WORKING WITH TEXT FILE

The **FileSystemObject** object is used to access the file system on a server.This object can manipulate files, folders, and directory paths. It is also possible to retrieve file system information with this object.

The **TextStream object** is used to access the contents of text filesThe **TextStream** object provides sequential access to the contents of any file where the contents are in text-readable form. You can create an instance of the**TextStream** object using the **CreateTextFile** or **OpenTextFile** methods of the **FileSystemObject** object

Methods of FileSystem & TextStream Object

Method	Description
OpenTextFile	Opens a file and returns a TextStream object that can be used to access the file
CreatTextFile	Creates a text file and returns a TextStream object that can be used to read from, or write to the file
FileExists	Checks if a specified file exists

Close	Closes an open TextStream file Syntax **object.close**
Read(n)	Reads a specified number of characters from a TextStream file and returns the result Syntax **object.Read(n)**
ReadAll	Reads an entire TextStream file and returns the result Syntax **object.Readall**
ReadLine	Reads one line from a TextStream file and returns the result Syntax **object.Readall**
Skip(n)	Skips a specified number of characters when reading a TextStream file Syntax **object.skeep(n)**
SkipLine	Skips the next line when reading a TextStream file Syntax **object.SkipLine**

Write	Writes a specified text to a TextStream file Syntax **object.Write(string)**
WriteLine	Writes a specified text and a new-line character to a TextStream file Syntax **object.WriteLine(String)**
WriteBlankLines(n)	Writes a specified number of new-line character to a TextStream file Syntax **object.WriteblankLines(n)**

Difference between AtEndOfLine & AtEndOfStream methods

AtEndOfLine	**AtEndOfStream**
Returns true if the file pointer is positioned immediately before the end-of line marker in a TextStream file, and false if not	Returns true if the file pointer is at the end of a TextStream file, and false if not

Create a text file called test.txt

```
<%
Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.CreateTextFile(Server.Mappath("ABCD2.txt")))
objFile.WriteLine("This is a test.")
objFile.Close
%>
```

Check a text file called ABCD2.txt exist or not

```
<%
Set objFSO = Server.CreateObject("Scripting.FileSystemObject")
If objFSO.FileExists(Server.Mappath("ABCD2.txt")) Then
Response.Write "File exists"
Else
Response.Write "File does not exist"
End If
objFile.close
Set objFile=nothing
Set objFSO=nothing
%>
```

Opens a file and displays the contents

```
<%
Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.OpenTextFile(Server.Mappath("test.txt"))
strText = objFile.ReadAll
response.write(strText)
objFile.close
Set objFile=nothing
Set objFSO=nothing
%>
```

Code to count the no of word

```
<% Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.OpenTextFile(Server.Mappath("test.txt"))
strText = objFile.ReadAll
objFile.Close
arrWords = Split(strText, " ")
Response.write(Ubound(arrWords) + 1 )
objFile.Close
objFile=nothing
objFSO=nothing
%>
```

To Display a line beginning with "A" in a file

```
<%  Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.OpenTextFile(Server.Mappath("test.txt"))
Do While Not objFile.AtEndofStream
 myLine = objFile.ReadLine
n=RIGHT(myLine,1)
If n== "A" Then
Response.Write(myLine)
End If
Loop
objFile.Close
objFile=nothing
objFSO=nothing
%>
```

To Display a replace all space with # in a file

```
<% Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.OpenTextFile(Server.Mappath("test.txt"))
strText = objFile.ReadAll
strNewText = Replace(strText, "", "# ")
Response.Write ( strNewText)
objFile.Close
objFile=nothing
objFSO=nothing
%>
```

Skip a line of a textfile

```
<% Set fs=Server.CreateObject("Scripting.FileSystemObject")
Set f=fs.OpenTextFile(Server.Mappath("test.txt"))
f.SkipLine
Response.Write(f.ReadLine)
f.Close
Set f=Nothing
Set fs=Nothing
%>
```

Writing Blank Line

```
<%
Set FSO = Server.CreateObject("Scripting.FileSystemObject")
Set obj = FSO.CreateTextFile(Server.MapPath("Abcd.txt"),true)
obj.WriteLine("I Love INDIA")
obj.WriteBlankLines(4)
obj.WriteLine("I Love My Live")
Response.write("Writing")
obj.Close
Set obj = Nothing
Set FSO = Nothing
%>
```

