High School Programming

Lecture: 13

WELCOME TO

Mahinur Rahaman Hridoy

BSc in CSE, Diploma
Software Developer

Recap Previous Lecture

- Arrays
- Logical Problem Solving

Agenda

- String

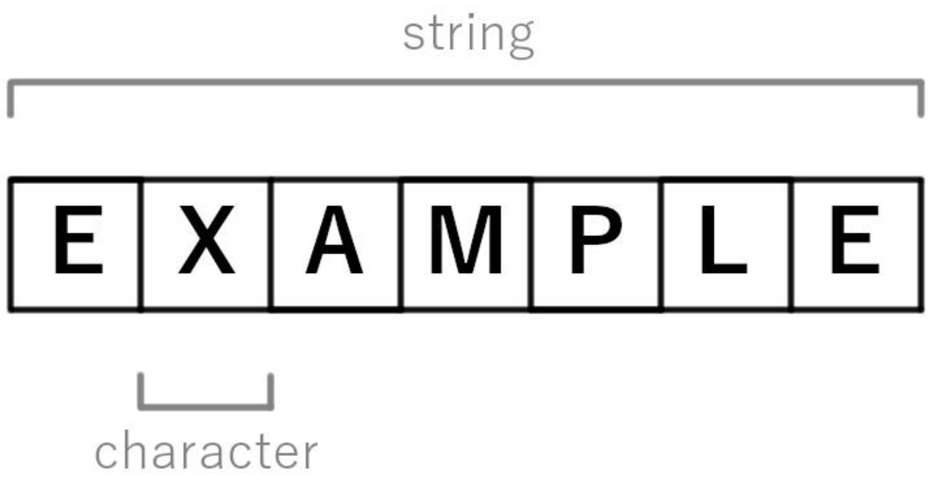
Strings

--In C#, the **string** is an object of the String class that represents a sequence of characters.

A **string** variable contains a collection of characters surrounded by double quotes:

Example-1: string greeting = “Hello”;

Example-2: string greeting2 = “Nice to meet you!”;



string

E X A M P L E

character

String build-in methods:

1. ToUpper()
2. ToLower()
3. Trim()
4. Copy()
5. Contains()
6. And more..

String build-in property:

1. ToUpper()
2. ToLower()
3. Trim()
4. Copy()
5. Contains()
6. And more..

Concatenation

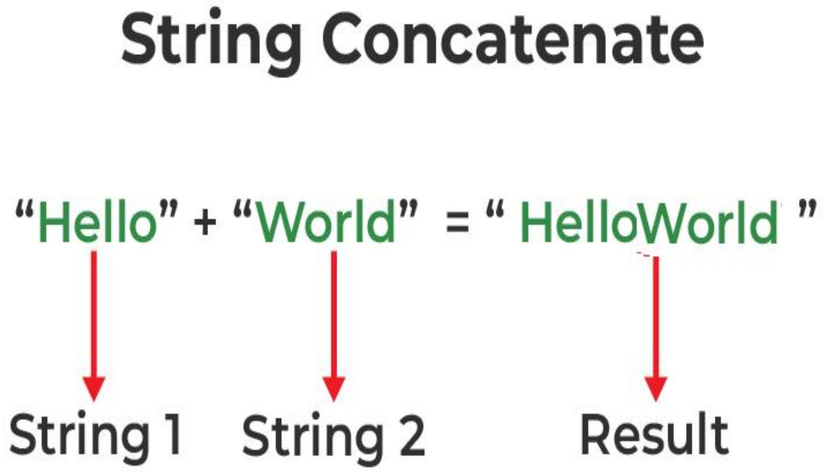
--The **+** operator can be used between strings to combine them. This is called **concatenation**.

