**Tổng quan về việc sử dụng FileSystemObject**

1. **FileSystemObject là gì**?  
   FSO là 1 công cụ mạnh chuyên dùng để xử lý các vấn đề liên quan đến Drive, Folders và Files. Bài viết này chủ yếu tập trung vào Folders và Files.
2. **Các ứng dụng cơ bản và code điển hình**  
   *2.1 -Khởi tạo kích hoạt công cụ FSO:*  
   Khi muốn sử dụng công cụ FSO thì ta phải khởi tạo kích hoạt trước khi gọi các lệnh liên quan đến FSO. Sau đây là cách khởi tạo phổ thông nhất mà ta thường gặp

Dim fso As Object

Set fso = CreateObject("Scripting.FileSystemObject")

Sau dòng lệnh trên thì ta đã có thể gọi các lệnh liên quan đến công cụ này để xử lý những yêu câu tiếp theo.  
*2.2 -Các phương thức phổ biến*  
a. CreateFolder (dùng để tạo 1 thư mục mới)  
Cơ bản của lệnh này như sau:

Dim fso As Object, NewFolder As String

Set fso = CreateObject("Scripting.FileSystemObject")

NewFolder = "D:\Sample"

fso.CreateFolder (NewFolder)

Sau khi thực thi các câu lệnh trên thì 1 thư mục có tên là Sample đã được tạo ra trong ổ đĩa D.  
Điều đáng chú ý là công cụ này có thể xử lý tốt tiếng Việt có dấu.  
Nếu thư mục này đã có sẵn rồi thì sẽ bị lỗi code. Để khắc phục điều này chúng ta sẽ cùng tìm hiểu trong phần sau.  
b. DeleteFolder (dùng để xóa 1 thư mục có sẵn)

Dim fso As Object, FolderToDelete As String

Set fso = CreateObject("Scripting.FileSystemObject")

FolderToDelete= "D:\Sample"

fso.DeleteFolder(FolderToDelete)

Sau khi thực thi các câu lệnh trên thì thư mục có tên là Sample trong ổ đĩa D sẽ bi xóa.  
***Ở đây ta bỏ qua tham số True và False của phương thức này. Mặc định luôn là True.***  
Chú ý là nếu có file(s) nào đang mở thì thư mục này không bị xóa, tuy nhiên các file trong thư mục này sẽ bị xóa hết trừ file(s) đang mở. Lúc này sẽ xuất hiện 1 thông báo lỗi không thể truy cập.  
Hoặc ta cung cấp tên 1 thư mục không tồn tại thì cũng sẽ gây ra lỗi. Vấn đề này sẽ được xử lý bằng 1 phương thức tiếp theo.  
c. FolderExists (dùng để kiểm tra xem thư mục có tồn tại hay không)  
Khi tạo 1 thư mục vào 1 đường dẫn đã có tồn tại 1 thư mục trùng tên thì sẽ gây ra lỗi. Để khắc phục vấn đề này ta sử dụng phương thức FolderExists

Dim fso As Object, NewFolder As String

Set fso = CreateObject("Scripting.FileSystemObject")

NewFolder = "D:\Sample"

If Not fso.FolderExists(NewFolder) then

fso.CreateFolder (NewFolder)

End If

Như vậy lỗi tạo thư mục trùng tên đã bị loại bỏ. Tương tự, ta có thể kiểm tra trước khi thực thi câu lệnh DeleteFolder

Dim fso As Object, FolderToDelete As String

Set fso = CreateObject("Scripting.FileSystemObject")

FolderToDelete= "D:\Sample"

If fso.FolderExists(NewFolder) then

fso.DeleteFolder(FolderToDelete)

End If

\*\*\* Ngoài ra ta có 1 cách tạo thư mục khác cũng dùng FSO như sau:

Dim fso As Object, NewFolder As String

Set fso = CreateObject("Scripting.FileSystemObject")

