< **Previous**   Unit 3 of 8 ⌄   **Next** >

✓ 100 XP

Exercise - Combine strings using string concatenation

10 minutes

Often, you'll need to combine data from many different sources, including literal strings and variables containing both text and numeric data. In this unit, you'll use string concatenation to combine two or more values into a new string.

What is string concatenation?

String concatenation is "programmer speak" for simply combining two or more `string` values into a new `string` value. Unlike addition, the second value is appended to the end of the first value, and so on. In the following exercise, you'll write code to concatenate `string` values together.

Concatenate a literal string and a variable

To concatenate two strings together, you use the *string concatenation operator*, which is the plus symbol `+`.

1. Select all of the code in the .NET Editor, and press `Delete` or `Backspace` to delete it.

2. Enter the following code in the code editor:

```C#
string firstName = "Bob";
string message = "Hello " + firstName;
Console.WriteLine(message);
```

3. Now, run the code. You'll see the following result in the output console:

Output

```
Hello Bob
```

Notice the order—the first string `"Hello "` is first in the new string, and the value in the `firstName` variable is appended to the end of it.

Concatenate multiple variables and literal strings

You can perform several concatenation operations in the same line of code.

1. Modify the code you wrote earlier to the following:

   C#

   ```csharp
   string firstName = "Bob";
   string greeting = "Hello";
   string message = greeting + " " + firstName + "!";
   Console.WriteLine(message);
   ```

   Here you create a more complex message by combining several variables and literal strings.

2. Now, run the code. You'll see the following result in the output console:

   Output

   ```
   Hello Bob!
   ```

Avoiding intermediate variables

In the previous steps, you used an extra variable to hold the new string that resulted from the concatenation operation. Unless you have a good reason to do so, you can (and should) avoid using intermediate variables by performing the concatenation operation as you need it.

1. Modify the code you wrote earlier to the following:

   C#

   ```csharp
   string firstName = "Bob";
   string greeting = "Hello";
   Console.WriteLine(greeting + " " + firstName + "!");
   ```

2. Now, run the code. The result in the output console should be the same even if you simplified the code:

```
Output

Hello Bob!
```

Recap

Here's what you've learned about string concatenation so far:

- String concatenation allows you to combine smaller literal and variable strings into a single string.
- Avoid creating intermediate variables if adding them doesn't increase readability.

Module complete:

Unlock achievement

.NET Editor

Press CTRL + M , TAB to exit the editor

🗑Clear   ▷Run   ⓘ

1

Output   ☺