

Lab 08a: Reading a Text File

An application can interact with files and data that are external to itself and user's inputs. In this exercise, we will write an application that reads the content of a text file.

Note that as external files and data may not always present, due to the facts that they are separated from the application, it is a good practice to use an exception handling technique within the codes.

File needed

A text file is needed for this exercise. Create and place a text file within the same execution directory as the application. The text file extension can be of any type. In fact, we can even create our own file extension. In this example, we will use file extension *.helloext*.

Type some sentences into the text file and save.

Method in reading a text file

There are a number of ways to read a text file. Most common ones are *System.IO.File* and *System.IO.StreamReader*. In this exercise, we will use *System.IO.File*.

Read content at its entirety

```
<variable> = System.IO.File.ReadAllText ("<textfile.ext>");  
string s_Content = System.IO.File.ReadAllText ("textfile.helloext");
```

Read content line by line, placing each line into a slot in an array variable

```
<array variable> = System.IO.File.ReadAllLines ("<textfile.ext>");  
string[] as_ContentByLines = System.IO.File.ReadAllLines ("textfile.helloext");
```

Sample Code - Reading content of a text file, all at once

```
//The following code reads content of a text file in its entirety
//The content is then placed into a string variable
//Use try..catch technique, in case the file is non-existence
try
{
    //Declare a string variable to receive the content of a text file
    string s_TextContent;

    //Open and read the text file, placing the content into a string variable
    //ReadAllText built-in method reads the entire content at once
    s_TextContent = System.IO.File.ReadAllText("textfile.helloext");

    //Display the content into screen, to see if it works
    Console.WriteLine(s_TextContent);
}
//Give an error message if file is not found
catch
{
    Console.WriteLine("File not found!");
}
```

Sample Code - Reading content of a text file, a line at a time

```
try
{
    //Declare a string array variable, and placing content of a text file into the array
    //As the content is read line by line, it is placed into an array, index by index
    string[] as_TextContentByLines = System.IO.File.ReadAllLines("textfile.helloext");

    //Display the content into screen, to see if it works
    //Using For loop to cycle through the array index to display the content
    for (int i_count = 0; i_count < as_TextContentByLines.Length; i_count++)
    {
        Console.WriteLine(as_TextContentByLines[i_count]);
    }

    //The specific content of an array can also be displayed by specifying its index
    Console.WriteLine(as_TextContentByLines[2]);
}
catch
{
    Console.WriteLine("File not found or array index out of bound!");
}
```

Full codes

```

static void Main(string[] args)
{
    //The following code reads content of a text file in its entirety
    //The content is then placed into a string variable
    //Use try..catch technique, in case the file is non-existence
    try
    {
        //Declare a string variable to receive the content of a text file
        string s_TextContent;

        //Open and read the text file, placing the content into a string variable
        //ReadAllText built-in method reads the entire content at once
        s_TextContent = System.IO.File.ReadAllText("textfile.helloext");

        //Display the content into screen, to see if it works
        Console.WriteLine(s_TextContent);
    }
    //Give an error message if file is not found
    catch
    {
        Console.WriteLine("File not found!");
    }

    try
    {
        //Declare a string array variable, and placing content of a text file into the array
        //As the content is read line by line, it is placed into an array, index by index
        string[] as_TextContentByLines = System.IO.File.ReadAllLines("textfile.helloext");

        //Display the content into screen, to see if it works
        //Using For loop to cycle through the array index to display the content
        for (int i_count = 0; i_count < as_TextContentByLines.Length; i_count++)
        {
            Console.WriteLine(as_TextContentByLines[i_count]);
        }

        //The specific content of an array can also be displayed by specifying its index
        Console.WriteLine(as_TextContentByLines[2]);
    }
    catch
    {
        Console.WriteLine("File not found or array index out of bound!");
    }
}

```