NewFolder= "Sample"

fso.GetFolder("D:").SubFolders.Add "NewFolder"

d. CopyFolder (dùng để copy 1 folder từ nơi này đến nơi khác)  
Trên thực tế ít khi chúng ta sử dụng phương thức này

Dim fso As Object, FolderToCopy As String, DesFolder As String

Set fso = CreateObject("Scripting.FileSystemObject")

FolderToCopy= "D:\Sample"

DesFolder= "E:\Sample"

fso.CopyFolder FolderToCopy, DesFolder

e. GetParentFolderName (dùng để lấy tên thư mục mẹ của 1 Folder hoặc của 1 file)  
Phương thức GetParentFolderName rất thuận tiện khi ta muốn lấy tên thư mục mẹ của 1 file hay 1 thư mục nào đó. Câu lệnh dưới đây sẽ trả về cho ta chuỗi **D:\Sample**, với Item1 là thư mục.

Dim fso As Object, CurrentFolder As String

Set fso = CreateObject("Scripting.FileSystemObject")

CurrentFolder= "D:\Sample\Item1"

Msgbox fso.GetParentFolderName(CurrentFolder)

Nếu chuỗi đường dẫn là 1 file thì sẽ trả về cho ta thư mục chứa file đó  
Code sau đây sẽ trả về cho ta chuỗi **D:\Sample\Item1**

Dim fso As Object, CurrentFile As String

Set fso = CreateObject("Scripting.FileSystemObject")

CurrentFile= "D:\Sample\Item1\Baitap.xls"

Msgbox fso.GetParentFolderName(CurrentFile)

f. GetAbsolutePathName (dùng để lấy tên đường dẫn đầy đủ của 1 file hoặc 1 thư mục)  
Code này sẽ trả về cho ta chuỗi **D:\Sample\Item1\Baitap.xls**

Dim fso As Object, CurrentFile As String

Set fso = CreateObject("Scripting.FileSystemObject")

CurrentFile= "D:\Sample\Item1\Baitap.xls"

Msgbox fso.GetAbsolutePathName(CurrentFile)

g. GetBaseName (dùng để lấy tên của 1 file không kèm theo phần mở rộng)  
Code này sẽ trả về cho ta chuỗi **Baitap**

Dim fso As Object, CurrentFile As String

Set fso = CreateObject("Scripting.FileSystemObject")

CurrentFile= "D:\Sample\Item1\Baitap.xls"

Msgbox fso.GetBaseName(CurrentFile)

h. GetExtensionName (dùng để lấy phần mở rộng của 1 file)  
Code này sẽ trả về cho ta chuỗi **xls**

Dim fso As Object, CurrentFile As String

Set fso = CreateObject("Scripting.FileSystemObject")

CurrentFile= "D:\Sample\Item1\Baitap.xls"

Msgbox fso.GetExtensionName (CurrentFile)

CopyFile (dùng để copy file(s) từ 1 thư mục đến thư mục khác)  
Cái thú vị của lệnh này là nếu ta chỉ ra đường dẫn và tên file đích, thì code sẽ lấy tên file mà ta đã chỉ định.  
Thông thường ta sẽ thực hiện code thế này để copy file

Dim fso As Object, FileToCopy As String, Des As String

Set fso = CreateObject("Scripting.FileSystemObject")

FileToCopy= "D:\Sample\Item1\Baitap.xls"

Des="D:\Sample\Item2\Baitap.xls"

fso.CopyFile FileToCopy, Des

Tuy nhiên nếu ta viết thế này thì code vẫn hiểu. Nếu ta không chỉ định tên file thì code sẽ lấy tên file gốc.  
Nếu ta muốn đổi tên file thì code sẽ cho ta tên file mới với cùng nội dung trong file.

