

ABOUT THE AUTHOR

A two-time Microsoft MVP & lifetime Excel enthusiast, Randy Austin founded Excel For Freelancers in 2017. Excel For Freelancers quickly became the most prominent resource Excel for developers to learn how to turn their passion for Excel into profits by building & selling their own excel-based applications for passive & recurring income.

With nearly 300,000 YouTube subscribers, 14,000,000 video views, 200+ comprehensive training videos, and a thriving 40,000 member Facebook community, Excel For Freelancers has positioned itself as the #1 Excel developers resource in the world.

Get free content, training, and downloads just by clicking any of the free resources below:



[WEBSITE](#)



[YOUTUBE](#)



[FACEBOOK](#)



[TWITTER](#)



[INSTAGRAM](#)



[TELEGRAM](#)



[RUMBLE](#)



Microsoft®
Most Valuable
Professional



OUR COURSES & PRODUCTS



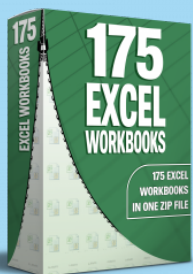
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[Click here to learn more](#)



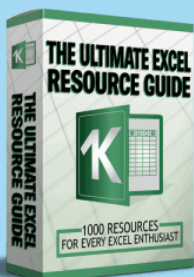
16 hour masterclass that will teach you the tips, tricks and techniques on how to create a dynamic single-click dashboard, and a ton more

[Click here to learn more](#)



Incredible Package of 175 of my BEST Applications into a SINGLE ZIP File which also includes the "175 Workbook Library".

[Click here to learn more](#)



With 1000 live links, continuously updating content, sort-able and filterable items, you will always have exactly what you need, when you need it.

[Click here to learn more](#)

Projects.....	3
VBAProject	3
Documents	3
Sheet1.....	3
(Declarations).....	3
Worksheet_BeforeDoubleClick [Sub]	3
Worksheet_Change [Sub]	3
Worksheet_SelectionChange [Sub]	4
Sheet2.....	7
(Declarations).....	7
Sheet3.....	9
(Declarations).....	9
ThisWorkbook	11
(Declarations).....	11
Modules.....	13
CodeRuntimeMacros.....	13
(Declarations).....	13
ResetCalc [Sub]	13
StartClock [Sub]	13
StopCalc [Sub]	13
StopClock [Sub]	13
DrivesMacros	15
(Declarations).....	15
DisplayDrives [Sub]	15
FadeOutMessages	17
(Declarations).....	17
FilesMovedMess [Sub]	17
FoldDefaultMess [Sub]	17
FileMacros.....	19
(Declarations).....	19
CopySelectedFiles [Sub]	19
DeleteSelectedFiles [Sub]	19
OpenFile [Sub]	19
PasteSelectedFiles [Sub]	20
ReloadFolder [Sub]	20
ResetFileFormats [Sub]	21
SelectedFilesFilter [Sub]	21
FilePreview.....	23
(Declarations).....	23
DocumentPreview [Sub]	23
FilterMacros	26
(Declarations).....	26
Filter_Clear [Sub]	26
Filter_Run [Sub]	26
FolderMacros	28
(Declarations).....	28
CloseFolder [Sub]	28
CreateNewFolder [Sub]	29
DeleteFolder [Sub]	29
ExpandFolder [Sub]	30
GoToHomeFolder [Sub]	31
GoToNextFolder [Sub]	32
GoToParentFolder [Sub]	32
GoToPrevFolder [Sub]	32
OpenFolder [Sub]	32
SetAsDefaultFolder [Sub]	33
Module1	35
(Declarations).....	35
Macro1 [Sub]	35
Macro2 [Sub]	35
MoveFilesMac	37
(Declarations).....	37
MoveFiles [Sub]	37
OptionMacros	39
(Declarations).....	39
ShowHideColumns [Sub]	39

ShowHideOptions [Sub]	39
SortingMacros	41
(Declarations).....	41
FileSort [Sub]	41
SetDefaultSort [Sub]	41
TabMacros	43
(Declarations).....	43
CloseTab [Sub]	43
OpenNewTab [Sub].....	43
SwitchTab [Sub].....	43

```

1 Option Explicit
2
3 Dim objFSO, objFolder As Object
4 Dim FilePath, NewFolder, FileName As String
5
6 Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
7     If Not Intersect(Target, Range("S9:X1000" )) Is Nothing And Range("B2" ).Value = False
8         Then OpenFile
9     End Sub
10
11 Private Sub Worksheet_Change(ByVal Target As Range)
12     If Target.Cells.CountLarge > 1 Then Exit Sub
13
14     If Not Intersect(Target, Range("S9:X1000" )) Is Nothing And Range("B2" ).Value = False
15         Then 'On File name change but not on load files
16             Application.CutCopyMode = False
17             If ActiveCell.Column < 4 Or ActiveCell.Column > 10 Or ActiveCell.Row < 9 Then 'On Drag
18                 & Drop Out of Proper Range
19                 If ActiveCell.Column <> 19 Then
20                     Application.EnableEvents = False
21                     Application.Undo
22                     Application.EnableEvents = True
23                     ResetCalc
24                 End
25             End If
26         End If
27
28         If Target.Value <> Empty And Target.Column = 19 Then 'On Filename Change
29             Set objFSO = CreateObject("Scripting.FileSystemObject" )
30             objFSO.MoveFile Range("X" & Target.Row).Value, Range("B5" ).Value & "\" & Target.
31             Value
32             Range("X" & Target.Row).Value = Range("B5" ).Value & "\" & Target.Value 'Update
33             File Path
34         End If
35
36         If ActiveCell.Column <> 19 Then 'Drag n Drop
37             Set objFSO = CreateObject("Scripting.FileSystemObject" )
38             Range("B2" ).Value = True
39             FilePath = Range("X" & Target.Row).Value 'Current File Path
40             NewFolder = Range("B" & ActiveCell.Row).Value 'New Folder
41             If Target.Row = 19 Then FileName = Target.Value Else: FileName = Range("S" & Target
42             .Row).Value 'FileName
43             'On Error GoTo FileExists
44             objFSO.MoveFile FilePath, NewFolder & "\" & FileName
45             ActiveCell.Value = Sheet2.Range("CloseFold" ).Value
46             ResetFileFormats
47             Range("BackupFoldersRng" ).Copy
48             Range("D9" ).PasteSpecial xlPasteFormats
49             Application.CutCopyMode = False
50             Range("B2" ).Value = False
51             StopCalc
52             ReloadFolder
53             ResetCalc
54         End If
55     End If
56
57     'Change Folder Name
58     If Not Intersect(Target, Range("D9:O1000" )) Is Nothing And Range("B2" ).Value = False
59         Then 'On Folder Name Change
60             If Target.Offset(0, -1).Value = Sheet2.Range("OpenFold" ).Value Or Target.Offset(0, -1)
61             .Value = Sheet2.Range("CloseFold" ).Value Then

```

1 2 3

```

54   1 2 3 Dim OldFolderName, NewFolderName As String
55   Set objFSO = CreateObject("Scripting.FileSystemObject" )
56   Set objFolder = objFSO.GetFolder(Range("B" & Target.Row).Value)
57   OldFolderName = Range("B" & Target.Row).Value 'Old Folder Name
58   NewFolderName = objFolder.ParentFolder & "\" & Target.Value
59   'On Error GoTo FileOpenMsg
60   Name OldFolderName As NewFolderName
61   Range("B" & Target.Row).Value = NewFolderName 'Update Path
62   Application.SendKeys "{numlock}%s" , True 'Reset Number Lock on Keyboard
63   End If
64   End If
65
66   'Run Filter on cell change
67   If Not Intersect(Target, Range("S7:W7" )) Is Nothing Then
68       On Error Resume Next 'Bug's can appear on Clearing filter if this is not added
69       If InStr(Target.Value, ":" ) = 0 Then Filter_Run
70   End If
71   Exit Sub
72   FileExists:
73   ResetCalc
74   Range("B2" ).Value = False
75   MsgBox "The selected file arleady exists"
76
77   Exit Sub
78   FileOpenMsg:
79   MsgBox "Folder cannot be renamed becuse a file or folder may be open in another program
or file manager"
80   End Sub
81
82   Private Sub Worksheet_SelectionChange(ByVal Target As Range)
83       If Target.Cells.CountLarge > 4 Then Exit Sub
84       If Target.Cells.CountLarge > 1 And Target.Row <> 4 Then Exit Sub
85       'On Selection Change of Folder area
86       If Not Intersect(Target, Range("D9:O1000" )) Is Nothing And Range("B2" ).Value = False
Then 'On Folder Selection
87           If Target.Value = Sheet2.Range("CloseFold" ).Value Then ExpandFolder
88           If Target.Value = Sheet2.Range("OpenFold" ).Value Then CloseFolder
89           If Target.Value = Sheet2.Range("DriveIcon" ).Value Then
90               Range("B5" ).Value = Range("B" & Target.Row).Value
91               OpenFolder
92           End If
93       End If
94
95       'On Selection Change of File Checkbox
96       If Not Intersect(Target, Range("Q9:Q1000" )) Is Nothing And Range("S" & Target.Row).Value
<> Empty Then
97           If Target.Value = Sheet2.Range("UncheckIcon" ).Value Then Target.Value = Sheet2.Range(
"CheckIcon" ).Value Else: Target.Value = Sheet2.Range("UncheckIcon" ).Value
98           Range("Q7" ).Value = Sheet2.Range("MixedIcon" ).Value 'Set Checkbox status to mixed
99           Range("A1" ).Select 'Unselect the current cell
100      End If
101      'On Selection change of any field
102      If Not Intersect(Target, Range("R9:W1000" )) Is Nothing And Range("B2" ).Value = False And
Range("S" & Target.Row).Value <> Empty Then 'And Target.Value <> Empty Then
103          Range("Q9:Q" & Range("Q9999" ).End(xlUp).Row).Value = Sheet2.Range("UncheckIcon" ).
Value
104          Range("B1" ).Value = Target.Row
105          Range("Q" & Target.Row).Value = Sheet2.Range("CheckIcon" )
106          Range("AP5" ).Value = "Open: " & Range("R" & Target.Row).Value
107          If Range("R" & Target.Row).Value = Sheet2.Range("PicIcon" ).Value And Range("Z27" ).
Value = Sheet2.Range("CheckIcon" ).Value Then DocumentPreview 'On Picture Selection

