Object-oriented Programming with C#

Tahaluf Training Center 2021

Day 7

1. **C# FileInfo**
2. StreamWriter in C#
3. StreamReader in C#

File Info

The FileInfo class provides more control on read/write operations on files by writing code manually for reading or writing bytes from a file.

To work with files in .NET framework, the important namespace used is system.IO namespace and similarly.

The constructors of the FileInfo class are explained as below:

1. **FileInfo(string):** A new instance of the FileInfo class is initialized and it acts as a wrapper for the path of the file.

2. **Attributes:** We can get or set the attributes for the current file or the current directory using Attributes property.

3. CreationTime: We can get or set the creation time for the current file or the current directory using Creation Time property.

4. Directory: We can get an instance of the parent directory using Directory property.

5. DirectoryName: We can get a string that represents the full path of the directory using Directory Name property.

6.Exists: We can get a value that indicates if a file exists or no using Exists property.

7.FullName: We can get the full path of the directory or the full path of the file using Full Name property.

8.IsReadOnly: We can get or set a value that can determine if the current file has read-only property using Is Read Only property.

9. LastAccessTime: We can get or set the time at which the current file or the current directory was last accessed by using the Last access time property.

10. Length: We can get the size of the current file in bytes using the length property.

11. Name: We can get the name of the file by using the name property.

Lets have a Demo

Create File

```
try
    {
        // the file location is specified where the file is to be created
        string location = @"C:\Users\User\Desktop\Tahaluf\OOP\Day 7\new.txt";
        // instance of the fileinfo class is created
        FileInfo file = new FileInfo(location);
        // an empty file is created
        file.Create();
        Console.WriteLine("Creation of file is successfull");
    }
    catch (IOException e)
    {
        Console.WriteLine("Failed attempt to create file " + e);
    }
```

Delete File

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
// the file location is specified where the file is to be
located
string location = @"C:\Users\User\Desktop\Tahaluf\OOP\Day
7\new.txt";
// instance of the fileinfo class is created
FileInfo file = new FileInfo(location);
// The specified file is deleted
file.Delete();
Console.WriteLine("Deletion of file is successfull");
```

Day 7

1. C# FileInfo
2. **StreamWriter in C#**
3. StreamReader in C#

Stream Writer

To write the characters into a stream that follows a specific encoding, we make use of the class called StreamWriter class in C# and the method StreamWriter.

Write() methods of StreamWriter class is responsible for writing characters into a stream.

TextWriter class is the base class of StreamWriter class that is the StreamWriter class is inherited from the TextWriter class and this TextWriter class provides several methods which can be used to write an object to a string, writing strings to a file, etc..

Working of StreamWriter class in C#:

- Streams are used in file operations of C# to read data from the files and to write data into the files.
- The stream makes the file is being read smoothly and the data is written to the file smoothly.

Stream Writer

- The data from the large files are broken down into small chunks and then sent to the stream.

- Then the application reads these chunks of data from the stream rather than trying to read the whole data at once. This is the advantage of using streams.

- The reason why the data from the files is broken into small chunks is that there is an impact on the performance of the application when the application tries to read the whole data from the file at once.

- So, whenever data is to be written into a file, the data is first written to the stream, and then the data is written to the file from the stream.

Stream Writer

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

```
//the path of the file and the file name is assigned to a string
variable
String pat = @"C:\Users\User\Desktop\new.txt";
//an instance of the string writer class is created, and the path of
the file is passed as a parameter to append text to the file
using (StreamWriter sw = File.AppendText(pat))
{
//data to be appended to the file is included
sw.WriteLine("Welcome to StreamWriter class in C#");
//the instance of the streamwriter class is closed after writing data
to the File
sw.Close();
//data is read from the file by taking the path of the file as
parameter
Console.WriteLine(File.ReadAllText(pat));
}
Console.ReadKey();
```

Create a program in C# to request lines from the user and write them in a text file.

Solution


```
string fileName = "out.txt";
using (StreamWriter file = File.CreateText(fileName))
{
    string line;
    do
    {
        line = Console.ReadLine();

        if (line.Length != 0)
        {
            file.WriteLine(line);
        }
    }
    while (line.Length != 0);
}
```

Day 7

1. C# FileInfo
2. StreamWriter in C#
3. **StreamReader in C#**

Stream Reader

To read the characters into a stream that follows a specific encoding, we make use of the class called StreamReader class in C# and the method StreamWriter.Read() method of StreamReader class is responsible for reading the next character or the next set of characters from the stream.

TextReader class is the base class of StreamReader class that is the StreamReader class is inherited from the TextReader class and this TextReader class provides several methods which can be used to read a character, block, line, etc.

System.IO.namespace is the namespace in which the StreamReader is defined and StreamReader class provides several read methods such as Peak, Read, ReadAsync, ReadBlock, ReadBlockAsync, ReadLine, ReadLineAsync, ReadToEnd, ReadToEndAsync, etc.

Stream Reader


```
//the path of the file and the file name is assigned to a string variable
String def = @"C:\Users\User\Desktop\Tahaluf\OOP\Day 7\new.txt";

//an instance of the string writer class is created, and the path of the file is passed as a parameter to
append text to the file
using (StreamWriter stwr = File.AppendText(def))
            {
                //data to be appended to the file is included
                stwr.WriteLine("Welcome to StreamjjjhbjkWriter class in C#");
                //the instance of the streamwriter class is closed after writing data to the File
                stwr.Close();
                try
                {
                    // an instance of stream reader class is created, and data is read from the file by
taking the path of the file as parameter
                    using (StreamReader read = new StreamReader(def))
                    {
                        //a string variable is defined
                        string line1;
                        // Data is being read one line after the other
                        while ((line1 = read.ReadLine()) != null)
                        {
                            Console.WriteLine(line1);


                        }
                    }
                }
                catch (Exception e)
                {
                    Console.WriteLine(e.Message);
                }
                Console.ReadKey();
```

Create a C# program that reads a text file and displays all its lines on the screen.

Write a program in C# to read a text file and make a copy in another file by changing the lowercase letters to uppercase.

On the E-Learning Portal