

# Visual Web Development

Terry Marris November 2007

## 19 Text Files

The problem with the birthday book (chapter 18) is that birthdays entered and stored on one day have to be re-input the next day. We need to preserve important data from day to day. We need a persistent data store. We need a file.

### 19.1 Line Structure

Some entries in the birthday book might be:

```
Tom    16 June
Dick   12 May
Harry  5 September
```

Each entry is a line of text. How can we distinguish between name and birthday? Names have different lengths and so do birthdays.

1. use a fixed size for name and for birthday

tom	16 June
dick	12 May
harry	5 September

2. separate name from birthday with a comma

```
tom,16June
dick,12 May
harry,5 September
```

Recalling our efforts with the String class (chapter 14): if we went with option 1 we could use Substring:

```
Dim strName = strLine.Substring(0, nameSize)
Dim strBirthday = strLine.Substring(nameSize, strLine.Size() - 1)
```

If we went with option 2 we could use Split:

```
Const comma As String = ","
Dim array() As String = Split(strLine, comma, 2)
Dim strName As String = array(0)
Dim strBirthday As String = array(1)
```

We shall choose option 2 because then we do not have to worry about how many characters there are in a person's name.

## 19.2 Writing to a Text File

The first thing is to import the VB namespaces that include the VB file handling methods.

```
Imports System.IO
Imports My
```

These must be the very first statements in the VB file.

We define some global constants.

```
' GLOBAL CONSTANTS
Const pathName As String = "E:\VisualWebDevelopment\TextFiles\"
Const fileName As String = "BirthdayBookFile.txt"
Const comma As String = ","
Const space As String = " "
Const blank As String = ""
```

pathName is where the text file is to be found. fileName is the name of the text file. We also define comma, space and blank because their use might make programming code easier to read - see later on.

To write a line of text to a file we:

- create a line of text from the input

```
Dim strLine As String = txtName.Text.ToString() + comma +  
                        txtBirthday.Text.ToString()
```

- open the text file for writing. This involves creating a link between the internal filename as its is known to the program (file), and the external filename as it is known to the operating system ("E:\VisualWebDevelopment\TextFiles\BirthdayBookFile.txt).

```
Dim file As StreamWriter =  
    Computer.FileSystem.OpenTextFileWriter(pathName + fileName, True)
```

True means create the file if it does not exist, or append lines of text to the file if it does exist

- write the line of text to the file

```
file.WriteLine(strLine)
```

- close the file

```
file.Close()
```

Closing the file involves flushing buffers holding data to be written to the file and breaking the connection between internal and external filenames.

File handling is fraught with errors waiting to happen. The path to the file might be wrong or the media (memory stick for example) might be full or missing or, for some reason, the line of text could not be written or read. So we always contain our file handling routines within Try ... Catch. Here, we have used a "catch all" case.

```
Try  
...  
    Dim strLine As String = txtName.Text.ToString() + comma +  
                            txtBirthday.Text.ToString()  
    Dim file As StreamWriter =  
        Computer.FileSystem.OpenTextFileWriter(pathName + fileName, True)  
    file.WriteLine(strLine)  
    file.Close()  
    ...  
Catch ex As Exception  
    lblError.Text = "Error writing to file: " + ex.ToString()  
End Try
```

## 19.3 Reading From a Text File

To retrieve a line of text from a file we:

- open the text file for reading

```
Dim file As StreamReader =  
    Computer.FileSystem.OpenTextFileReader(pathName + fileName)
```

- loop for as long as the end of the file has not been reached

```
While Not file.EndOfStream()  
    ...  
End While
```

- retrieve a line of text each time round the loop

```
Dim strLine As String = file.ReadLine()
```

- process the line of text. Here, we are displaying the line of text in a list box.

```
lstBirthdays.Items.Add(strLine)
```

- close the file when there are no more lines to be retrieved from the file

```
file.Close()
```

