Vbscript ip file lab essay sample

Objective

In this lab, students will complete the following objectives. * Create a VBScript program using NotePad++.

* Write a two-dimensional array of IP addresses to a text file. * Read the IP Addresses text file into a script.

* Append new Room/PC/IP address data to the text file.

* Use the object Scripting. FileSystemObject.

Element K Network Connections

For this lab, we will only need to connect to vlab-PC1. The computer vlab-PC1 is the computer on the left side while vlab-PC2 is on the right. If you leave the cursor on the PC icon for a few seconds, a tool-tip message will appear indicating the hostname of the PC. Open vlab-PC1 and log in as Administrator with the password password.

Lab Overview

We are going to start our lab with the same two-dimensional array of IP addresses. Rather than accessing this array of IP address using the room and computer index values, we are going to write the array of IP addresses to a Text file named IP_Addresses. csv where each line of the file will contain the Comma Separated Values (CSV) for room#, computer#, IP_Address. We will then write a separate VBScript program that will append four new lines of data that will represent the four computers in the new room 106 to the IP_Addresses. csv file. Lastly, we will open and read the newly appended IP_Addresses. csv and display it contents in a meaningful way.

Note: All captures must be text only—DO NOT capture the NotePad++ application window or the command prompt window. Use copy and paste of text only.

Task 1: Download and Open IP_ArrayFile_start. vbs in NotePad++

* Open NotePad++ and from the menu, select File/Open. Open the file IP_File_start. vbs in the C: Scripts directory. If you do not see this file, you can download it and extract it from the eCollege Doc Sharing file IP_File_start. zip.

* Modify the Programmer Header as needed and Save As the VBScript file as IP_FileWrite. vbs.

* The line dim ipAddress(5, 3) declare 6×4 two-dimensional array. The 5 and 3 give the maximum index value. Since indices always start at 0, this is a 6×4 array.

* The lines that follow initialize the array locations with IP addresses. The first index (0.. 5) represents the rooms 100 through 105. The second index (0.. 3) represent the four computers in each room.

* The IP address of the third computer in room 104 can be found in the array element or component ipAddress(4, 2). This value is " 192. 168. 10. 45". Look at the array carefully to determine the meaning of the index values.

Note: If you need to upload a file into the C: Scripts directory on vlab-PC1 or download a file from the vlab-PC1 C: Scripts file, open Windows Explorer (E). As you can see on the right, your local computer drives (in this case, the C:

and D:) are mapped to file copy and paste (or drag and drop) easy to accomplish.

Task 2: Add the Code to Write the Array Data to a File

Pseudocode| Comments|

Define the following constantsREAD = 1, WRITE = 2, APPEND = 8, ASCII = 0Define the variable fileName and initialize it to " IP_Addresses. csv" Define the variable ipAddrStr and initialize it to "" Set fso to the " ScriptingFileSystemObject" using CreateObject Set ipFileObj = fso. CreateTextFile(filename, True, ASCII)For room = 0 to 5 For computer = 0 to 3 ipAddrStr = CStr(room+100) & "," & CStr(computer+1) & "," & ipAddress(room, computer) & vbCrLf Use ipFileObj to Write(IpAddrStr) NextNext Close the fileSet ipFileObj = fso. OpenTextFile(fileName, READ, ASCII)WScript. Echo ipFileObj. ReadAllipFileObj. Close | Named constants should be all caps and the Const prefix should be used to differentiate it from a variable whose value can change. fileName is a variable that contains the name of the file we will write. ipAddrStr will be used later to store individual records that we will write to the file " C: ScriptsIP_Addresses. csv". You always need to use " ScriptingFileSystemObject" to read, write, or append files.

The CreateTextFile method has one required argument and two optional arguments. The first argument value is the file name as string constant or variable. The argument set to True means we will over-write the file if it exists. The third argument specifies the textfile format -1= Unicode, 0= ASCII, 1= system default format. Note the use of the variable filename. Once

ipFileObj is linked to a file, it can be used just like WScript. StdOut. You can use the Write( ), WriteLine( ), and WriteBlankLines( ) methods. Nested For loops are used to access the two-dimensional array of IP addresses. The outside loop specifies the room 0.. 5 (representing rooms 100.. 105) and the inside loop specifies the computer 0.. 3 (representing computers 1.. 4). In the first pass through the loop when room = 0 and computer = 0, ipAddrStr is set to the value" 100, 1, 192. 168. 10. 11". Close the IP_Addresses. csv file using ipFileObj. Close Here we are opening the IP_Addresses. csv file and displaying its contents to verify the File Creation and formatting. .| Add the code indicated by the pseudocode shown below to write the Array to a text file. Actual code and identifiers for variables and methods within the pseudocode will be in bold font. Carefully read the comments to the right of the pseudocodewill be in italics.