```



```

1 2
108 If Range("R" & Target.Row).Value = Sheet2.Range("WordIcon" ).Value And Range("Z28" ).
Value = Sheet2.Range("CheckIcon" ).Value Then DocumentPreview 'On Word Selection
109 If Range("R" & Target.Row).Value = Sheet2.Range("PDFIcon" ).Value And Range("Z29" ).
Value = Sheet2.Range("CheckIcon" ).Value Then DocumentPreview 'On PDF Selection
110 If Range("R" & Target.Row).Value = Sheet2.Range("ExcelIcon" ).Value And Range("Z30" ).
Value = Sheet2.Range("CheckIcon" ).Value Then DocumentPreview 'On Excel Selection
111 If Range("R" & Target.Row).Value = Sheet2.Range("VideoIcon" ).Value And Range("Z31" ).
Value = Sheet2.Range("CheckIcon" ).Value Then DocumentPreview 'On Video Selection
112 If Range("R" & Target.Row).Value = Sheet2.Range("AudioIcon" ).Value And Range("Z32" ).
Value = Sheet2.Range("CheckIcon" ).Value Then DocumentPreview 'On Audio Selection
113
114 End If
115
116 'On Selection Change of any column header
117 If Not Intersect(Target, Range("S6:W6" )) Is Nothing Then FileSort
118
119 If Not Intersect(Target, Range("Z11:Z15" )) Is Nothing Then ShowHideColumns
120
121 If Not Intersect(Target, Range("Z17:Z22" )) Is Nothing Then SetDefaultSort
122 If Not Intersect(Target, Range("Z24:Z25,Z27:Z32" )) Is Nothing Then
123 If Target.Value = Sheet2.Range("CheckIcon" ).Value Then Target.Value = Sheet2.Range(
"UncheckIcon" ).Value Else: Target.Value = Sheet2.Range("CheckIcon" ).Value
124 End If
125
126
127 'Select All/Note
128 If Not Intersect(Target, Range("Q7" )) Is Nothing Then
129 Dim LastRow As Long
130 LastRow = Range("Q9999" ).End(xlUp).Row
131 If Target.Value = Sheet2.Range("CheckIcon" ).Value Then
132 Range("Q7,Q9:Q" & LastRow).Value = Sheet2.Range("UncheckIcon" ).Value
133 Else:
134 Range("Q7,Q9:Q" & LastRow).Value = Sheet2.Range("CheckIcon" ).Value
135 End If
136 Range("A1" ).Select
137 End If
138
139 'Tab Selections
140 If Not Intersect(Target, Range("D4:P4" )) Is Nothing Then
141 'On Error Resume Next
142 If Cells(4, Target.Column).Value = Sheet2.Range("AddTabIcon" ).Value Then
143 OpenNewTab
144 Else:
145 SwitchTab
146 End If
147
148 End If
149
150 End Sub
151
152

```


A

ActiveCell, 3
Application, 3, 4

C

Cancel, 3
Cells, 3-5
CloseFolder, 4
Column, 3, 5
Copy, 3
CountLarge, 3, 4
CreateObject, 3, 4
CutCopyMode, 3

D

DocumentPreview, 4, 5

E

Empty, 3, 4
EnableEvents, 3
ExpandFolder, 4
Explicit, 3

F

FileExists, 4
FileName, 3
FileOpenMsg, 4
FilePath, 3
FileSort, 5
Filter_Run, 4

G

GetFolder, 4

I

InStr, 4
Intersect, 3-5

L

LastRow, 5

M

MoveFile, 3
MsgBox, 4

N

Name, 4
NewFolder, 3
NewFolderName, 4

O

objFolder, 3, 4
objFSO, 3, 4
Offset, 3
OldFolderName, 4
OpenFile, 3
OpenFolder, 4
OpenNewTab, 5

P

ParentFolder, 4
PasteSpecial, 3

R

Range, 3-5
ReloadFolder, 3
ResetCalc, 3, 4
ResetFileFormats, 3
Row, 3-5

S

SendKeys, 4
SetDefaultSort, 5
Sheet2, 3-5
ShowHideColumns, 5
StopCalc, 3
SwitchTab, 5

T

Target, 3-5

U

Undo, 3

V

Value, 3-5

W

Worksheet_BeforeDoubleClick, 3
Worksheet_Change, 3
Worksheet_SelectionChange, 4

X

xlPasteFormats, 3
xlUp, 4, 5

1 Option Explicit

2

E

Explicit, 7

1 Option Explicit

2

E

Explicit, 9

1 Option Explicit
2

E

Explicit, [11](#)


```
1 Option Explicit
2 Dim StartTime, SecondsElapsed As Double
3
4 Sub ResetCalc() 'This macro reset Calculations, screen updating and events and should be
  used at the end of a macro when StopCalc has been used
5   With Application
6     .EnableEvents = True 'Re-enabled Events (optional)
7     .Calculation = xlCalculationAutomatic 'Resets workbook Calculations to Automatic
8     .ScreenUpdating = True 'Turns on Screen Updating
9   End With
10 End Sub
11
12
13
14 'Start Clock and StopClock Macros can be used inside other macros to set a Time on Macros.
  This helps test the speed of macros, so when you make changes, you can see how it affects
  the speed of those macros
15
16 Sub StartClock() 'Starts the Timer
17   StartTime = Timer 'Starts Timer
18 End Sub
19
20 Sub StopCalc() 'This macro stops calculations and updating. Disabling events can be
  included as well. By adding this macro, at the beginning of a Macro then using the ResetCalc,
  your code is sped up considerably
21   With Application
22     .EnableEvents = False 'Turns Off Enabling Events '(optional)
23     .Calculation = xlCalculationManual 'Sets Workbook Calculation to Manual
24     .ScreenUpdating = False 'Turns Off Screen Updating
25   End With
26 End Sub
27
28 Sub StopClock()
29   SecondsElapsed = Round(Timer - StartTime, 2) 'Determine how many seconds code took to run
30   MsgBox "This code ran successfully in " & SecondsElapsed & " seconds" , vbInformation
  'Notifies the user how long the macro took to run (in seconds)
31 End Sub
32
33
```

A

Application, [13](#)

C

Calculation, [13](#)

E

EnableEvents, [13](#)

Explicit, [13](#)

M

MsgBox, [13](#)

R

ResetCalc, [13](#)

Round, [13](#)

S

ScreenUpdating, [13](#)

SecondsElapsed, [13](#)

StartClock, [13](#)

StartTime, [13](#)

StopCalc, [13](#)

StopClock, [13](#)

T

Timer, [13](#)

V

vbInformation, [13](#)

X

xlCalculationAutomatic, [13](#)

xlCalculationManual, [13](#)

```
1 Option Explicit
2
3 Sub DisplayDrives()
4 Dim objFSO, objDrive, objName As Object
5 Dim RowNum As Long
6 Set objFSO = CreateObject("Scripting.FileSystemObject")
7 With Sheet1
8 StopCalc
9 .Range("B2").Value = True 'Set Folder Load to True
10 .Range("A9:B9999,D9:O9999,Q9:Q9999,S9:X9999").ClearContents 'Clear Existing Files &
    Folders
11 RowNum = 9
12 For Each objDrive In objFSO.Drives
13 Set objName = objFSO.GetDrive(objFSO.GetDriveName(objDrive))
14 On Error GoTo NextDrive
15 .Range("B" & RowNum).Value = objDrive.Path & "\"
16 .Range("D" & RowNum).Value = Sheet2.Range("DriveIcon").Value
17 .Range("E" & RowNum).Value = Format((objDrive.TotalSize - objDrive.FreeSpace) /
    objDrive.TotalSize, "percent") 'Show Percent Full
18 .Range("E" & RowNum & ":H" & RowNum).Merge
19 .Range("I" & RowNum).Value = objDrive.VolumeName & " (" & objDrive.DriveLetter &
    ":)"
20 RowNum = RowNum + 1
21 NextDrive:
22 Next objDrive
23 .Range("B5,H5").Value = "My Computer"
24 .Range("G5").Value = Sheet2.Range("MyComputer").Value
25 ResetCalc
26 .Range("B2").Value = False 'Set Folder Load to False
27 .Range("A1").Select
28 End With
29 End Sub
30
```

C

ClearContents, [15](#)
CreateObject, [15](#)

D

DisplayDrives, [15](#)
DriveLetter, [15](#)
Drives, [15](#)

E

Explicit, [15](#)

F

Format, [15](#)
FreeSpace, [15](#)

G

GetDrive, [15](#)
GetDriveName, [15](#)

