



Strings and Characters



Strings

“ A string is a series of characters treated as a single unit.”

String literals are the constants and are of the form:

“Hello world”

“This is a string literal”

The class String

There are several constructors for initializing strings in the class String.

If characterArray is an array of characters, then the following utilize constructors:

```
St1 = new string(characterArray);
```

```
St2 = new string(characterArray, 6, 3);
```

```
St3 = new string('C',5);
```

Look at the example on page 635 of your text.

CopyTo method and Length Property

In the following *instance* represents an instance of a string.

`instance.Length` – We've seen this.

`instance.CopyTo(sourceIndex, destination, destinationIndex, count)`



```
instance.CopyTo(sourceIndex, destination,  
destinationIndex, count)
```

```
instance.CopyTo(int,char[],int,int)
```

Copies the characters in the string instance starting at the sourceindex into the character array destination beginning at location destinationindex. The variable count contains the number of characters to copy.

More string methods:

`instance.StartsWith("Here")`

Returns true if the instance string starts with the characters H, e, r, and e, and false otherwise

`instance.EndsWith("st")`

Returns true if the instance string ends with the string "st"; i.e, the last two characters are s and t, and false otherwise.

Yet More Methods

`instance.IndexOf(string)`

`instance.IndexOf(string,int)`

`instance.IndexOf(string,int,int)`

`instance.LastIndexOf(string)`

`instance.LastIndexOf(string,int)`

`instance.LastIndexOf(string,int,int)`

More!

`instance.IndexOfAny(char[])`

`instance.IndexOfAny(char[],int)`

`instance.IndexOfAny(char[],int,int)`

`instance.LastIndexOfAny(char[])`

`instance.LastIndexOfAny(char[],int)`

`instance.LastIndexOfAny(char[],int,int)`

To See How They Work

We will demonstrate the methods on the previous two slides with the program in figure 15.6 in your text. That program is available on the CD that came with your book.

Extracting Substrings (15.8)

`instance.Substring(int)`

Extract the substring starting at `int` and going to the end.

`instance.Substring(int,int)`

Extract the substring starting at the first index and taking the second `int` number of characters.

Miscellaneous String Methods

`instance.ToUpper()`

`instance.ToLower()`

`instance.Replace(char,char)`

`instance.Trim()`

StringBuilder

We will not consider the StringBuilder class. It deals with dynamic strings and capacities. So, skip Sections 15.11, 15.12, 15.13 and 15.14.

char Methods

Char.IsLetter(inputCharacter)

Char.IsLower(inputCharacter)

Char.IsUpper(inputCharacter)

Char.IsDigit(inputCharacter)

Char.IsPunctuation(inputCharacter)

Char.IsSymbol(inputCharacter)

Regular Expression

The last section in Chapter 15 deals with implementing regular expressions, to verify that strings satisfy certain properties. The topic of regular expressions is beyond the scope of this course, so we will omit this section.