



**CSE 1530**

**Introduction to Computer Use II:  
Programming**

Winter 2006 (Section M)

Topic F: External Files and Databases -  
Using Classes and Objects

Wednesday, March 29 2006

Bill Kapralos

CSE 1530, Winter 2006, Bill Kapralos

### Overview (1):

- **Before We Begin**
  - Some administrative details
  - Some questions to consider
- **Reading/Displaying the Contents of a File**
  - Opening a file
  - Reading the Contents of a file
  - Closing a file
- **Writing to a File**
  - Overview
  - Writing to a new file → example program

## Before We Begin

### Administrative Details (1):

- **Exercise 7-6**
  - Due Monday, April 3 2006 before noon
  - I will be in the Glade Lab today
  - We will use a portion of Friday's lecture to go over the exercise in greater detail
- **Last Lecture is Monday, April 3**
  - Entire lecture will be review for exam

### Some Questions to Consider (1):

- What is a reference ?
- What must we do to allow us to incorporate the required file-related references ?
- How can we create (instantiate) an object of some class (two steps) ?
- What are the two keywords required to create an object ?

## Reading & Displaying the Contents of a File (cont.)

## Recall (1):

### ▫ The OpenTextFile Function

- Lets take a closer look at the description for the `OpenTextFile` method

```
Function OpenTextFile(FileName As String, _  
    [IOMode As IOMode = ForReading], _  
    [Create As Boolean = False], _  
    [Format As Tristate = TristateFalse]) _  
    As TextStream  
Member of Scripting.FileSystemObject  
Open a file as a TextStream
```

## Recall (2):

### ▫ The OpenTextFile Function (cont.)

- Function description tells you how to use the method
  - Provides a description of the arguments and their type
  - The first argument is of type `String` and it denotes the name of the file you wish to open (e.g., the name of the file that was obtained using the Open File dialog)
  - Returns an object of type `TextStream`
  - Should be familiar with using the return value of a function → nothing new except the return value is an object!

## Opening a File (1):

### ▫ Using the OpenTextFile Function

- We can now use the function to open the desired file
  - We know the return type
  - We know the number of arguments and their type → we will pass the one required argument and leave the other three optional arguments to their default value
- First thing to do is to define a reference (variable) of type `TextStream` → nothing new!
  - `Dim myTextStream As textStream`

## Opening a File (2):

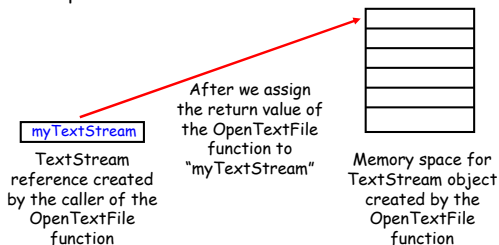
### ▫ Using the OpenTextFile Function (cont.)

- Recall the two steps to creating (instantiating an object) → we have created the reference (name) and now we need to actually create the object (e.g., using the `New` keyword)
  - But, since we are assigning the return value of a function to the previously declared variable, there is no need to actually create the object → it has already been created by the function!
  - We are simply copying the object created by the function to our reference (name)

## Opening a File (3):

### ▫ Using the OpenTextFile Function (cont.)

- Graphical illustration



## Opening a File (4):

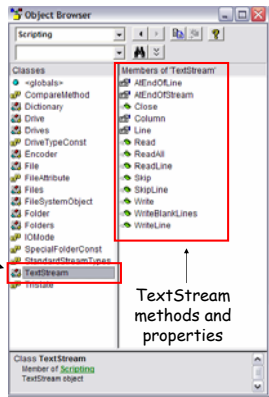
### ▫ Now We Can Open a File and Obtain a TextStream Object

- But how do we access (read) the contents of the file we just opened? → using the `TextStream` object!
- The `TextStream` object contains various methods to access the file contents
  - `Read`, `ReadAll`, `ReadLine`, `Write`, `WriteBlankLines` and `WriteLine`
  - Once again, you can obtain information via the Object Browser

### Opening a File (5):

- **TextStream Object**
  - Object Browser

TextStream class

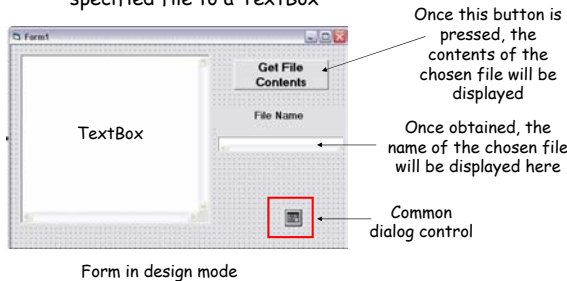


### Reading the Contents of a File (1):

- **Example → Reading The Entire Contents of a File and Displaying it in a TextBox**
  - Add a TextBox to a Form
    - Set its **MultiLine** property to **True**
    - Set its **ScrollBar** property to **Both**
  - We will use the **ReadAll** method of the TextStream class to obtain the entire contents of the file
    - Return type of the function is String
    - Assign the return type of the function to the TextBox we just created → its that simple!

### Reading the Contents of a File (2):

- **Putting it All Together**
  - Develop a program that displays contents of a user-specified file to a TextBox



### Reading the Contents of a File (3):

- **Putting it All Together (cont.)**
  - Lets look at the Visual Basic code...

### Closing a File (1):

- **All Files Should Be Closed**
  - Once we are done with the file (e.g., the file is no longer needed), it should always be closed
    - Once the file is closed, its contents can no longer be accessed
    - Closing the file ensures there is no un-intended changes made to it
  - A file can be closed by using the "Close" method of the TextStream class
    - **FileContent.Close** → FileContent is the name of the TextStream object we previously defined