M

Merge, [15](#)

N

NextDrive, [15](#)

O

objDrive, [15](#)
objFSO, [15](#)
objName, [15](#)

P

Path, [15](#)

R

Range, [15](#)
ResetCalc, [15](#)
RowNum, [15](#)

S

Sheet1, [15](#)
Sheet2, [15](#)
StopCalc, [15](#)

T

TotalSize, [15](#)

V

Value, [15](#)
VolumeName, [15](#)

Option Explicit

```
1
2
3
4 Sub FilesMovedMess()
5     Dim i As Integer
6     Dim delay As Double
7     Dim StartTime As Double
8     With ActiveSheet.Shapes("FilesMovedMsg" )
9         .Visible = True
10        For i = 1 To 100
11            .Fill.Transparency = i / 100
12            delay = 0.01
13            StartTime = Timer
14            Do
15                DoEvents
16            Loop While Timer - StartTime < delay
17        Next i
18        .Visible = False
19    End With
20 End Sub
```

```
21
22
23
24
25
26
27
28 Sub FoldDefaultMess()
29     Dim i As Integer
30     Dim delay As Double
31     Dim StartTime As Double
32
33     With ActiveSheet.Shapes("SetDefaultMsg" )
34         .Visible = True
35         For i = 1 To 100
36             .Fill.Transparency = i / 100
37             delay = 0.008
38             StartTime = Timer
39             Do
40                 DoEvents
41             Loop While Timer - StartTime < delay
42         Next i
43         .Visible = False
44     End With
45 End Sub
```

A

ActiveSheet, [17](#)

D

delay, [17](#)

DoEvents, [17](#)

E

Explicit, [17](#)

F

FilesMovedMess, [17](#)

Fill, [17](#)

FoldDefaultMess, [17](#)

I

i, [17](#)

S

Shapes, [17](#)

StartTime, [17](#)

T

Timer, [17](#)

Transparency, [17](#)

V

Visible, [17](#)

```

1 Option Explicit
2 Dim LastRow, FileRow, LastFiltRow As Long
3 Dim Resp As Integer
4
5 Sub CopySelectedFiles()
6     With Sheet1
7         LastRow = .Range("S9999").End(xlUp).Row
8         If LastRow < 9 Then Exit Sub
9         .Range("EA3:EH999").ClearContents 'Clear any remaining Results
10        .Range("Q8:X" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range("DA2:DH3"
11        ), CopyToRange:=.Range("EA2:EH2"), Unique:=True
12    End With
13 End Sub
14
15 Sub DeleteSelectedFiles()
16     Dim DeleteFilename As String
17     With Sheet1
18         If .Range("Z24").Value = Sheet2.Range("CheckIcon").Value Then
19             Resp = MsgBox("Are you sure you want to delete the selected files?", vbYesNo,
20             "Delete Files")
21             If Resp = vbNo Then Exit Sub
22         End If
23         LastRow = .Range("S9999").End(xlUp).Row
24         If LastRow < 9 Then Exit Sub
25         .Range("BA3:BH99,CA3:CH999").ClearContents 'Clear any remaining Criteria Or Results
26         .Range("BA3").Value = Sheet2.Range("CheckIcon").Value
27         .Range("Q8:X" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range("BA2:BH3"
28         ), CopyToRange:=.Range("CA2:CH2"), Unique:=True
29         LastFiltRow = .Range("CH99999").End(xlUp).Row
30         If LastFiltRow < 3 Then
31             MsgBox "Please select files to delete first"
32             Exit Sub
33         End If
34         StopCalc
35         For FileRow = 3 To LastFiltRow
36             DeleteFilename = .Range("CH" & FileRow).Value 'Determine Source Filename
37             'On Error Resume Next
38             Kill (DeleteFilename)
39             'On Error GoTo 0
40             NextFile:
41         Next FileRow
42     End With
43     ReloadFolder
44     ResetCalc
45 End Sub
46
47 Sub OpenFile()
48     Dim LastRow, LastFiltRow, FiltRow As Long
49     With Sheet1
50         'Check for Multiple Selected Files
51         If Application.WorksheetFunction.CountIf(.Range("Q9:Q1000"), Sheet2.Range("CheckIcon"
52         )) > 1 Then
53             LastRow = .Range("S9999").End(xlUp).Row
54             .Range("BA3:BH9999,CA3:CH9999").ClearContents 'Clear any previous Filters
55             .Range("BA3").Value = Sheet2.Range("CheckIcon").Value 'Add icon for filtering
56             .Range("Q8:X" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
57             "BA2:BH3"), CopyToRange:=.Range("CA2:CH2"), Unique:=True
58             LastFiltRow = .Range("CC999").End(xlUp).Row 'Last Filtered Row
59             For FiltRow = 3 To LastFiltRow

```

1 2 3 4


```

1 2 3 4
57 ThisWorkbook.FollowHyperlink .Range("CH" & FiltRow).Value
58 Next FiltRow
59 Else:
60 ThisWorkbook.FollowHyperlink .Range("X" & ActiveCell.Row).Value
61 End If
62 End With
63 End Sub
64
65
66 Sub PasteSelectedFiles()
67 Dim SourceFilename, DestinationFilename As String
68 With Sheet1
69 LastRow = .Range("EC9999" ).End(xlUp).Row
70 If LastRow < 3 Then
71 MsgBox "No files have been copied to the clipboard to paste. Please select and copy
72 files to be pasted"
73 Exit Sub
74 End If
75 StopCalc
76 For FileRow = 3 To LastRow
77 SourceFilename = .Range("EH" & FileRow).Value 'Determine Source Filename
78 DestinationFilename = .Range("B5" ).Value & "\" & .Range("EC" & FileRow).Value 'Determine Destination Filename
79 'On Error Resume Next
80 If Dir(DestinationFilename) <> "" Then 'File already exists
81 'On Error GoTo 0
82 Resp = MsgBox("The file " & DestinationFilename & vbCrLf & "already exists.
83 Would you like to overwrite this file?" , vbYesNo, "File Exists" )
84 If Resp = vbNo Then GoTo NextFile
85 End If
86 'Two Different Copy Methods, use just one
87 FileCopy SourceFilename, DestinationFilename 'Simple File Copy Method
88 'Call ObjFile.CopyFile(SourceFilename, DestinationFilename, True) 'Using FSO Option
89 -True means Overwrite, False means don't overwrite
90 NextFile:
91 Next FileRow
92 ReloadFolder 'Reload Current Folder
93 End With
94 ResetCalc
95 End Sub
96
97
98
99 Sub ReloadFolder()
100 Dim ObjFile, objFSO, objFolder As Object
101 Set objFSO = VBA.CreateObject("Scripting.FileSystemObject" )
102
103 With Sheet1
104 .Range("B2" ).Value = True 'Seat Load Files True
105 .Range("Q9:Q9999,S9:X9999" ).ClearContents 'Clear Existing Folder to reload
106 FileRow = 9
107 Set objFolder = objFSO.GetFolder(.Range("B5" ).Value)
108 For Each ObjFile In objFolder.Files
109 .Range("Q" & FileRow).Value = Sheet2.Range("UncheckIcon" ).Value
110 .Range("S" & FileRow).Value = ObjFile.Name
111 .Range("T" & FileRow).Value = ObjFile.Type
112 .Range("U" & FileRow).Value = ObjFile.Size
113 .Range("V" & FileRow).Value = ObjFile.DateCreated
114 .Range("W" & FileRow).Value = ObjFile.DateLastModified
115 .Range("X" & FileRow).Value = ObjFile.Path
116 FileRow = FileRow + 1
117 Next ObjFile

```

```
1 2
114 .Range("A1" ).Select
115 .Range("B2" ).Value = False 'Set Load Files To False
116 ActiveWindow.ScrollRow = 9
117 End With
118 End Sub
119
120 Sub ResetFileFormats()
121 Sheet1.Range("BackupFilesRng" ).Copy
122 Sheet1.Range("R9" ).PasteSpecial xlPasteFormats
123 Sheet1.Range("FR9:FR1000" ).Copy
124 Sheet1.Range("R9" ).PasteSpecial xlPasteFormulas
125 End Sub
126
127 Sub SelectedFilesFilter()
128 With Sheet1
129 LastRow = .Range("S9999" ).End(xlUp).Row
130 If LastRow < 9 Then Exit Sub
131 .Range("BA3:BH99,CA3:CH999" ).ClearContents 'Clear any remaining Criteria Or Results
132 .Range("BA3" ).Value = Sheet2.Range("CheckIcon" ).Value
133 .Range("Q7:X" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range("BA2:BH3"
), CopyToRange:=.Range("CA2:CH2" ), Unique:=True
134 End With
135 End Sub
```

A

ActiveCell, 20
ActiveWindow, 21
AdvancedFilter, 19, 21
Application, 19

C

ClearContents, 19-21
Copy, 21
CopySelectedFiles, 19
CopyToRange, 19, 21
CountIf, 19
CreateObject, 20
CriteriaRange, 19, 21

D

DateCreated, 20
DateLastModified, 20
DeleteFilename, 19
DeleteSelectedFiles, 19
DestinationFilename, 20
Dir, 20

E

Explicit, 19

F

FileCopy, 20
FileRow, 19, 20
Files, 20
FiltRow, 19, 20
FollowHyperlink, 20

G

GetFolder, 20

K

Kill, 19

L

LastFiltRow, 19
LastRow, 19-21

M

MsgBox, 19, 20

N

Name, 20
NextFile, 19, 20

O

ObjFile, 20
objFolder, 20
objFSO, 20
OpenFile, 19

P

PasteSelectedFiles, 20
PasteSpecial, 21
Path, 20

R

Range, 19-21
ReloadFolder, 19, 20
ResetCalc, 19, 20
ResetFileFormats, 21
Resp, 19, 20
Row, 19-21