Dim fso As Object, FileToCopy As String, Des As String

Set fso = CreateObject("Scripting.FileSystemObject")

FileToCopy= "D:\Sample\Item1\Baitap.xls"

Des="*D*:\Sample\Item2\"

fso.CopyFile FileToCopy, Des

j. DeleteFile (dùng để xoá 1 file có thuộc tính ẩn hoặc không ẩn)  
Code sau đây sẽ xoá 1 tập tin tên **Baitap.xls** trong thư mục Item1  
*Lưu ý là ở đây chúng ta bỏ qua tham số thứ 2 của phương thức này.*  
Nếu file chỉ định để xoá không tồn tại thì sẽ gây ra lỗi code

Dim fso As Object, FileToDelete As String

Set fso = CreateObject("Scripting.FileSystemObject")

FileToDelete= "D:\Sample\Item1\Baitap.xls"

fso.DeleteFile(FileToDelete)

k. FileExists (dùng để kiểm tra sự tồn tại của 1 tập tin)  
Để khắc phục lỗi có thể xảy ra khi ta yêu cầu xoá 1 tập tin không tồn tại thì phương thức FileExists sẽ giúp chúng ta thực hiện điều này.

Dim fso As Object, FileToDelete As String

Set fso = CreateObject("Scripting.FileSystemObject")

FileToDelete= "D:\Sample\Item1\Baitap.xls"

If fso.FileExists(FileToDelete) Then

fso.DeleteFile(FileToDelete)

End If

l. CreateTextFile (dùng để xuất dữ liệu thành 1 file dạng txt)  
Với phương thức này ta nên lưu ý đến các tham số  
fso.CreateTextFile(filename, overwrite, unicode)  
Code sau đây sẽ xuất nội dung từ A1 đến A10 vào file txt

Dim fso As Object, MyFile As Object

Dim FileName As String, *I* As Long

Set fso = CreateObject("Scripting.FileSystemObject")

FileName = "D:\Test.txt"

Set MyFile = fso.CreateTextFile(FileName, *True*, *True*)

With MyFile

For *I* = 1 To 10

.WriteLine cells(i,1)

Next

.Close

End With

Nếu ta muốn xuất nhiều cột thì phải nối các cột lại với nhau bằng 1 dấu phân cách để khi cần truy xuất.  
Code sau đây dùng dấu tab để nối dữ liệu của cột A và B.  
Lưu ý là nếu có quá nhiều cột thì ta nên dùng vòng lặp lồng để nối các cột trước khi ghi vào file

Dim fso As Object, MyFile As Object

Dim FileName As String, *I* As Long

Set fso = CreateObject("Scripting.FileSystemObject")

FileName = "D:\Test.txt"

Set MyFile = fso.CreateTextFile(FileName, *True*, *True*)

With MyFile

For *I* = 1 To 10

.WriteLine cells(i,1) & vbTab & cells(i,2)

Next

.Close

End With

m. OpenTextFile (dùng để mở 1 file txt)  
  
Đối với phương thức này chúng ta cần lưu ý đến các tham số.  
fspenTextFile(filename, mode, create, format)  
Tham số 1: tên đường dẫn file  
Tham số 2: để đọc file (1), để ghi đè (2), để cập nhật tiếp (8).  
Tham số 3: tạo file mới (True), bỏ trống mặc định là False.  
Tham số 4: as unicode (-1), as ASCII (1), system default (-2).  
Code sau đây sẽ mở file Test.txt có sẵn để ghi tiếp dữ liệu vào dòng cuối.  
Nếu file này không tồn tại sẽ gây ra lỗi. Ta có thể dùng phương thức FileExists để bẫy lỗi này.

