String Handling Lab

Lab 1: Initializing Strings

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "Helmet";
string desc = new("A large size gray helmet");
string astericks = new('*', 40);

Console.WriteLine(name);
Console.WriteLine(desc);
Console.WriteLine(astericks);
```

Try It Out

Run the application and view the output.

Lab 2: Immutability

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name1 = "10 speed ";
string name2 = "bicycle";

// Performing concatentation creates
// a new area in memory for 'name2' and the old
// memory space occupied by 'name1'
// is marked to be released
name1 = name1 + name2;

Console.WriteLine(name1);
```

Try It Out

Run the application and view the output.

Lab 3: String Properties

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle";

Console.WriteLine(name.Length);
```

Try It Out

Run the application and view the output.

Index a String

Add the following code at the end of the **Program.cs** file.

```
Console.WriteLine(name[3]);
```

Try It Out

Run the application and view the output.

Use the Range Operator

Add the following code at the end of the **Program.cs** file to use the **Range** operator.

```
Console.WriteLine(name[3..8]);
```

Try It Out

Run the application and view the output.

Lab 4: Compare Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle";

int result = String.Compare(name, "10 speed bicycle");

// The two parameters will sort in the same position
Console.WriteLine(result);  // Result = 0
```

Try It Out

Run the application and view the output.

Compare with One Less Character

Remove the 'e' from the end of 'bicycle'.

```
string name = "10 speed bicycle";

int result = String.Compare(name, "10 speed bicycl");

// 'name' follows 2nd parameter when sorting
Console.WriteLine(result);  // Result = 1
```

Try It Out

Run the application and view the output.

Compare with More Characters

Change the value in the Compare() method to the string in bold.

```
string name = "10 speed bicycle";

int result = String.Compare(name, "10 speed bicycle 123");

// 'name' precedes 2nd parameter when sorting
Console.WriteLine(result);  // Result = -1
```

Try It Out

Run the application and view the output.

Lab 5: Substring Method

Substring Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

0 = Where to start

7 = Total number of characters to retrieve

```
string name = "10 Speed Bicycle";

Console.WriteLine(name.Substring(0, 7));
```

Try It Out

Run the application and view the output.

Use the Range Operator

Change the code in the Console.WriteLine() to the code below to achieve the same results as above.

```
string name = "10 Speed Bicycle";

Console.WriteLine(name[..8]);
```

Try It Out

Run the application and view the output.

Use the Substring() Method

Change the code in the Console.WriteLine() to the code below to return the string 'Bicycle'.

```
string name = "10 Speed Bicycle";

Console.WriteLine(name.Substring(9));
```

Try It Out

Run the application and view the output.

Use the Range Operator

Change the code in the Console.WriteLine() to view the same string 'Bicycle'.

```
string name = "10 Speed Bicycle";

Console.WriteLine(name[9..]);
```

Try It Out

Run the application and view the output.

Lab 6: Methods to See if Something Exists in a String

Contains Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle";

bool result = name.Contains("10 speed");

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

EndsWith Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle";

bool result = name.EndsWith("cycle");

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

StartsWith Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle";

bool result = name.StartsWith("10");

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Use the IsNullOrEmpty() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "";

bool result = String.IsNullOrEmpty(name);

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Use the IsNullOrWhiteSpace() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = " ";  // THERE IS A SINGLE SPACE IN HERE

bool result = String.IsNullOrWhiteSpace(name);

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Lab 7: Methods to Locate a Position in a String

Use the IndexOf() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle";

int result = name.IndexOf("bicycle");

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Use the LastIndexOf() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 speed bicycle, 10 speed mountain bike";

int result = name.LastIndexOf("speed");

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Lab 8: Strings and Arrays

Use the Split() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 Speed Bicycle";

string[] arr = name.Split(" ");

for (int index = 0; index < arr.Length; index++) {
  Console.WriteLine(arr[index]);
}
```

Try It Out

Run the application and view the output.

Use the Join() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string[] arr = { "10", "Speed", "Bicycle" };

string name = String.Join(" ", arr);

Console.WriteLine(name);
```

Try It Out

Run the application and view the output.

Lab 8: Changing Strings

Use the Format() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "{0} speed bicycle {1}";

string result = String.Format(name, "10", "(Blue)");

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Use the ToLower() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 Speed Bicycle";

Console.WriteLine(name.ToLower());
```

Try It Out

Run the application and view the output.

Use the ToUpper() Method

Change the ToLower() to ToUpper().

```
string name = "10 Speed Bicycle";

Console.WriteLine(name.ToUpper());
```

Try It Out

Run the application and view the output.

Use the Trim() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "   10 Speed Bicycle     ";

Console.WriteLine(name);
Console.WriteLine(name.Length);

name = name.Trim();

Console.WriteLine();
Console.WriteLine(name);
Console.WriteLine(name.Length);
```

Try It Out

Run the application and view the output.

Use the TrimEnd() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "   10 Speed Bicycle     ";

Console.WriteLine(name);
Console.WriteLine(name.Length);

name = name.TrimEnd();

Console.WriteLine();
Console.WriteLine(name);
Console.WriteLine(name.Length);
```

Try It Out

Run the application and view the output.

Use the TrimStart() Method

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "   10 Speed Bicycle     ";

Console.WriteLine(name);
Console.WriteLine(name.Length);

name = name.TrimStart();

Console.WriteLine();
Console.WriteLine(name);
Console.WriteLine(name.Length);
```

Try It Out

Run the application and view the output.

Lab 9: Escape Sequences

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string name = "10 Speed Bicycle \"Blue\"";

Console.WriteLine(name);
```

Try It Out

Run the application and view the output.

Use the \n Escape Sequence

Modify the ***name*** variable

```
string name = "10 Speed Bicycle \nBlue";
```

Try It Out

Run the application and view the output.

Use the \t Escape Sequence

Modify the ***name*** variable

```
string name = "10\tSpeed\tBicycle\t\"Blue\"";
```

Try It Out

Run the application and view the output.

Use the \\ Escape Sequence

Modify the ***name*** variable so it looks like a path and file name.

```
string name = "D:\\Samples\\Test.txt";
```

Try It Out

Run the application and view the output.

Lab 10: Verbatim String Literals

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string fileName = @"D:\Samples\Test.txt";

Console.WriteLine(fileName);

string value = @"D:\nTest\t";

Console.WriteLine(value);
```

Try It Out

Run the application and view the output.

Multi-Line String Literal

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
string desc = @"This 10 speed bicycle has excellent
  qualities and was built for the road. Come try this
    fantastic 10 speed bike today.";

Console.WriteLine(desc);
```

Try It Out

Run the application and view the output.

Lab 11: String Interpolation

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
int id = 1;
string name = "Bicycle";

string result = $"ID = {id}, Name='{name}'";

Console.WriteLine(result);
```

Try It Out

Run the application and view the output.

Lab 12: StringBuilder Class

Open the **Program.cs** file and replace the entire contents of the file with the following code.

```
using System.Text;

StringBuilder sb = new(1024);

sb.Append("Helmet");
sb.AppendLine(" - 1");

sb.AppendFormat("{0} - {1}", "10 Speed Bicycle", 2);

Console.WriteLine(sb.ToString());
```

Try It Out

Run the application and view the output.