* Save your program with S.

* Press and enter cscript FileWrite. vbs. Click OK to start your script run. Your script will run in a docked Console window in NotePad++ as shown below.

* Another alternative for running this program is to open a Windows Command Prompt (CLI), change directory to the C: Scripts folder and run the program by entering the command cscript IP_FileWrite. vbs.

* Execute the dir *. csv command to verify the existence of the IP_Addresses. csv file.

* If your run has any errors or your file does not contain the data shown above, debug your program and re-run it until you have the correct results.

When you have achieved the correct results, copy and paste your IP_FileWrite. vbs program from NotePad++ into the specified textbox in your lab-report document. Also copy the Windows command prompt run and file verification into the specified textbox into your lab-report document.|

Task 3: Create the IP_AppendRead. vbs Program

The IP_AppendRead. vbs program will create data for a new room with four computers with assigned IP addresses. The program will append these four records (one for each IP address) to the IP_Addresses. csv file. Then the program will open the IP_Addresses. csv file and display the contents of the file in a one record per line descriptive format. The new room records that were appended should be displayed.

* Open NotePad++ and create a new file called IP_AppendRead. vbs. Save this new script file in the C: Scripts directory. Do not copy and paste any code from your IP_WriteFile. vbs program. This program should be written from scratch.

* Add the code indicated by the pseudocode shown below to append the new room data to IP_Addresses. csv. Actual code and identifiers for variables and methods within the pseudocode will be in bold font. Carefully read the comments to the right of the pseudocode; they will be in italics.

Pseudocode| Comments|

Add an appropriate programmer header

Define the following constantsREAD = 1, WRITE = 2, APPEND = 8, ASCII = 0Define the variable fileName and initialize it to " IP_Addresses. csv" Define the variable ipAddrStr and initialize it to "" Define newRoom and initial it to " 106" Define comp1_IP and initialize it to " 192. 168. 10. 59" Define comp2_IP and initialize it to " 192. 168. 10. 60" Define comp3_IP and initialize it to " 192. 168. 10. 61" Define comp4_IP and initialize it to " 192. 168. 10. 62" Set fso to the object " Scripting. FileSystemObject"

| Named constants should be all caps and the const prefix should be used to differentiate it from a variable whose value can change. fileName is a variable that contains the name of the file we will write. ipAddrStr will be used to build the record string that will be written to the file " C: ScriptsIP_Addresses. csv". The variables newRoom, comp1_IP, comp2_IP, comp3_IP, and comp4_IP are set to values that represent a new room (106) that contains four computers with specific IP addresses. Use the CreateObject( ) method to Set fso as an object of type: " Scripting. FileSystemObject".| Pseudocode| Comments|

ipAddrStr = _ newRoom & ", 1," & comp1_IP & vbCrLf & _ newRoom & ", 2," & comp2_IP & vbCrLf & _ newRoom & ", 3," & comp3_IP & vbCrLf & _ newRoom & ", 4," & comp4_IP & vbCrLfIf file identified by fileName does not exist Then Beep Speaker Twice

Display Message:

" File Does Not Exist!!!" & newline & " You Must Create the File Before You can Read the File !!" Quit ProgramEnd IfSet File object ipFileObj using fso. OpenTextFile( ) make sure file is set APPEND and ASCII format

Use ipFileObj object to append ipAddrStr to end of the file identified by fileNameClose the file| ipAddrStr is set to a value that represents the four PCs in the new room 106. Note the use of the Line continuation character _ . Files can be checked for existence using the fso method FileExists(). The NOT Boolean operator may be useful here. If C: ScriptsIP_Addresses. csv does not exist, Beep speaker twice and display error message. Exit the program with WScript. Quit. Set ipAddrFile to file fileName for APPEND and ASCII. Set ipFileObj = fso. OpenTextFile(fileName, APPEND, ASCII)Once ipFileObj is linked to a file, it can be used just like WScript. StdOut. You can use the Write( ), WriteLine( ), and WriteBlankLines( ) methods. Use the Close method of ipFileObj|

* At this point, Save your IP_AppendRead. vbs script using S. Run your program in NotePad++ with entering cscript IP_AppendRead. vbs or from the Windows Command Prompt in the C: Scripts directory.