As always we include our file handling with a Try ... Catch.

```
Try  
    ...  
    Dim file As StreamReader =  
        Computer.FileSystem.OpenTextFileReader(pathName + fileName)  
    While Not file.EndOfStream()  
        Dim strLine As String = file.ReadLine()  
        ...  
        lstBirthdays.Items.Add(strLine)  
    End While  
    file.Close()  
Catch ex As Exception  
    lblError.Text = "Error reading from file: " + ex.ToString()  
End Try
```

## 19.4 The Entire VB File

The entire VB file is shown below.

```
Imports System.IO
Imports My

Partial Class _Default
    Inherits System.Web.UI.Page

    ' GLOBAL CONSTANTS
    Const pathName As String = "E:\VisualWebDevelopment\TextFiles\"
    Const fileName As String = "BirthdayBookFile.txt"
    Const comma As String = ","
    Const space As String = " "
    Const blank As String = ""

    Protected Sub Page_Load(ByVal sender As Object,
                            ByVal e As System.EventArgs) Handles Me.Load

    End Sub

    Protected Sub btnAdd_Click(ByVal sender As Object,
                               ByVal e As System.EventArgs) Handles btnAdd.Click
        Try
            lstBirthdays.Items.Clear()
            If txtName.Text.Length() = 0 Or txtBirthday.Text.Length() = 0
            Then
                Return ' empty fields
            End If
            Dim strLine As String = txtName.Text.ToString() + comma +
                                   txtBirthday.Text.ToString()

            Dim file As StreamWriter =
            Computer.FileSystem.OpenTextFileWriter(pathName + fileName, True)
            file.WriteLine(strLine)
            file.Close()
            txtName.Text = blank
            txtBirthday.Text = blank
        Catch ex As Exception
            lblError.Text = "Error writing to file: " + ex.ToString()
        End Try
    End Sub

    Protected Sub Calendar_SelectionChanged(ByVal sender As Object, ByVal e
        As System.EventArgs) Handles Calendar.SelectionChanged
        txtBirthday.Text = Calendar.SelectedDate.ToLongDateString()
    End Sub
```

```

Protected Sub btnList_Click(ByVal sender As Object,
                            ByVal e As System.EventArgs) Handles btnList.Click
    Try
        lstBirthdays.Items.Clear()
        Dim file As StreamReader =
            Computer.FileSystem.OpenTextFileReader(pathName + fileName)
        While Not file.EndOfStream()
            Dim strLine As String = file.ReadLine()
            Dim strArray() As String = Split(strLine, comma)
            Dim strName As String = strArray(0)
            Dim strBirthday As String = strArray(1)
            strLine = strName + comma + space + strBirthday
            lstBirthdays.Items.Add(strLine)
        End While
        file.Close()
    Catch ex As Exception
        lblError.Text = "Error reading from file: " + ex.ToString()
    End Try
End Sub

Protected Sub btnFind_Click(ByVal sender As Object,
                            ByVal e As System.EventArgs) Handles btnFind.Click
    Try
        Dim boolFound As Boolean = False
        lstBirthdays.Items.Clear()
        Dim file As StreamReader =
            Computer.FileSystem.OpenTextFileReader(pathName + fileName)
        While Not file.EndOfStream()
            Dim strLine As String = file.ReadLine()
            Dim strArray() As String = Split(strLine, comma)
            Dim strName As String = strArray(0)
            Dim strBirthday As String = strArray(1)
            strLine = strName + comma + space + strBirthday
            strName = strName.ToLower()
            txtName.Text = txtName.Text.ToLower()
            If strName.Equals(txtName.Text.ToString()) Then
                boolFound = True
                lstBirthdays.Items.Add(strLine)
            End If
        End While
        file.Close()
        If Not boolFound Then
            lstBirthdays.Items.Add(txtName.Text + " not found")
        End If
        txtName.Text = blank
    Catch ex As Exception
        lblError.Text = "Error reading from file: " + ex.ToString()
    End Try
End Sub
End Class

```

## 19.5 Exercises

1. Explain each line of the Birthday Book program shown above in §19.4
2. Try out the birthday Book program.