S

ScrollRow, 21
SelectedFilesFilter, 21
Sheet1, 19-21
Sheet2, 19-21
Size, 20
SourceFilename, 20
StopCalc, 19, 20

T

ThisWorkbook, 20

U

Unique, 19, 21

V

Value, 19-21
VBA, 20
vbCrLf, 20
vbNo, 19, 20
vbYesNo, 19, 20

W

WorksheetFunction, 19

X

xlFilterCopy, 19, 21
xlPasteFormats, 21
xlPasteFormulas, 21
xlUp, 19-21

Option Explicit

```

1
2
3
4 Sub DocumentPreview()
5     Dim FileName As String
6     Dim wmp As Object
7     StopCalc
8     With Sheet1
9         On Error Resume Next
10        '           With .WindowsMediaPlayer1
11        '               .Visible = False
12        '               .Left = Sheet1.Range("AC9").Left
13        '               .Top = Sheet1.Range("AC9").Top
14        '               .Width = 300
15        '               .Height = 218
16        '           End With
17        On Error Resume Next
18        .Shapes("PreviewFile" ).Delete 'Clear Any Previous Shape Preview (if any)
19        On Error GoTo 0
20        FileName = .Range("X" & ActiveCell.Row).Value
21        'On Word, Excel or PDF Section
22        If .Range("R" & ActiveCell.Row).Value = Sheet2.Range("WordIcon" ).Value Or _
23        .Range("R" & ActiveCell.Row).Value = Sheet2.Range("ExcelIcon" ).Value Or _
24        .Range("R" & ActiveCell.Row).Value = Sheet2.Range("PDFIcon" ).Value Then
25            On Error Resume Next
26            .OLEObjects.Add(FileName:=FileName, Link:=False, DisplayAsIcon:=False).Name =
27            "PreviewFile"
28            On Error GoTo 0
29            If .Range("R" & ActiveCell.Row).Value = Sheet2.Range("ExcelIcon" ).Value Then
30                Workbooks(Dir(FileName)).Close SaveChanges:=False
31            If .Range("R" & ActiveCell.Row).Value = Sheet2.Range("ExcelIcon" ).Value Then
32                Workbooks(2).Close SaveChanges:=False
33            End If
34            'On Picture Selection
35            If .Range("R" & ActiveCell.Row).Value = Sheet2.Range("PicIcon" ).Value Then
36                .Pictures.Insert(FileName).Name = "PreviewFile"
37                With .Shapes("PreviewFile" )
38                    If .Width > .Height Then .Width = 200 Else: .Height = 300
39                End With
40            End If
41            'On Video or Audio Selection
42            If .Range("R" & ActiveCell.Row).Value = Sheet2.Range("VideoIcon" ).Value Or _
43            .Range("R" & ActiveCell.Row).Value = Sheet2.Range("AudioIcon" ).Value Then
44                '           With .WindowsMediaPlayer1
45                '               .URL = FileName
46                '               .Left = Sheet1.Range("AC9").Left
47                '               .Top = Sheet1.Range("AC9").Top
48                '               .Width = 300
49                '               .Height = 218
50                '               .Visible = True
51                '           End With
52            Else:
53                With .Shapes("PreviewFile" )
54                    .Left = Sheet1.Range("AC9" ).Left + 15
55                    .Top = Sheet1.Range("AC9" ).Top + 10
56                    .IncrementLeft 10
57                    .IncrementTop 10
58                End With
59            End If
60            ResetCalc
61        End With

```

1

59
60

¹
End Sub

→, [23](#)

A

ActiveCell, [23](#)
Add, [23](#)

C

Close, [23](#)

D

Delete, [23](#)
Dir, [23](#)
DisplayAsIcon, [23](#)
DocumentPreview, [23](#)

E

Explicit, [23](#)

F

FileName, [23](#)

H

Height, [23](#)

I

IncrementLeft, [23](#)
IncrementTop, [23](#)
Insert, [23](#)

L

Left, [23](#)
Link, [23](#)

N

Name, [23](#)

O

OLEObjects, [23](#)

P

Pictures, [23](#)

R

Range, [23](#)
ResetCalc, [23](#)
Row, [23](#)

S

SaveChanges, [23](#)
Shapes, [23](#)
Sheet1, [23](#)
Sheet2, [23](#)
StopCalc, [23](#)

T

Top, [23](#)

V

Value, [23](#)

W

Width, [23](#)
wmp, [23](#)
Workbooks, [23](#)

Option Explicit

```
1  
2  
3  
4 Sub Filter_Clear()
```

```
5 StopCalc
```

```
6 With Sheet1
```

```
7 .Range("S7:W7").Value = .Range("AM7:AQ7").Value 'Reset Default Search Text
```

```
8 .Range("BA3:BH3,CA3:CA9999").ClearContents 'Clear any Previous Search Criteria or  
Results
```

```
9 .Shapes("ClearFiltBtn").Visible = msoFalse
```

```
10 End With
```

```
11 ReloadFolder
```

```
12 ResetCalc
```

```
13 End Sub
```

```
14  
15  
16 Sub Filter_Run()
```

```
17 Dim LastRow, LastFiltRow As Long
```

```
18 StopCalc
```

```
19 With Sheet1
```

```
20 ReloadFolder
```

```
21 .Range("B2").Value = True 'Set File Load To True
```

```
22 LastRow = .Range("S9999").End(xlUp).Row
```

```
23 .Range("CA3:CA9999").ClearContents
```

```
24 If InStr(.Range("S7").Value, ":") <> 0 Or .Range("S7").Value = Empty Then .Range(  
"BC3").ClearContents Else .Range("BC3").Value = "*" & .Range("S7").Value & "*" ←
```

```
'Either Clear or Set Name Filter
```

```
25 If InStr(.Range("T7").Value, ":") <> 0 Or .Range("T7").Value = Empty Then .Range(  
"BD3").ClearContents Else .Range("BD3").Value = "*" & .Range("T7").Value & "*" ←
```

```
'Either Clear or Set File Type Filter
```

```
26 If InStr(.Range("U7").Value, ":") <> 0 Or .Range("U7").Value = Empty Then .Range(  
"BE3").ClearContents Else .Range("BE3").Value = .Range("U7").Value 'Either Clear or ←
```

```
Set File Size
```

```
27 If InStr(.Range("V7").Value, ":") <> 0 Or .Range("V7").Value = Empty Then .Range(  
"BF3").ClearContents Else .Range("BF3").Value = .Range("V7").Value 'Either Clear ←
```

```
or Set Date Created
```

```
28 If InStr(.Range("W7").Value, ":") <> 0 Or .Range("W7").Value = Empty Then .Range(  
"BG3").ClearContents Else .Range("BG3").Value = .Range("W7").Value 'Either Clear ←
```

```
or Set Date Modified
```

```
29 .Range("Q8:X" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range("BA2:BH3"  
) , CopyToRange:=.Range("CA2:CH2"), Unique:=True ←
```

```
30 LastFiltRow = .Range("CC9999").End(xlUp).Row
```

```
31 .Range("Q9:Q9999,S9:X9999").ClearContents
```

```
32 .Shapes("ClearFiltBtn").Visible = msoCTrue
```

```
33 If LastFiltRow < 3 Then
```

```
34 .Range("B2").Value = False 'Set File Load To False
```

```
35 ResetCalc
```

```
36 Exit Sub
```

```
37 End If
```

```
38 .Range("S9:X" & LastFiltRow + 6).Value = .Range("CC3:CH" & LastFiltRow).Value
```

```
39 .Range("B2").Value = False 'Set File Load To False
```

```
40 End With
```

```
41 ResetCalc
```

```
42 End Sub
```


A

AdvancedFilter, [26](#)

C

ClearContents, [26](#)

CopyToRange, [26](#)

CriteriaRange, [26](#)

E

Empty, [26](#)

Explicit, [26](#)

F

Filter_Clear, [26](#)

Filter_Run, [26](#)

I

InStr, [26](#)

L

LastFiltRow, [26](#)

LastRow, [26](#)

M

msoCTrue, [26](#)

msoFalse, [26](#)

R

Range, [26](#)

ReloadFolder, [26](#)

ResetCalc, [26](#)

Row, [26](#)

S

Shapes, [26](#)

Sheet1, [26](#)

StopCalc, [26](#)

U

Unique, [26](#)

V

Value, [26](#)

Visible, [26](#)

