C# - READING FROM AND WRITING TO TEXT FILES

The **StreamReader** and **StreamWriter** classes are used for reading from and writing data to text files. These classes inherit from the abstract base class Stream, which supports reading and writing bytes into a file stream.

The StreamReader Class

The **StreamReader** class also inherits from the abstract base class TextReader that represents a reader for reading series of characters. The following table describes some of the commonly used **methods** of the StreamReader class –

Sr.No.	Method & Description
1	**public override void Close** It closes the StreamReader object and the underlying stream, and releases any system resources associated with the reader.
2	**public override int Peek** Returns the next available character but does not consume it.
3	**public override int Read** Reads the next character from the input stream and advances the character position by one.

Example

The following example demonstrates reading a text file named Jamaica.txt. The file reads –

```
Down the way where the nights are gay
And the sun shines daily on the mountain top
I took a trip on a sailing ship
And when I reached Jamaica
I made a stop
```

```
using System;
using System.IO;
```

```
namespace FileApplication {
   class Program {
      static void Main(string[] args) {
         try {
            // Create an instance of StreamReader to read from a file.
            // The using statement also closes the StreamReader.
            using (StreamReader sr = new StreamReader("c:/jamaica.txt")) {
               string line;

               // Read and display lines from the file until
               // the end of the file is reached.
               while ((line = sr.ReadLine()) != null) {
                  Console.WriteLine(line);
               }
            }
         } catch (Exception e) {
            // Let the user know what went wrong.
            Console.WriteLine("The file could not be read:");
            Console.WriteLine(e.Message);
         }
         Console.ReadKey();
      }
   }
}
```

Guess what it displays when you compile and run the program!

The StreamWriter Class

The **StreamWriter** class inherits from the abstract class TextWriter that represents a writer, which can write a series of character.

The following table describes the most commonly used methods of this class –

Sr.No.	Method & Description
1	**public override void Close** Closes the current StreamWriter object and the underlying stream.
2	**public override void Flush** Clears all buffers for the current writer and causes any buffered data to be written to the underlying stream.
3	**public virtual void Write***boolvalue*

	Writes the text representation of a Boolean value to the text string or stream. *InheritedfromTextWriter*.
4	**public override void Write***charvalue* Writes a character to the stream.
5	**public virtual void Write***decimalvalue* Writes the text representation of a decimal value to the text string or stream.
6	**public virtual void Write***doublevalue* Writes the text representation of an 8-byte floating-point value to the text string or stream.
7	**public virtual void Write***intvalue* Writes the text representation of a 4-byte signed integer to the text string or stream.
8	**public override void Write***stringvalue* Writes a string to the stream.
9	**public virtual void WriteLine** Writes a line terminator to the text string or stream.

For a complete list of methods, please visit Microsoft's C# documentation.

Example

The following example demonstrates writing text data into a file using the StreamWriter class –

Live Demo

```
using System;
using System.IO;

namespace FileApplication {
   class Program {
      static void Main(string[] args) {
         string[] names = new string[] {"Zara Ali", "Nuha Ali"};

         using (StreamWriter sw = new StreamWriter("names.txt")) {

            foreach (string s in names) {
               sw.WriteLine(s);
            }
         }

         // Read and show each line from the file.
         string line = "";
         using (StreamReader sr = new StreamReader("names.txt")) {
            while ((line = sr.ReadLine()) != null) {
               Console.WriteLine(line);
            }
         }
         Console.ReadKey();
      }
   }
}
```

When the above code is compiled and executed, it produces the following result –

```
Zara Ali
Nuha Ali
```