VB.NET - 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:

S.N	Method Name & Purpose
1	**Public Overrides Sub Close** It closes the StreamReader object and the underlying stream and releases any system resources associated with the reader.
2	**Public Overrides Function Peek As Integer** Returns the next available character but does not consume it.
3	**Public Overrides Function Read As Integer** Reads the next character from the input stream and advances the character position by one character.

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
```

```
Imports System.IO
Module fileProg
   Sub Main()
      Try
          ' Create an instance of StreamReader to read from a file.
          ' The using statement also closes the StreamReader.
          Using sr As StreamReader = New StreamReader("e:/jamaica.txt")
              Dim line As String
              ' Read and display lines from the file until the end of
              ' the file is reached.
              line = sr.ReadLine()
              While (line <> Nothing)
                  Console.WriteLine(line)
                  line = sr.ReadLine()
              End While
          End Using
      Catch e As Exception
```

```
            ' Let the user know what went wrong.
            Console.WriteLine("The file could not be read:")
            Console.WriteLine(e.Message)
        End Try
        Console.ReadKey()
    End Sub
End Module
```

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 shows some of the most commonly used methods of this class:

S.N	Method Name & Purpose
1	**Public Overrides Sub Close** Closes the current StreamWriter object and the underlying stream.
2	**Public Overrides Sub Flush** Clears all buffers for the current writer and causes any buffered data to be written to the underlying stream.
3	**Public Overridable Sub Write** ***valueAsBoolean*** Writes the text representation of a Boolean value to the text string or stream. *InheritedfromTextWriter.*
4	**Public Overrides Sub Write** ***valueAsChar*** Writes a character to the stream.
5	**Public Overridable Sub Write** ***valueAsDecimal*** Writes the text representation of a decimal value to the text string or stream.
6	**Public Overridable Sub Write** ***valueAsDouble*** Writes the text representation of an 8-byte floating-point value to the text string or stream.
7	**Public Overridable Sub Write** ***valueAsInteger*** Writes the text representation of a 4-byte signed integer to the text string or stream.
8	**Public Overrides Sub Write** ***valueAsString*** Writes a string to the stream.
9	**Public Overridable Sub WriteLine**

Writes a line terminator to the text string or stream.

The above list is not exhaustive. For complete list of methods please visit Microsoft's documentation

Example:

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

```
Imports System.IO
Module fileProg
   Sub Main()
      Dim names As String() = New String() {"Zara Ali", _
         "Nuha Ali", "Amir Sohel", "M Amlan"}
      Dim s As String
      Using sw As StreamWriter = New StreamWriter("names.txt")
          For Each s In names
              sw.WriteLine(s)
          Next s
      End Using
      ' Read and show each line from the file.
      Dim line As String
      Using sr As StreamReader = New StreamReader("names.txt")
          line = sr.ReadLine()
          While (line <> Nothing)
              Console.WriteLine(line)
              line = sr.ReadLine()
          End While
      End Using
      Console.ReadKey()
   End Sub
End Module
```

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

```
Zara Ali
Nuha Ali
Amir Sohel
M Amlan
```