X

xlFilterCopy, [26](#)

xlUp, [26](#)

```

1
2 Option Explicit
3 Dim objFSO, objFolder, ObjSubFolder, ObjFile As Object
4 Dim Rng, FoundLast As Range
5
6
7 Sub CloseFolder()
8     Dim SubfolderQty, RowNum, ItemNum, LastRow, ColNum, NextRow, FileRow, ActRow, FoldRow As
9     Long
10    Set objFSO = CreateObject("Scripting.FileSystemObject" )
11    StopCalc
12    With Sheet1
13        Set Rng = Sheet1.Range("D9:O1000" )
14        Set FoundLast = Rng.Find(What:="*" , After:=Rng.Cells(1), Lookat:=xlPart, LookIn:=
15        xlFormulas, SearchOrder:=xlByRows, SearchDirection:=xlPrevious, MatchCase:=False)
16        LastRow = FoundLast.Row
17        .Range("B2" ).Value = True 'Set Load Files True
18        ActiveCell.Value = Sheet2.Range("CloseFold" ).Value
19        RowNum = ActiveCell.Row
20        ColNum = ActiveCell.Column
21        If ActiveCell.Offset(1, 0).Value <> Empty Then GoTo SkipCopyPaste
22        NextRow = ActiveCell.Offset(0, 0).End(xlDown).Row - 1
23        If NextRow > 9999 Then 'Closing Last Folder
24            .Range("B" & RowNum + 1 & ":B" & LastRow).ClearContents
25            .Range("D" & RowNum + 1 & ":O" & LastRow).ClearContents
26            GoTo SkipCopyPaste
27        End If
28        .Range("D" & NextRow + 1 & ":O" & LastRow).Copy
29        .Range("D" & RowNum + 1).PasteSpecial xlPasteValues
30        .Range("B" & NextRow + 1 & ":B" & LastRow).Copy
31        .Range("B" & RowNum + 1).PasteSpecial xlPasteValues
32        .Range("D" & LastRow - NextRow + RowNum + 1 & ":O" & LastRow).ClearContents
33        .Range("B" & LastRow - NextRow + RowNum + 1 & ":B" & LastRow).ClearContents
34        SkipCopyPaste:
35        FileRow = 9
36        .Range("Q9:Q9999,S9:X9999" ).ClearContents 'Clear Existing Folder to reload
37        On Error GoTo NoPath
38        Set objFolder = objFSO.GetFolder(.Range("B" & RowNum).Value)
39        On Error Resume Next
40        For Each ObjFile In objFolder.Files
41            .Range("Q" & FileRow).Value = Sheet2.Range("UncheckIcon" ).Value
42            .Range("S" & FileRow).Value = ObjFile.Name
43            .Range("T" & FileRow).Value = ObjFile.Type
44            .Range("U" & FileRow).Value = ObjFile.Size
45            .Range("V" & FileRow).Value = ObjFile.DateCreated
46            .Range("W" & FileRow).Value = ObjFile.DateLastModified
47            .Range("X" & FileRow).Value = ObjFile.Path
48            FileRow = FileRow + 1
49        Next ObjFile
50        .Range("A1" ).Select
51        .Range("B5" ).Value = .Range("B" & RowNum).Value 'Set Current Folder
52        ActiveWindow.ScrollRow = 9
53        ResetCalc
54        .Range("H5" ).Value = Replace(.Range("B5" ).Value, "\", " > " ) 'FilePath
55        .Range("B2" ).Value = False 'Set Load Files To False
56        .Range("A1" ).Select
57    End
58
59    ResetCalc
60    Exit Sub
61    NoPath:

```

```

1 2
60 .Range("B2" ).Value = False 'Set Load Files To False
61 ResetCalc
62 'MsgBox "This folder may have been moved or deleted. Please refresh the folder list"
63 End With
64 End Sub
65
66 Sub CreateNewFolder()
67 Dim ParentFolder As String
68 Dim FoldRng, FindParentRng As Range
69 Dim SubfolderQty, RowNum, ItemNum, LastRow, ColNum, NextRow, FileRow, ActRow, FoldRow As
Long
70 With Sheet1
71 .Range("B2" ).Value = True 'Set Folder Load to True
72 Set objFSO = CreateObject("Scripting.FileSystemObject" )
73 Set FoldRng = .Range("B8:B" & .Range("B99999" ).End(xlUp).Row) 'Set Current Folder
Range
74 ParentFolder = .Range("B5" ).Value
75 On Error GoTo NoPath
76 Set objFolder = objFSO.CreateFolder(ParentFolder & "\New Folder" )
77 Set FindParentRng = FoldRng.Find(ParentFolder, , xlValues, xlWhole)
78 Set Rng = .Range("D9:O999" )
79 Set FoundLast = Rng.Find(What:="*", After:=Rng.Cells(1), Lookat:=xlPart, LookIn:=
xlFormulas, SearchOrder:=xlByRows, SearchDirection:=xlPrevious, MatchCase:=False)
80 LastRow = FoundLast.Row
81 If Not FindParentRng Is Nothing Then
82 FoldRow = FindParentRng.Row
83 .Range("D" & FoldRow + 1 & ":O" & LastRow).Copy
84 .Range("D" & FoldRow + 2).PasteSpecial xlPasteValues
85 .Range("A" & FoldRow + 1 & ":B" & LastRow).Copy
86 .Range("A" & FoldRow + 2).PasteSpecial xlPasteValues
87 Application.CutCopyMode = False
88 Else:
89 FoldRow = LastRow 'Already At Top Level, Add New Folder on Last Row
90 End If
91 .Range("B2" ).Value = False 'Set Folder Load to False
92 .Range("B" & FoldRow + 1).Value = ParentFolder & "\New Folder"
93 .Range("A" & FoldRow + 1).Value = False 'New Folder contains no subfolders
94 ColNum = .Range("O" & FoldRow + 1).End(xlToLeft).Column
95 If ColNum < 5 Then ColNum = 5
96 If ColNum <> 5 Then .Cells(FoldRow + 1, ColNum - 1).ClearContents 'clear any icons
97 .Cells(FoldRow + 1, ColNum - 1).Value = Sheet2.Range("CloseFold" ).Value 'Add Folder
icon is necessary
98 .Cells(FoldRow + 1, ColNum).Value = "New Folder"
99 .Cells(FoldRow + 1, ColNum).Select
100 Application.Wait Now + 0.00001
101 Application.SendKeys "{BACKSPACE}"
102 Application.SendKeys "^{z}"
103 Exit Sub
104 NoPath:
105 .Range("B2" ).Value = False 'Set Folder Load to False
106 MsgBox "Please check that 'New Folder' does not already exist, or select a drive or
folder to add a new folder"
107 End With
108 End Sub
109
110
111 Sub DeleteFolder()
112 Dim FoldPath As String
113 Dim Resp As Integer
114 With Sheet1
115 If .Range("Z25" ).Value = Sheet2.Range("CheckIcon" ).Value Then
1 2 3

```

```

1 2 3
116     Resp = MsgBox("Are you sure you want to delete the selected folder and all of it's
117         contents?" , vbYesNo, "Delete Files" )
118     If Resp = vbNo Then Exit Sub
119     End If
120     Set objFSO = CreateObject("Scripting.FileSystemObject" )
121     FoldPath = .Range("B5" ).Value 'Folder Path
122     GoToParentFolder
123     objFSO.DeleteFolder FoldPath
124     GoToParentFolder
125     End With
126
127 End Sub
128
129
130 Sub ExpandFolder()
131     Dim SubfolderQty, RowNum, ItemNum, LastRow, ColNum, NextRow, FileRow, ActRow, FoldRow,
132     HeadCol As Long
133     Set objFSO = CreateObject("Scripting.FileSystemObject" )
134     StopCalc
135     With Sheet1
136         Set Rng = .Range("D9:O1000" )
137         Set FoundLast = Rng.Find(What:="*" , After:=Rng.Cells(1), Lookat:=xlPart, LookIn:=
138             xlFormulas, SearchOrder:=xlByRows, SearchDirection:=xlPrevious, MatchCase:=False)
139         .Range("B2" ).Value = True 'Set Load Files True
140         .Range("Q9:Q9999,S9:X9999" ).ClearContents 'Clear Files
141         On Error Resume Next
142         .Shapes("PreviewFile" ).Delete
143         On Error GoTo 0
144         If ActiveCell.Value <> Sheet2.Range("DriveIcon" ).Value Then ActiveCell.Value = Sheet2.
145             Range("OpenFold" ).Value
146             RowNum = ActiveCell.Row
147             ActRow = ActiveCell.Row
148             ColNum = ActiveCell.Column + 1
149             LastRow = FoundLast.Row
150             Sheet1.Range("B3" ).Value = .Range("B5" ).Value 'Set Previous Folder
151             .Range("B5" ).Value = .Range("B" & ActiveCell.Row).Value
152             On Error GoTo NoPath
153             Set objFolder = objFSO.GetFolder(.Range("B5" ).Value)
154
155             Set ObjSubFolder = objFolder.Subfolders
156             On Error Resume Next
157             SubfolderQty = ObjSubFolder.Count
158             If Err.Number <> 0 Then
159                 'On Error GoTo SkipFolder
160
161                 For Each ObjSubFolder In objFolder.Subfolders
162                     SubfolderQty = SubfolderQty + 1
163                 Next ObjSubFolder
164             End If
165             'On Error GoTo 0
166             If SubfolderQty < 1 Then GoTo SkipExpand
167             .Range("D" & RowNum + 1 & ":O" & LastRow).Copy
168             .Range("D" & RowNum + 1 + SubfolderQty & ":O" & LastRow + SubfolderQty).PasteSpecial
169             xlPasteValues
170             .Range("A" & RowNum + 1 & ":B" & LastRow).Copy
171             .Range("A" & RowNum + 1 + SubfolderQty & ":B" & LastRow + SubfolderQty).PasteSpecial
172             xlPasteValues
173             Application.CutCopyMode = False
174             Cells(RowNum, ColNum).Select
175             .Range("D" & RowNum + 1 & ":O" & RowNum + SubfolderQty).ClearContents