Dim fso As Object, MyFile As Object

Dim FileName As String, *I* As Long

Set fso = CreateObject("Scripting.FileSystemObject")

FileName = "D:\Test.txt"

Set MyFile = fso.OpenTextFile(FileName, 8,*False*, -1)

With MyFile

For *I* = 1 To 10

.WriteLine cells(i,1) & vbTab & cells(i,2)

Next

.Close

End With

Khi ta muốn Import dữ liệu từ 1 file txt vào Excel thì cũng dùng phương thức OpenTextFile.  
Phần này được dùng tương đối nhiều và cũng là 1 trong những cái phần quan trọng và khó.  
Trước tiên ta phải biết được dấu phân cách dữ liệu của file txt. Giả định trong bài này là dấu vbTab.  
Khi ta dùng phương thức để đọc dữ liệu từ file txt thì fso sẽ cho ta 1 chuỗi dữ liệu liên tục được nối với nhau bằng những dấu xuống dòng (vbCrLf).  
Sau khi đọc từ file ta có 1 chuỗi:  
"a b c vbCrLf d e f vbCrLf h i j vbCrLf k l m....................................."  
Bước tiếp theo là ta dùng hàm Split của VBA để tách cái chuỗi dài vô tận này ra, dựa trên các dấu (vbCrLf).  
Sau khi tách chuỗi trên ra ta sẽ có như sau:  
"a b c"  
"d e f"  
"h i j"  
"k l m"  
Tiếp theo ta sẽ dùng hàm Split để tách tiếp những phần tử được ngăn cách bởi dấu vbTab  
Kết quả sau khi tách như sau:  
"a" "b" "c"  
"d" "e" " f"  
................  
Đến đây ta sẽ gán các giá trị này vào vùng dữ liệu đích.  
  
Chúng ta cùng tham khảo 1 code cơ bản để import dữ liệu từ 1 file txt  
Vì là code cơ bản nên ta chưa loại bỏ hết những phát sinh có thể xảy ra.

Sub ImportTextToExcel()

Dim fso As Object, TextSource As Object, TotalLines, TextItem

Dim ItemsOfLine As String, Delimiter As String, FilesToOpen As String, Res()

Dim *K* As Long, Cols As Integer, LineNum As Long

Set fso = CreateObject("Scripting.FileSystemObject")

Delimiter = vbTab

FilesToOpen = "D:\Sample\Test.txt"

Set TextSource = fso.OpenTextFile(FilesToOpen, 1, *False*, -2)

TotalLines = Split(TextSource.ReadAll, vbCrLf)

ReDim Res(1 To 1 + UBound(TotalLines), 1 To 1)

For LineNum = LBound(TotalLines) To UBound(TotalLines)

ItemsOfLine = TotalLines(LineNum)

TextItem = Split(ItemsOfLine, Delimiter)

If UBound(Res, 2) < UBound(TextItem) + 1 Then

ReDim Preserve Res(1 To 1 + UBound(TotalLines), 1 To UBound(TextItem) + 1)

End If

*K* = *K* + 1

For Cols = LBound(TextItem) To UBound(TextItem)

Res(*K*, Cols + 1) = TextItem(Cols)

Next

Next

[*A1*].Resize(*K*, UBound(Res, 2)) = Res

End Sub

Trong bài số 1 chúng ta đã làm quen với những phương thức cơ bản của FileSytemObject  
Bài này chúng ta sẽ cũng tham khảo các ứng dụng thực tế và các cách liên kết các phương thức với nhau  
Dưới đây là code dùng duyệt qua tất cả các file trong 1 thư mục. Tìm xem file nào có đuôi là **tmp**thì xóa bỏ.  
Code này sẽ xóa cả file mang thuộc tính ẩn hoặc file hệ thống.

Sub DeleteTmpFile()

Dim fso As Object, ObjFile As Object

Set fso = CreateObject("Scripting.FileSystemObject")

With fso.GetFolder(ThisWorkbook.Path)

For Each ObjFile In .Files

If fso.GetExtensionName(ObjFile) = "tmp" Then

fso.DeleteFile (ObjFile)

End If

Next

End With

End Sub

Cũng với code trên ta có thể tạo ra 2 Sub riêng biệt. Khi cần chỉ thay đổi đường dẫn tại Sub Main

Sub DeleteTmpFile(ByVal StrFolder As String)

Dim fso As Object, ObjFile As Object

Set fso = CreateObject("Scripting.FileSystemObject")