Example:

```
string firstName = "John ";
string lastName = "Doe";
string name = firstName + lastName;
Console.WriteLine(name);
```

Alternative Method:

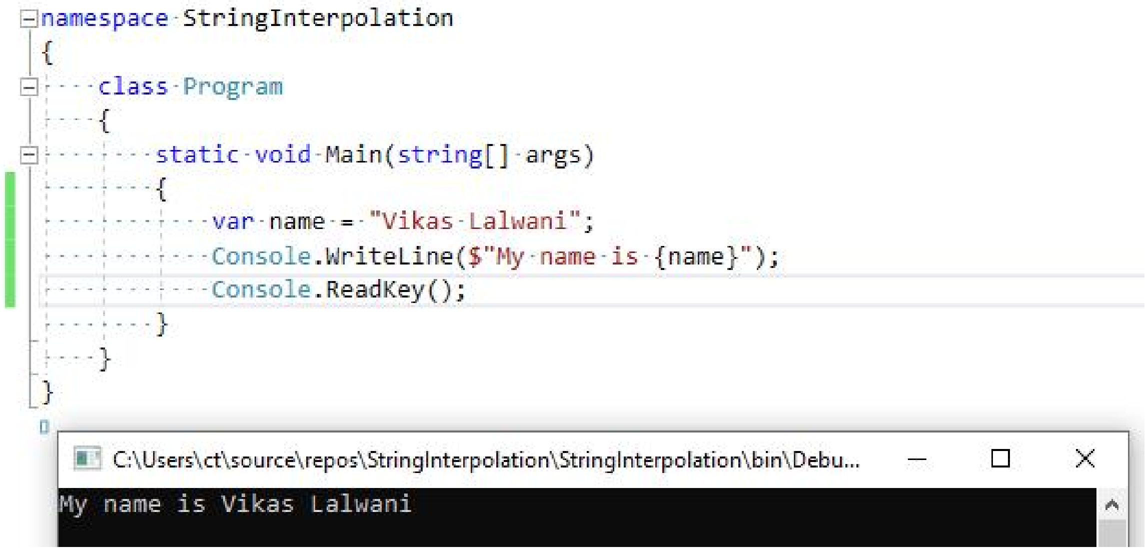
```
string firstName = "John ";
string lastName = "Doe";
string name = string.Concat(firstName, lastName);
Console.WriteLine(name);
```



Interpolation

-- Another option of string concatenation, is **string interpolation**, which substitutes values of variables into placeholders in a string. Note that you do not have to worry about spaces, like with concatenation:

```
string firstName = "John ";
string lastName = "Doe";
string name =  $"My full name is: {firstName} {lastName}";
Console.WriteLine(name);
```



```
namespace StringInterpolation
{
    class Program
    {
        static void Main(string[] args)
        {
            var name = "Vikas Lalwani";
            Console.WriteLine($"My name is {name}");
            Console.ReadKey();
        }
    }
}
```

C:\Users\ct\source\repos\StringInterpolation\StringInterpolation\bin\Debu...

```
My name is Vikas Lalwani
```

Access Strings

-- You can **access** the characters in a string by referring to its index number inside square brackets [].

Example-1:

```
    string myString = "Hello ";
    Console.WriteLine(myString[0]);  // Outputs "H"
```

Example-2:

```
    string myString = "Hello ";
    Console.WriteLine(myString[1]);  // Outputs "e"
```

Access Strings

Atlernative Method:

```
// Full name
string name = "John Doe";

// Location of the letter D
int charPos = name.IndexOf("D");

// Get last name
string lastName = name.Substring(charPos);

// Print the result
Console.WriteLine(lastName);
```

Special Characters

-- Because strings must be written within quotes, C# will misunderstand this string, and generate an error:

```
string txt = "We are the so-called "Vikings" from the north.";
```

The solution to avoid this problem, is to use the **backslash escape character.**

Special Characters

The backslash (\) escape character turns special characters into string characters:

Example-1:

```
string txt = "We are the so-called \"Vickings\" from the north.";
```

Example-2:

```
string txt = "It\'s alright." ;
```

Example-3:

```
string txt = "The character \\ is called backslash.;
```

Escape Character	Result	Description
\’	‘	Single quote
\”	“	Double quote
\\	\	Backlash

Useful Resource

Learn
C# Programming
For Loop

Learn
SQL Server
Database

Learn
ASP.Net Core

Learn
Cyber Security

Contact Me

Phone	Website	Mail	Facebook
+8801321869515	www.mrhridoy.me	mrhridoy.me@gmail.com	Mahinur Rahaman Hridoy

Thank You