```

1 2

```

171
172 For Each ObjSubFolder In objFolder.Subfolders
173     .Cells(RowNum + 1, ColNum).Value = Sheet2.Range("CloseFold" ).Value
174     .Cells(RowNum + 1, ColNum + 1).Value = ObjSubFolder.Name
175     .Range("B" & RowNum + 1).Value = ObjSubFolder.Path
176     If ObjSubFolder.Subfolders.Count = 0 Then .Range("A" & RowNum + 1).Value = False
177     Else: .Range("A" & RowNum + 1).Value = True
178     RowNum = RowNum + 1
179 Next ObjSubFolder
180
181 SkipExpand:
182 FileRow = 9
183 Set objFolder = objFSO.GetFolder(.Range("B" & ActRow).Value)
184 For Each ObjFile In objFolder.Files
185     .Range("Q" & FileRow).Value = Sheet2.Range("UncheckIcon" ).Value
186     .Range("S" & FileRow).Value = ObjFile.Name
187     .Range("T" & FileRow).Value = ObjFile.Type
188     .Range("U" & FileRow).Value = ObjFile.Size
189     .Range("V" & FileRow).Value = ObjFile.DateCreated
190     .Range("W" & FileRow).Value = ObjFile.DateLastModified
191     .Range("X" & FileRow).Value = ObjFile.Path
192     FileRow = FileRow + 1
193 Next ObjFile
194 .Range("B5" ).Value = .Range("B" & ActRow).Value 'Set Current Folder
195 .Range("H5" ).Value = Replace(.Range("B5" ).Value, "\", " > " ) 'FilePath
196
197 'Sort Columns Based On Defaults
198 Dim SortRng As Range
199 Set SortRng = .Range("Z17:Z22" ).Find(Sheet2.Range("CheckIcon" ).Value, , xlValues,
200 xlWhole)
201 If Not SortRng Is Nothing Then
202     For HeadCol = 19 To 23 'Set loop for each column header to remove any Ascending or
203     Descending
204     .Cells(6, HeadCol).Value = Replace(Replace(Replace(Replace(.Cells(6, HeadCol).
205     Value, " " & Sheet2.Range("SortAlphaAZ" ).Value, "" ), " " & Sheet2.Range(
206     "SortAlphaZA" ).Value, "" ), " " & Sheet2.Range("SortNum19" ).Value, "" ), " "
207     & Sheet2.Range("SortNum91" ).Value, "" ) 'Remove any Ascending or Descending
208     Alpha or Numerical Icons
209 Next HeadCol
210 .Cells(6, SortRng.Row + 2).Select
211 End If
212
213 .Range("A1" ).Select
214 .Range("B2" ).Value = False 'Set Load Files To False
215 ActiveWindow.ScrollRow = 9
216
217 ResetCalc
218 End
219 Exit Sub
220 NoPath:
221 .Range("B2" ).Value = False
222 ResetCalc
223 'MsgBox "This folder may have been moved or deleted. Please refresh the folder list"
224 End With
225 End Sub
226
227 Sub GoToHomeFolder()
228 Sheet1.Range("B3" ).Value = Sheet1.Range("B5" ).Value 'Set Previous Folder
229 Sheet1.Range("B5" ).Value = Sheet1.Range("B6" ).Value 'Set Home Folder as Current folder
230
231 OpenFolder

```

```

1
224 End Sub
225 Sub GoToNextFolder()
226     Sheet1.Range("B5").Value = Sheet1.Range("B3").Value 'Set Current folder
227     Sheet1.Range("B4").Value = Sheet1.Range("B3").Value 'Set Next Folder
228     Sheet1.Range("B3").Value = Sheet1.Range("B5").Value 'Set Previous Folder
229     OpenFolder
230 End Sub
231 Sub GoToParentFolder()
232     Set objFSO = CreateObject("Scripting.FileSystemObject")
233     On Error GoTo HighestLevel
234     Set objFolder = objFSO.GetFolder(Sheet1.Range("B5").Value)
235     Sheet1.Range("B3").Value = Sheet1.Range("B5").Value 'Set Previous Folder
236     On Error GoTo DriveLoc
237     Sheet1.Range("B5").Value = objFolder.ParentFolder.Path
238     OpenFolder
239     Exit Sub
240 DriveLoc:
241     DisplayDrives
242     Exit Sub
243 HighestLevel:
244     MsgBox "You are at the the drive level"
245 End Sub
246 Sub GoToPrevFolder()
247     If Sheet1.Range("B3").Value = Empty Then
248         MsgBox "You are at the last record" 'Future Versions of this applicaiton save a
                history of these folders
249         Exit Sub
250     End If
251     Sheet1.Range("B5").Value = Sheet1.Range("B3").Value 'Set Current folder
252     Sheet1.Range("B4").Value = Sheet1.Range("B3").Value 'Set Next Folder
253     OpenFolder
254 End Sub
255
256 Sub OpenFolder()
257     Dim SubfolderQty, RowNum, ItemNum, LastRow, ColNum, NextRow, FileRow, ActRow, FoldRow,
                HeadCol As Long
258     StopCalc
259     Set objFSO = CreateObject("Scripting.FileSystemObject")
260     With Sheet1
261         .Range("B2").Value = True 'Set Folder Load to True
262         Set objFolder = objFSO.GetFolder(.Range("B5").Value)
263         Set ObjSubFolder = objFolder.Subfolders
264         On Error Resume Next
265         SubfolderQty = ObjSubFolder.Count
266         .Range("G5").Value = Sheet2.Range("CloseFold").Value 'Set Icon
267         RowNum = 9
268         .Range("A9:B9999,D9:09999,Q9:Q9999,S9:X9999").ClearContents
269         On Error Resume Next
270         .Shapes("PreviewFile").Delete 'Remove any Preview File
271         On Error GoTo 0
272         .Range("E9:H99").UnMerge
273         .Range("FD9:FP1000").Copy 'Added in Part 2
274         .Range("D9").PasteSpecial xlPasteFormats 'Added in Part 2
275         Application.CutCopyMode = False 'Added in Part 2
276         Sheet1.Range("B3").Value = Sheet1.Range("B5").Value 'Set Previous Folder
277         On Error GoTo NextFold
278         For Each ObjSubFolder In objFolder.Subfolders
279             .Range("D" & RowNum).Value = Sheet2.Range("CloseFold").Value
280             .Range("E" & RowNum).Value = ObjSubFolder.Name
281             .Range("B" & RowNum).Value = ObjSubFolder.Path
282             On Error Resume Next

```

1 2 3

```

1 2 3
283     If ObjSubFolder.Subfolders.Count = 0 Then .Range("A" & RowNum).Value = False Else:
284         .Range("A" & RowNum).Value = True
285     NextFold:
286     RowNum = RowNum + 1
287     Next ObjSubFolder
288     FileRow = 9
289     For Each ObjFile In objFolder.Files
290         .Range("X" & FileRow).Value = ObjFile.Path
291         .Range("Q" & FileRow).Value = Sheet2.Range("UncheckIcon" ).Value
292         .Range("S" & FileRow).Value = ObjFile.Name
293         .Range("T" & FileRow).Value = ObjFile.Type
294         .Range("U" & FileRow).Value = ObjFile.Size
295         .Range("V" & FileRow).Value = ObjFile.DateCreated
296         .Range("W" & FileRow).Value = ObjFile.DateLastModified
297         FileRow = FileRow + 1
298     Next ObjFile
299     .Range("H5" ).Value = Replace(.Range("B5" ).Value, "\", " > " ) 'FilePath
300     'Sort Columns Based On Defaults
301     Dim SortRng As Range
302     Set SortRng = .Range("Z17:Z22" ).Find(Sheet2.Range("CheckIcon" ).Value, , xlValues,
303     xlWhole)
304     If Not SortRng Is Nothing Then
305         For HeadCol = 19 To 23 'Set loop for each column header to remove any Ascending or
306         Descending
307             .Cells(6, HeadCol).Value = Replace(Replace(Replace(Replace(.Cells(6, HeadCol).
308             Value, " " & Sheet2.Range("SortAlphaAZ" ).Value, "" ), " " & Sheet2.Range(
309             "SortAlphaZA" ).Value, "" ), " " & Sheet2.Range("SortNum19" ).Value, "" ), " "
310             & Sheet2.Range("SortNum91" ).Value, "" ) 'Remove any Ascending or Descending
311             Alpha or Numerical Icons
312         Next HeadCol
313         .Cells(6, SortRng.Row + 2).Select
314     End If
315     ActiveWindow.ScrollRow = 9
316     .Range("B2" ).Value = False 'Set Load Files To False
317     .Range("A1" ).Select
318 End With
319 ResetCalc
320 End Sub

Sub SetAsDefaultFolder()
321     Sheet1.Range("B6" ).Value = Sheet1.Range("B5" ).Value 'Set Default Folder
322     FoldDefaultMess
323 End Sub

```


A

ActiveCell, 28, 30
 ActiveWindow, 28, 31, 33
 ActRow, 28-32
 After, 28-30
 Application, 29, 30, 32

C

Cells, 28-31, 33
 ClearContents, 28-30, 32
 CloseFolder, 28
 ColNum, 28-32
 Column, 28-30
 Copy, 28-30, 32
 Count, 30-33
 CreateFolder, 29
 CreateNewFolder, 29
 CreateObject, 28-30, 32
 CutCopyMode, 29, 30, 32

D

DateCreated, 28, 31, 33
 DateLastModified, 28, 31, 33
 Delete, 30, 32
 DeleteFolder, 29, 30
 DisplayDrives, 32
 DriveLoc, 32