* After running your program, use the type IP_Addresses. csv command from the Windows Command Prompt in C: Script or open the file in NotePad++ to verify that the new data for room 106 has been written.

C: Scripts> type IP_Addresses. csv

100, 1, 192. 168. 10. 11

100, 2, 192. 168. 10. 12 Important Note: After you have verified that the room 106 data has been 100, 3, 192. 168. 10. 13sucessfully written to IP_Addresses. csv, re-run your previously written 100, 4, 192. 168. 10.

14IP_FileWrite. vbs program to restore the original data. 101, 1, 192. 168. 10. 19

101, 2, 192. 168. 10. 20

101, 3, 192. 168. 10. 21

101, 4, 192. 168. 10. 22

102, 1, 192. 168. 10. 27

102, 2, 192. 168. 10. 28

102, 3, 192. 168. 10. 29

102, 4, 192. 168. 10. 30

103, 1, 192. 168. 10. 35

103, 2, 192. 168. 10. 36

103, 3, 192. 168. 10. 37

103, 4, 192. 168. 10. 38

104, 1, 192. 168. 10. 43

104, 2, 192. 168. 10. 44

104, 3, 192. 168. 10. 45

104, 4, 192. 168. 10. 46

105, 1, 192. 168. 10. 51

105, 2, 192. 168. 10. 52

105, 3, 192. 168. 10. 53

105, 4, 192. 168. 10. 54

106, 1, 192. 168. 10. 59Return to your program in NotePad++ and finish writing the remainder of your program 106, 2, 192. 168. 10. 60using the pseudocode found on the next page. 106, 3, 192. 168. 10. 61

106, 4, 192. 168. 10. 62

The pseudocode shown below will the steps needed to open and read the appended file IP_Addresses. cvs and display the information in the format

” The IP Address in Room ” room# ” for Computer ” computer# ” is ” ipAddress Pseudocode| Comments|

Using fso. OpenTextFile(), create the ipFileObj object that has opened filename for Reading in ASCII format. Do Until ipFileObj is at EndOfStream room = ipFileObject. Read 3 characters ipFileObj. Skip 1 character computer = ipFileObj. Read 1 character ipFileObj. Skip 1 character ipAddress = ipFileObj. Read 13 characters ipAddrFile. Skip newline Display Message: “ The IP Address in Room ” & room & ” for Computer “

& computer & ” is ” & ipAddressEnd LoopClose File
| Set ipFileObj to open fileName for Reading in ASCII format. Set ipFileObj = fso. OpenTextFile(fileName, READ, ASCII)Once ipFileObj is linked to a file, it can be used just like WScript. StdIn. You can use the Read( ), ReadLine( ), Skip( ) and SkipLine methods. The Do/Until loop will display all of the IP_Addresses. csv records including the room 106 records that were added. Use the Close method of ipFileObj

where room#, computer# and ipAddress are values read from the IP_Addresses. csv file.

Task 4: Finish Program and Run it Showing Error-Handling

* Finish writing your program and save it. Run your program using in NotePad++ or open a Windows Command Prompt and run it from the C:

Scripts directory. Your run should look like the following. Note the highlighted records in the run. These are the appended records for room 106.

* If you did not get the results shown on the right, you will need to debug your program and run it again until you get these results. To remove duplcate records or restore the original data in the IP_Addresses. csv file, re-run the IP_FileWrite. vbs program.

When you have achieved the correct results copy and paste your IP_AppendRead. vbs program from NotePad++ into the specified textbox in your lab-report document. Also copy the Run run into the specified textbox in your lab-report document.|

The . csv extension that we used on our IP_Addresses file indicates a Comma Separated Value format. This kind of file can be opened by Microsoft Excel and Microsoft Access. The file could be easily edited using Excel or even made into a database in Access by adding some field names. Below is our IP_Addresses. csv file opened in Excel (left) and Access (right).