 1 'exit the for loop
20. End If
21.
22. Next x
23. End Sub
24.
25. Sub DownloadFilefromURL()
26.
27. On Error GoTo ErrLab 'if the codes below in DownloadfilefromURL occur an error, it will run the
"ErrLab" statement - just like if-else statement
28.
29. Dim i As Long, j As Integer
```

```

30. Dim LastRow2 As Long
31. Dim c As Long
32. Dim n, x As Integer
33. Dim str As String
34. Dim errorfile As String
35. Dim fso As Object
36. Set fso = CreateObject("Scripting.FileSystemObject") 'set fso as a variable to access FileSystemObject
object is used to access the file system on a server
37.
38. errorfile = "C:\          \error.txt" 'set errorfile as a
variable to read error.txt that contains error line number (look at ErrLab statement for the creation of
error.txt )
39. n = FreeFile() 'set n to load FreeFile function
40. If fso.FileExists(errorfile) Then 'check if errorfile exists
41.     Open errorfile For Input As #n 'if yes, open errorfile as #n
42.     Line Input #n, str 'reads #n and assigns it as a str variable
43.     Close #n
44.     x = val(str) 'assign x as a numeric value
45. Else
46.     x = 1 'if errorfile doesn't exist, assign x = 1
47. End If
48.
49. LastRow2 =
ThisWorkbook.Worksheets("attachmentURL_vba").Range("A1").Offset(ThisWorkbook.Worksheets("attachment
URL_vba").Rows.Count - 1, 0).End(xlUp).Row 'set column "A" to read from first to last row from
attachmentURL_vba tab
50.
51. For i = x To LastRow2 'loop from x to last row
52.     DoEvents
53.     Application.StatusBar = CStr(i) 'have a status bar in the lower left hand corner
54.     WORK_FLOW_TYPE = UCase(ThisWorkbook.Worksheets("attachmentURL_vba").Cells(i, "B").Value)
'set column "B" as WORK_FLOW_TYPE and turn all rows in the column into capital letters in attachmentURL_vba
tab
55.     Category = UCase(ThisWorkbook.Worksheets("attachmentURL_vba").Cells(i, "C").Value) 'set
column "C" as Category and turn all rows in the column into capital letters in attachmentURL_vba tab
56.
57.     GetFolderName 'call GetFolderName Function from above
58.
59.     Dim URL_LIST As String
60.     Dim URL_ARRAY() As String
61.     URL_LIST = ThisWorkbook.Worksheets("attachmentURL_vba").Cells(i, "D").Value 'set column D as a list
in the cell from attachmentURL_vba tab
62.     URL_ARRAY = Split(URL_LIST, ";") 'split strings when URL/cell sees ";"
63.     Dim URL As String
64.     For j = LBound(URL_ARRAY) To UBound(URL_ARRAY) 'read from 1st string to last string in the list
from each row. e.g. abcde.jpg, Lbound = abcde, ubound = .jpg
65.         URL = Trim(URL_ARRAY(j)) 'trim empty spaces within the list
66.         Dim fileName As String
67.         fileName = fso.GetFileName(URL) 'get file name from URL
68.         Dim strSavePath As String

```

```

69.     Dim ext As String
70.     Dim buf, ret As Long
71.     buf = Split(URL, ".")           'split URL when URL sees "."
72.     ext = buf(UBound(buf))         'ext is set to read the extension only, exclude the period. e.g. (pdf,
                                     jpg, msg, etc) from URL
73.     'strSavePath = "C:\           ." & "\" & FolderName & "\" &
Range("A" & i).Value & "_" & fileName & "." & ext
74.     strSavePath = "\\
                                     " & "\" & FolderName & "\" & Range("A" & i).Value & "_" & fileName & "."
& ext           'save as: 1 of the 4 folders in claims_filing_cabinets -> take attachmentURL_vba's column A + _ +
filename + ext. E.g. CLP0102011_OriginalCorrespondence.msg
75.
76.     ret = URLDownloadToFile(0, URL, strSavePath, 0, 0) 'download URLs to files
77.     Next j
78.     Next i
79.     Dim DeleteFile As String
80.     DeleteFile = "C:\           \error.txt" 'set DeleteFile as variable
                                     of the path of the file that you want to delete
81.
82.     If fso.FileExists(DeleteFile) Then 'check if DeleteFile exists
83.         fso.DeleteFile (DeleteFile) 'if DeleteFile exists, It will delete the file from source location
84.     End If
85. Exit Sub
86.
87. ErrLab:
88.     Dim s As String
89.     n = FreeFile() 'use this function to get the file number as an integer and then use the Open statement to
save as
90.     Open "C:\           \error.txt" For Output As #n 'set #n as a
variable to save to your designated location; from your location, it will save as error.txt - it can be changed to
your designated name
91.
92.     s = i
93.     Print #n, s           'save the error line number to error.txt
94.     Close #n
95.     MsgBox "Error on Line : " & s 'A pop up box that shows error line number
96. End Sub
97.
98. Sub Button1_Click()
99.     DownloadFilefromURL
100.     End Sub

```