E

Empty, 28, 32
 Err, 30
 ExpandFolder, 30
 Explicit, 28

F

FileRow, 28-33
 Files, 28, 31, 33
 Find, 28-31, 33
 FindParentRng, 29
 FoldDefaultMess, 33
 FoldPath, 29, 30
 FoldRng, 29
 FoldRow, 28-30, 32
 FoundLast, 28-30

G

GetFolder, 28, 30-32
 GoToHomeFolder, 31
 GoToNextFolder, 32
 GoToParentFolder, 30, 32
 GoToPrevFolder, 32

H

HeadCol, 30-33
 HighestLevel, 32

I

ItemNum, 28-30, 32

L

LastRow, 28-30, 32
 Lookat, 28-30
 LookIn, 28-30

M

MatchCase, 28-30
 MsgBox, 29, 30, 32

N

Name, 28, 31-33
 NextFold, 32, 33
 NextRow, 28-30, 32
 NoPath, 28-31
 Now, 29
 Number, 30

O

ObjFile, 28, 31, 33
 objFolder, 28-33
 objFSO, 28-32
 ObjSubFolder, 28, 30-33
 Offset, 28
 OpenFolder, 31, 32

P

ParentFolder, 29, 32
 PasteSpecial, 28-30, 32
 Path, 28, 31-33

R

Range, 28-33
 Replace, 28, 31, 33
 ResetCalc, 28, 29, 31, 33
 Resp, 29, 30
 Rng, 28-30
 Row, 28-31, 33
 RowNum, 28-33

S

ScrollRow, 28, 31, 33
 SearchDirection, 28-30
 SearchOrder, 28-30
 SendKeys, 29
 SetAsDefaultFolder, 33
 Shapes, 30, 32
 Sheet1, 28-33
 Sheet2, 28-33
 Size, 28, 31, 33
 SkipCopyPaste, 28
 SkipExpand, 30, 31
 SortRng, 31, 33
 StopCalc, 28, 30, 32
 SubfolderQty, 28-30, 32
 Subfolders, 30-33

U

UnMerge, 32

V

Value, 28-33
 vbNo, 30
 vbYesNo, 30

W

Wait, 29
 What, 28-30

X

xlByRows, 28-30
 xlDown, 28
 xlFormulas, 28-30
 xlPart, 28-30
 xlPasteFormats, 32
 xlPasteValues, 28-30
 xlPrevious, 28-30
 xlToLeft, 29
 xlUp, 29

xlValues, 29, 31, 33
 xlWhole, 29, 31, 33

```
1 Option Explicit
2
3 Sub Macro1()
4     '
5     ' Macro1 Macro
6     '
7     '
8     '
9     ActiveSheet.Shapes.Range(Array("PreviewFile" )).Select
10    Selection.verb Verb:=xlPrimary
11 End Sub
12 Sub Macro2()
13     '
14     ' Macro2 Macro
15     '
16     '
17     '
18     Range("X21" ).Select
19 End Sub
```

A

ActiveSheet, [35](#)

Array, [35](#)

E

Explicit, [35](#)

M

Macro1, [35](#)

Macro2, [35](#)

R

Range, [35](#)

S

Selection, [35](#)

Shapes, [35](#)

V

Verb, [35](#)

X

xlPrimary, [35](#)

```
1 Option Explicit
2
3 Sub MoveFiles()
4     Dim objFSO As Object
5     Dim SendToFldr As FileDialog
6     Dim MoveToFolder, FileName, FilePath As String
7     Dim LastRow, LastFiltRow, FiltRow, FilesMoved As Long
8     Set SendToFldr = Application.FileDialog(msoFileDialogFolderPicker)
9     Set objFSO = CreateObject("Scripting.FileSystemObject")
10    With SendToFldr
11        .Title = "Select a Folder To Move To"
12        If .Show <> -1 Then GoTo NoSel:
13        MoveToFolder = .SelectedItems(1)
14    End With
15    With Sheet1
16
17        'Run Advanced filter based on selected items
18        LastRow = .Range("S9999").End(xlUp).Row
19        .Range("A03,BA3:BH9999,CA3:CH9999").ClearContents 'Clear any previous Filters & Files
20        Moved Count
21        .Range("BA3").Value = Sheet2.Range("CheckIcon").Value 'Add icon for filtering
22        .Range("Q8:X" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range("BA2:BH3"),
23        CopyToRange:=.Range("CA2:CH2"), Unique:=True
24        LastFiltRow = .Range("CC999").End(xlUp).Row 'Last Filtered Row
25        For FiltRow = 3 To LastFiltRow
26            FileName = .Range("CC" & FiltRow).Value 'File Name
27            FilePath = .Range("CH" & FiltRow).Value 'Current File Path
28            'On Error GoTo FileExists
29            If Dir(MoveToFolder & "\" & FileName) <> "" Then 'File Exists
30                If vbYes = MsgBox(MoveToFolder & "\" & FileName & " already exists in the
31                destination folder. Would you like to replace this file?", vbYesNo, "File
32                Exists" ) Then objFSO.DeleteFile MoveToFolder & "\" & FileName Else: GoTo
33                DoNotReplace
34            End If
35            objFSO.MoveFile FilePath, MoveToFolder & "\" & FileName
36            FilesMoved = FilesMoved + 1
37            DoNotReplace:
38        Next FiltRow
39
40        StopCalc
41        ReloadFolder
42        ResetCalc
43        .Range("A03").Value = FilesMoved
44        FilesMovedMess
45        NoSel:
46    End With
47 End Sub
```

A

AdvancedFilter, [37](#)
Application, [37](#)

C

ClearContents, [37](#)
CopyToRange, [37](#)
CreateObject, [37](#)
CriteriaRange, [37](#)

D

DeleteFile, [37](#)
Dir, [37](#)
DoNotReplace, [37](#)

E

Explicit, [37](#)

F

FileDialog, [37](#)
FileName, [37](#)
FilePath, [37](#)
FilesMoved, [37](#)
FilesMovedMess, [37](#)
FiltRow, [37](#)

L

LastFiltRow, [37](#)
LastRow, [37](#)

M

MoveFile, [37](#)
MoveFiles, [37](#)
MoveToFolder, [37](#)
MsgBox, [37](#)
msoFileDialogFolderPicker, [37](#)

N

NoSel, [37](#)

O

objFSO, [37](#)

R

Range, [37](#)
ReloadFolder, [37](#)
ResetCalc, [37](#)
Row, [37](#)

S

SelectedItems, [37](#)
SendToFldr, [37](#)
Sheet1, [37](#)
Sheet2, [37](#)
Show, [37](#)
StopCalc, [37](#)

T

Title, [37](#)

U

Unique, [37](#)

V

Value, [37](#)
vbYes, [37](#)
vbYesNo, [37](#)

X

xlFilterCopy, [37](#)
xlUp, [37](#)

Option Explicit

```
1
2
3
4
5
6 Sub ShowHideColumns()
7   If ActiveCell.Value = Sheet2.Range("CheckIcon" ).Value Then 'Hide Column
8     ActiveCell.Value = Sheet2.Range("UncheckIcon" ).Value
9     Sheet1.Cells(1, ActiveCell.Row + 9).EntireColumn.Hidden = True
10  Else: 'Show Column
11    ActiveCell.Value = Sheet2.Range("CheckIcon" ).Value
12    Sheet1.Cells(1, ActiveCell.Row + 9).EntireColumn.Hidden = False
13  End If
14  Sheet1.Range("A1" ).Select
15 End Sub
16
17 Sub ShowHideOptions()
18   With Sheet1
19     If .Range("AA1" ).EntireColumn.Hidden = True Then
20       .Range("Z:AA" ).EntireColumn.Hidden = False
21       .Range("AP4" ).Value = "Hide Options"
22     Else:
23       .Range("Z:AA" ).EntireColumn.Hidden = True
24       .Range("AP4" ).Value = "Show Options"
25     End If
26   End With
27 End Sub
```

A

ActiveCell, [39](#)

C

Cells, [39](#)

E

EntireColumn, [39](#)

Explicit, [39](#)

H

Hidden, [39](#)

R

Range, [39](#)

Row, [39](#)

S

Sheet1, [39](#)

Sheet2, [39](#)

ShowHideColumns, [39](#)

ShowHideOptions, [39](#)

V

Value, [39](#)