With fso.GetFolder(StrFolder)

For Each ObjFile In .Files

If fso.GetExtensionName(ObjFile) = "tmp" Then

fso.DeleteFile (ObjFile)

End If

Next

End With

End Sub

Sub Main()

Dim path As String

path = ThisWorkbook.path

DeleteTmpFile (path)

End Sub

Để cho linh động hơn trong việc chọn thư mục, ta sẽ bổ sung code thế này

Sub DeleteTmpFile()

Dim fso As Object, ObjFile As Object

Dim path As String, chk As Boolean

chk = Application.FileDialog(msoFileDialogFolderPicker).Show

If Not chk Then Exit Sub

path = Application.FileDialog(msoFileDialogFolderPicker).SelectedItems(1)

Set fso = CreateObject("Scripting.FileSystemObject")

With fso.GetFolder(path)

For Each ObjFile In .Files

If fso.GetExtensionName(ObjFile) = "tmp" Then

fso.DeleteFile (ObjFile)

End If

Next

End With

End Sub

Chỉ cần điều chỉnh lại 1 chút thì từ code trên ta sẽ tạo ra 1 code để liệt kê tất cả file excel trong 1 thư mục chỉ định  
Khi đã có được 1 mảng chứa tên file rồi thì ta có thể xử lý theo yêu cầu cụ thể cho từng trường hợp.

Sub GetFileList()

Dim fso As Object, ObjFile As Object

Dim path As String, chk As Boolean, Res(), *I* As Long

chk = Application.FileDialog(msoFileDialogFolderPicker).Show

If Not chk Then Exit Sub

path = Application.FileDialog(msoFileDialogFolderPicker).SelectedItems(1)

Set fso = CreateObject("Scripting.FileSystemObject")

With fso.GetFolder(path)

For Each ObjFile In .Files

If fso.GetExtensionName(ObjFile) Like "xls\*" Then

*I*=*I*+1

ReDim Preserve Res(1 To *I*)

Res(*I*)=ObjFile.Name

End If

Next

End With

End Sub

Hoặc ta có thể tách ra thành 1 UDF và 1 Sub Main thế này

Sub Main()

Dim path As String, chk As Boolean, Sarr()

With Application.FileDialog(msoFileDialogFolderPicker)

chk = .Show

If Not chk Then Exit Sub

path = .SelectedItems(1)

Sarr = GetFileList(path)

End With

End Sub

Function GetFileList(ByVal StrFolder As String)

Dim fso As Object, ObjFile As Object

Dim Res(), *K* As Long

Set fso = CreateObject("Scripting.FileSystemObject")

With fso.GetFolder(StrFolder)

For Each ObjFile In .Files

If fso.GetExtensionName(ObjFile) Like "xls\*" Then

*K* = *K* + 1

ReDim Preserve Res(1 To *K*)

Res(*K*) = ObjFile.Name

End If

Next

End With

GetFileList = Res

End Function

Đây là 1 code dùng để import dữ liệu từ 1 hay nhiều file txt vào file excel.  
Với code này bạn chỉ cần kiểm tra xem file txt sử dụng dấu phân cách gì và sửa lại chỗ Delimiter= vbTab  
Nếu dấu phân cách là dấu phẩy (,) thì sửa lại thế này Delimiter= ","  
Lưu ý là code này khai báo chỉ số lớn nhất của mảng chỉ có 65536, nếu dữ liệu có nhiều hơn thì sẽ gây ra lỗi. Trong trường hợp đó bạn có thể sửa số 65536 thành 1 số lớn hơn. (Chắc hiếm gặp)  
Code này sẽ xử lý tất cả file text xong rồi mới gán dữ liệu 1 lần xuống sheet tại ô A2. Bạn có thể sửa lại cho phù hợp.