Counting the lines

```
<%
Set FSO = Server.CreateObject("Scripting.FileSystemObject")
Set obj = FSO.OpenTextFile(Server.MapPath("Abcd.txt"))
L=0
Do Until obj.AtEndOfStream
Response.Write obj.ReadLine & "<br>"
L=L+1
Loop
Response.Write(no of Lines=” & L+1)
obj.Close
Set obj = Nothing
Set FSO = Nothing
%>
```

Search Text in File

```
<%
strSearch = "hello"
Set FSO = Server.CreateObject("Scripting.FileSystemObject")
Set obj = FSO.OpenTextFile(Server.MapPath("Abcd.txt"))
Do Until obj.AtEndOfStream
strText = obj.ReadLine
If InStr(strText,strSearch) >0 Then
Response.Write strText & "<br>"
End IF
Loop
obj.Close
Set obj = Nothing
Set FSO = Nothing
%>
```

To Find the no of Lines Starting with A or I in a file abc.txt and displaying these lines in uppercase

```
<%
Set FileObj=Server.CreateObject("Scripting.FileSystemObject")
Set objfile= FileObj.OpenTextFIle(Server.MapPath("faips.txt"))
cv=0
Do WHILE objfile.AtEndOfStream<>True
ch=objfile.ReadLine
ch1=Left(ch,1)
if ch1="A" OR ch1="V" then
cv=cv+1
Response.write(Ucase(ch))
Response.Write("<Br>")
end if
LOOP
Response.Write(" No of Lines Starting with A or I in File =" & cv)
objFile.Close
objFile=nothing
objFSO=nothing
%>
```

To Display the content of file and Find the no of Lines and Words in a file faips.txt"

```
<%
Set FileObj=Server.CreateObject("Scripting.FileSystemObject")
Set objfile= FileObj.OpenTextFIle(Server.Mappath(faips.txt"))
cv=0
cl=0
Do WHILE objfile.AtEndOfStream<>True
ch=objfile.Read(1)
Response.write(ch)
if ch=" " then
cv=cv+1
elseif ch=Chr(13) then
cl=cl+1
end if
loop
Response.Write("<br><br><br>")
Response.Write(" No of Space in File =" & cv & "<br>")
Response.Write(" No of Words in File =" & cv+1 & "<br>")
Response.Write(" No of Lines in File =" & cl+1 & "<br>")
objFile.Close
objFile=nothing
objFSO=nothing
%>
```

To Find the no of Uppercase/Lowercase Letters/Digits in a file faips.txt"

```
<%
Set FileObj=Server.CreateObject("Scripting.FileSystemObject")
Set newsfile= FileObj.OpenTextFIle(Server.Mappath(“faips.txt"))
UP=0  ' uppercase letter
LO=0  ' lowercase letter
DI=0  ' Digit
sp=0  ' Space
cl=0  ' lines
Do WHILE newsfile.AtEndOfStream<>True
ch=newsfile.Read(1)
Response.Write(ch)
if ch>="A" and ch<="Z" then
UP=UP+1
elseif ch>="a" and ch<="z" then
LO=LO+1
elseif ch>="0" and ch<="9" then
DI=DI+1
end if
loop
Response.Write("<br><br>")
Response.Write(" No of Uppercase letter=  "  & UP & "<br>")
Response.Write(" No of Lowercase letter=  "  & LO & "<br>")
Response.Write(" No of Digit=  "  & DI & "<br>")
%>
```

Study the code given below (Answer Yourself)

```
<%
Set FileObj=_____ ("Scripting.FileSystemObj ect")
Set mfile= FileObj._____("d:\contentvnlths.txt")
Do UNTIL newsfile._____
ch= newsfile.Read(1)
If ch=". " Then
Ch="; "
end if
Response.Write(ch)
Loop
newsfile._____
%>
```

1. Fill in the blanks to complete the above code fragment that reads the text in the file Headlines.txt and displays the same on the web page after converting all full stops (.) to semi-colons (;).
2. Rewrite the code for the loop using DO WHILE instead of DO LTNTIL
3. How would the output of the above script change if we used the Readline method instead ofRead(l) ?