Option Explicit

```

1 Sub FileSort()
2
3   Dim ActCol, LastRow, HeadCol As Long
4   Dim ActHeader, SortType As String 'Dimension String variables
5   With Sheet1
6     .Range("B2").Value = True 'Set File Load To True
7     StopCalc 'Run Macro to Stop Calculation & Screen Updating
8     ActHeader = ActiveCell.Value 'Determine Active Header Value for sorting
9     LastRow = .Range("S9999").End(xlUp).Row 'Get the last Row of the displayed table
10    .Sort.SortFields.Clear 'Clear any previous sorts on the table.
11    If InStr(ActHeader, Sheet2.Range("SortAlphaAZ").Value) <> 0 Or InStr(ActHeader, Sheet2
12      .Range("SortNum19").Value) <> 0 Then SortType = "Desc" 'Check if the active column
13      is set for Ascending and if so, then set Sort Type to Descending
14    If InStr(ActHeader, Sheet2.Range("SortAlphaZA").Value) <> 0 Or InStr(ActHeader, Sheet2
15      .Range("SortNum91").Value) <> 0 Then SortType = "Asc" 'Check if the active column is
16      set for Descending and if so, then set Sort Type to Ascending
17    For HeadCol = 19 To 23 'Set loop for each column header to remove any Ascending or
18      Descending
19      .Cells(6, HeadCol).Value = Replace(Replace(Replace(Replace(.Cells(6, HeadCol).Value,
20        " " & Sheet2.Range("SortAlphaAZ").Value, "" ), " " & Sheet2.Range("SortAlphaZA"
21        ).Value, "" ), " " & Sheet2.Range("SortNum19").Value, "" ), " " & Sheet2.Range(
22        "SortNum91").Value, "" ) 'Remove any Ascending or Descending Alpha or Numerical
23      Icons
24    Next HeadCol
25    'Run Sorting
26    If SortType = "Desc" Then 'If the sort type is Descending
27      .Sort.SortFields.Add Key:=.Cells(9, ActiveCell.Column), SortOn:=xlSortOnValues,
28      Order:=xlDescending, DataOption:=xlSortNormal 'Set Sort Parameters for Descending
29      Sort
30      If ActiveCell.Column < 21 Then ActiveCell.Value = ActiveCell.Value & " " & Sheet2.
31      Range("SortAlphaZA").Value Else: ActiveCell.Value = ActiveCell.Value & " " &
32      Sheet2.Range("SortNum91").Value 'Add in Descending Numerical or Alpha Character
33    Else: 'Sort type is Ascending
34      .Sort.SortFields.Add Key:=.Cells(9, ActiveCell.Column), SortOn:=xlSortOnValues,
35      Order:=xlAscending, DataOption:=xlSortNormal 'Set Sort Parameters for Ascending
36      Sort
37      If ActiveCell.Column < 21 Then ActiveCell.Value = ActiveCell.Value & " " & Sheet2.
38      Range("SortAlphaAZ").Value Else: ActiveCell.Value = ActiveCell.Value & " " &
39      Sheet2.Range("SortNum19").Value 'Add in Descending Numerical or Alpha Character
40    End If 'End of Setting Sort Parameters
41    With .Sort 'With Sort on Reports sheet
42      .SetRange Sheet1.Range("Q9:X" & LastRow) 'Set Sort Range
43      .Apply 'Run Sort
44    End With 'End if With Sort
45    .Range("B1").ClearContents 'Clear Selected Row on Sort
46    .Range("B2").Value = False 'Set File Load To False
47    .Range("A1").Select 'Select A1 in order to unselect Range
48  End With 'End With Reports Sheet
49  ResetCalc 'Run Macro to Restart Calculation & Screen Updating
50 End Sub

```

```

37 Sub SetDefaultSort()
38   Dim SortRow As Long
39   SortRow = ActiveCell.Row
40   Sheet1.Range("Z17:Z22").Value = Sheet2.Range("UncheckIcon").Value
41   Sheet1.Range("Z" & SortRow).Value = Sheet2.Range("CheckIcon").Value
42 End Sub

```


A

ActCol, [41](#)
ActHeader, [41](#)
ActiveCell, [41](#)
Add, [41](#)
Apply, [41](#)

C

Cells, [41](#)
Clear, [41](#)
ClearContents, [41](#)
Column, [41](#)

D

DataOption, [41](#)

E

Explicit, [41](#)

F

FileSort, [41](#)

H

HeadCol, [41](#)

I

InStr, [41](#)

K

Key, [41](#)

L

LastRow, [41](#)

O

Order, [41](#)

R

Range, [41](#)
Replace, [41](#)
ResetCalc, [41](#)
Row, [41](#)

S

SetDefaultSort, [41](#)
SetRange, [41](#)
Sheet1, [41](#)
Sheet2, [41](#)
Sort, [41](#)
SortFields, [41](#)
SortOn, [41](#)
SortRow, [41](#)
SortType, [41](#)
StopCalc, [41](#)

V

Value, [41](#)

X

xlAscending, [41](#)
xlDescending, [41](#)
xlSortNormal, [41](#)
xlSortOnValues, [41](#)
xlUp, [41](#)

```

1 Option Explicit
2 Dim SelCol, PrevTab, NewTab, CurTab As Long
3
4
5 Sub CloseTab()
6     With Sheet1
7         CurTab = .Range("AN5").Value
8         SelCol = CurTab * 4
9         If CurTab = 1 Then
10            .Shapes("CloseTabBtn").Visible = msoFalse
11            Exit Sub
12        End If
13        .Cells(4, SelCol).Value = Sheet2.Range("AddTabIcon").Value
14        .Range("RangeTab" & CurTab).ClearContents 'Clear files & Folder Data
15        .Range("AV" & CurTab + 1).ClearContents 'Clear Folder
16        .Range("AN5").Value = 1 'Set Tab Back to 1
17        .Range("D4").Select
18    End With
19 End Sub
20
21 Sub OpenNewTab()
22     With Sheet1
23         StopCalc
24         .Range("B2").Value = True
25         SelCol = ActiveCell.Column
26         PrevTab = .Range("AN5").Value
27         NewTab = Round(ActiveCell.Column / 4, 0)
28         .Range("AN5").Value = NewTab 'Set New Tab Number
29         'Preserve Existing Tab Data
30         .Range("RangeMain").Copy
31         .Range("RangeTab" & PrevTab).PasteSpecial xlPasteValues
32         .Range("RangeTab" & PrevTab).PasteSpecial xlPasteFormulas
33         .Range("AV" & PrevTab + 1).Value = .Range("B5").Value 'Preserve Existing folder in
34         correct location
35         .Cells(4, PrevTab * 4).Value = .Cells(4, PrevTab * 4).Value 'Set Fixed Folder For
36         Previous Tab (From Dynamic to Fixed)
37
38         .Cells(4, SelCol).Formula = .Range("TabFormula").Formula 'Set Dynamic Folder Name
39
40         .Range("D9:O1000,Q9:Q1000,S9:X1000").ClearContents 'Clear Existing Folders & Files
41         .Range("B5").Value = .Range("B6").Value 'Set initial folder based on default folder
42         If SelCol > 4 Then
43             With .Shapes("CloseTabBtn")
44                 .Visible = msoCTrue
45                 .Left = Sheet1.Cells(4, SelCol).Left
46                 .IncrementLeft 52
47             End With
48         Else:
49             .Shapes("CloseTabBtn").Visible = msoFalse
50         End If
51         .Range("B2").Value = False
52         OpenFolder
53         ResetCalc
54     End With
55 End Sub
56
57 Sub SwitchTab()
58     With Sheet1
59         StopCalc
60         .Range("B2").Value = True

```

```
1 2
60 SelCol = ActiveCell.Column
61 PrevTab = .Range("AN5" ).Value
62 NewTab = Round(ActiveCell.Column / 4, 0)
63 .Range("AN5" ).Value = NewTab 'Set New Tab
64 'Preserve Existing Tab Data
65 .Range("RangeMain" ).Copy
66 .Range("RangeTab" & PrevTab).PasteSpecial xlPasteValues
67 .Range("RangeTab" & PrevTab).PasteSpecial xlPasteFormulas
68 .Range("AV" & PrevTab + 1).Value = .Range("B5" ).Value 'Preserve Existing folder in
correct location
69 .Cells(4, PrevTab * 4).Value = .Cells(4, PrevTab * 4).Value 'Set Fixed Folder For
Previous Tab
70
71 'Load Selected Tab Data
72 .Range("RangeTab" & NewTab).Copy
73 .Range("RangeMain" ).PasteSpecial xlPasteValues
74 .Range("RangeMain" ).PasteSpecial xlPasteFormulas
75 .Range("B5" ).Value = .Range("AV" & NewTab + 1).Value 'Set Folder For Tab
76 .Cells(4, SelCol).Formula = .Range("TabFormula" ).Formula 'Set Dynamic Folder Name
77 If SelCol > 4 Then
78     With .Shapes("CloseTabBtn" )
79         .Visible = msoCTrue
80         .Left = Sheet1.Cells(4, SelCol).Left
81         .IncrementLeft 52
82     End With
83 Else:
84     .Shapes("CloseTabBtn" ).Visible = msoFalse
85 End If
86
87 .Range("B2" ).Value = False
88 .Range("S7" ).Select
89 ResetCalc
90 End With
91 End Sub
```

A

ActiveCell, [43](#), [44](#)

C

Cells, [43](#), [44](#)
ClearContents, [43](#)
CloseTab, [43](#)
Column, [43](#), [44](#)
Copy, [43](#), [44](#)
CurTab, [43](#)

E

Explicit, [43](#)

F

Formula, [43](#), [44](#)

I

IncrementLeft, [43](#), [44](#)

L

Left, [43](#), [44](#)

M

msoCTrue, [43](#), [44](#)
msoFalse, [43](#), [44](#)

N

NewTab, [43](#), [44](#)

O

OpenFolder, [43](#)
OpenNewTab, [43](#)

P

PasteSpecial, [43](#), [44](#)
PrevTab, [43](#), [44](#)

R

Range, [43](#), [44](#)
ResetCalc, [43](#), [44](#)
Round, [43](#), [44](#)

S

SelCol, [43](#), [44](#)
Shapes, [43](#), [44](#)
Sheet1, [43](#), [44](#)
Sheet2, [43](#)
StopCalc, [43](#)
SwitchTab, [43](#)

V

Value, [43](#), [44](#)
Visible, [43](#), [44](#)

X

xlPasteFormulas, [43](#), [44](#)
xlPasteValues, [43](#), [44](#)

Thank You!

This source code was created and made available to help you gain a better understanding of how VBA is used to create amazing Excel-based applications.

Thank you so much for your continued shares, likes and support. It really helps.



Excel For Freelancers