Sub ImportTextToExcel()

Dim fso As Object, FilesToOpen, TextSource As Object, TotalLines, Res()

Dim ItemsOfLine, TextItem, Delimiter As String, n As Byte

Dim *K* As Long, *X* As Byte, Cols As Integer, LineNum As Long

Set fso = CreateObject("Scripting.FileSystemObject")

Delimiter = vbTab

FilesToOpen = Application.GetOpenFilename("Text Files (\*.txt), \*.txt", , , , *True*)

If Not IsArray(FilesToOpen) Then Exit Sub

For *X* = LBound(FilesToOpen) To UBound(FilesToOpen)

Set TextSource = fso.OpenTextFile(FilesToOpen(*X*), 1, , -2)

TotalLines = Split(TextSource.ReadAll, vbCrLf)

For LineNum = LBound(TotalLines) To UBound(TotalLines)

ItemsOfLine = TotalLines(LineNum)

TextItem = Split(ItemsOfLine, Delimiter)

If UBound(TextItem) + 1 > n Then

ReDim Preserve Res(1 To 65536, 1 To UBound(TextItem) + 1)

n = UBound(TextItem) + 1

End If

If ItemsOfLine <> String(Len(ItemsOfLine), vbTab) Then

*K* = *K* + 1

For Cols = LBound(TextItem) To UBound(TextItem)

Res(*K*, Cols + 1) = TextItem(Cols)

Next

End If

Next

Next

[*A2*].Resize(*K*, UBound(Res, 2)) = Res

End Sub

Và đây là 1 mẫu code để lấy tất cả file trong 1 thư mục mẹ và file trong thư mục con nếu có.  
Code này sưu tầm của thành viên Siwtom

Sub Main()

Dim fso As Object, res() As String

Set fso = CreateObject("Scripting.FileSystemObject")

With Application.FileDialog(msoFileDialogFolderPicker)

If .Show Then

ReDim res(1 To 1)

GetAllFiles .SelectedItems(1), fso, res

If UBound(res) > 1 Then [*A5*].Resize(UBound(res) - 1) = Application.Transpose(res)

End If

End With

End Sub

Function GetAllFiles(ByVal StrFolder As String, fso As Object, res() As String)

Dim objFolder As Object, objSubFolder As Object, File

Set objFolder = fso.GetFolder(StrFolder)

For Each File In objFolder.Files

res(UBound(res)) = fso.GetBaseName(File)

ReDim Preserve res(1 To UBound(res) + 1)

Next

For Each objSubFolder In objFolder.SubFolders

GetAllFiles objSubFolder.Path, fso, res

Next objSubFolder

End Function

Đừng vội, sau những phần cơ bản sẽ là những phần phức tạp. Cơ bản thì lúc nào cũng chỉ có bấy nhiêu thôi. Nhưng ráp chúng lại với nhau mới là vấn đề.  
Nếu nóng vội thì cứ lấy code này về mà thức suy nghĩ nhé  
Code này sẽ lấy hết tên file và thư mục chứa trong thư mục chỉ định. Chạy Sub Main nhé

Public k As Long, Res(1 To 10000, 1 To 2)

Sub Main()

Dim strPath As String, fso As Object, chk As Boolean

Set fso = CreateObject("Scripting.FileSystemObject")

chk = Application.FileDialog(msoFileDialogFolderPicker).Show

If Not chk Then Exit Sub

strPath = Application.FileDialog(msoFileDialogFolderPicker).SelectedItems(1)

If Len(strPath) Then Call GetAllFolders(strPath, fso)

If k Then [*A5*].Resize(k, 2) = Res

k = 0

End Sub

Sub GetAllFolders(ByVal strFolder As String, ByRef fso As Object)

Dim objFolder As Object, objSubFolder As Object, File

Set objFolder = fso.GetFolder(strFolder)

For Each objSubFolder In objFolder.subfolders

Call GetAllFolders(objSubFolder.Path, fso)

For Each File In objSubFolder.Files

k = k + 1

Res(k, 1) = fso.getbasename(File)

Res(k, 2) = fso.GetParentFolderName(File)

Next

Next objSubFolder

End Sub