

VISUAL BASIC – HOW TO CREATE A FILE

This page concentrates on creating a text file with a **VBScript**. I expect that you can think of many scripting scenarios where it would be advantageous to output the data, not to the screen, but to a file.

Software inventory, registry values and event logs are just some of the examples that spring to mind.

Remember, that scripting starts at the folder level, therefore, even if you are primarily interested in saving to a file, your script must first get or create the parent folder.

Our Mission and Goal

Our mission on this page is clear, to create a file. Naturally, we need a folder to hold our file, therefore, our script will test to see if the folder exists, and if not, then our code will create a folder.

Example 1 - Basic VBScript to Create a File with CreateTextFile

The idea behind this script is to highlight the code need to create a file. The cost of focussing on one method is that **Example 1** works fine the first time you execute the script, but if you run it a second time you get an error. To cure the problem, move on to the second script.

Else keep altering the *strDirectory*. Here is what I did as a work-around:

- 1st Run *strDirectory* = "E: \logs"
- 2nd Run *strDirectory* = "E: \logs\guy1"
- 3rd Run *strDirectory* = "E: \ logs\guy2"

I admit this is poor scripting, but my dilemma is keeping it simple and highlighting the learning steps, versus a slick script that is difficult to follow.

To tell the whole truth, I fervently believe that you learn more when scripts go wrong - providing you can fix the error quickly.

Prerequisites

This is a script that will create a file equally well on a **Windows Server** or an XP machine. Should you get permission errors, I recommend that you logon as administrator.

Instructions for Creating Files

1. Copy and paste the example script below into notepad or a **VBScript** editor.
2. Decide whether to change the values for *strFile* and *strDirectory*.
3. Save the file with a **.vbs** extension, for example: **NewFile.vbs**.
4. Double click **NewFile.vbs** and check Windows Explorer for *strDirectory*.

Sample VBScript to Create a File

```
' NewFile.vbs  
' Sample VBScript to create a file using FileSystemObject  
' Author Guy Thomas http://computerperformance.co.uk/  
' Version 1.6 - August 2010
```

```
' -----'
Option Explicit
Dim objFSO, objFSOText, objFolder, objFile
Dim strDirectory, strFile
strDirectory = "E:\logs\guy1"
strFile = "\Summer.txt"
' Create the File System Object
Set objFSO = CreateObject("Scripting.FileSystemObject")
' Create the Folder specified by strDirectory on line 10
Set objFolder = objFSO.CreateFolder(strDirectory)
' -- The heart of the create file script
'-----
'Creates the file using the value of strFile on Line 11
' -----
Set objFile = objFSO.CreateTextFile(strDirectory & strFile)
Wscript.Echo "Just created " & strDirectory & strFile
Wscript.Quit

' End of FileSystemObject example: NewFile VBScript
```

VBScript Tutorial - Learning Points

Note 1: All FSO scripts begin by creating a **File System Object** with **CreateObject("Scripting.FileSystemObject")**. You really do need the word "**Scripting**".

Note 2: The specific method for creating the file is called: **CreateTextFile**.

Note 3: Observe that we also employed the sister method **CreateFolder**. In fact, this causes a problem, if you run the script for a second time because it generates the file exists error **800A003A**. So, in **Example 2** we are going to introduce error-correcting code.

Example 2 - Create a File with error-correcting Code

This script is much better than the primitive **Example 1**. My idea is to add code, which copes with situations where the folder already exists. It also employs the **objShell** to run the Windows Explorer so that you can test that the script works as designed.

Sample Script to Create a File and Check if a File Already Exists

```
' NewFileEC.vbs
' Sample VBScript to create a file with error-correcting Code
' Author Guy Thomas http://computerperformance.co.uk/
' VBScript Create File
' -----'
Option Explicit
Dim objFSO, objFolder, objShell, objFile

Dim strDirectory, strFile
strDirectory = "e:\logs"
strFile = "\Summer.txt"
' Create the File System Object
Set objFSO = CreateObject("Scripting.FileSystemObject")
' Check that the strDirectory folder exists
If objFSO.FolderExists(strDirectory) Then
Set objFolder = objFSO.GetFolder(strDirectory)
Else
Set objFolder = objFSO.CreateFolder(strDirectory)
WScript.Echo "Just created " & strDirectory
```

```
End If
If objFSO.FileExists(strDirectory & strFile) Then
Set objFolder = objFSO.GetFolder(strDirectory)
Else
Set objFile = objFSO.CreateTextFile(strDirectory & strFile)
Wscript.Echo "Just created " & strDirectory & strFile
End If
set objFolder = nothing
set objFile = nothing
If err.number = vbEmpty then
Set objShell = CreateObject("WScript.Shell")
objShell.run ("Explorer" & " " & strDirectory & "\" )
Else WScript.echo "VBScript Error: " & err.number
End If
WScript.Quit
' End of VBScript to create a file with error-correcting Code
```

VBScript Tutorial - Learning Points

Note 1: If you do not add error-correcting code, then there is a real danger that running the script again will destroy the original file. What **.FileExists** does is to check to see if there is already a file with the name as that referenced by **strFile**.

Note 2: This script also features code which protects against the folder already existing - **FolderExists**.

Note 3: A tiny point, but the script produces an error when its first run, unless I added set **objFolder** and **objFile = nothing**. Here is an example where I always learn more when things go wrong. I have known for a long time that I should nullify objects when I have finished them, but idleness and wanting to keep the script short have meant I have omitted these set objxyz = nothing from of other scripts. Sometimes I get away with it, but not in this example.

Note 4: **objShell = CreateObject("WScript.Shell")** is my way of showing that the script has achieved its goal. The active part is **objShell.run**. In this example, it is as if you click on the Start (Menu), Run and type **Explorer**.

Summary of VBScript Create Files

With file system objects there is a natural progression, create the folder, then the file, then read or write text to that file.

This page shows you to creating a file using the **CreateTextFile** method. Next step let us explore reading or writing data to the file.