# Exercise: Functional Programming

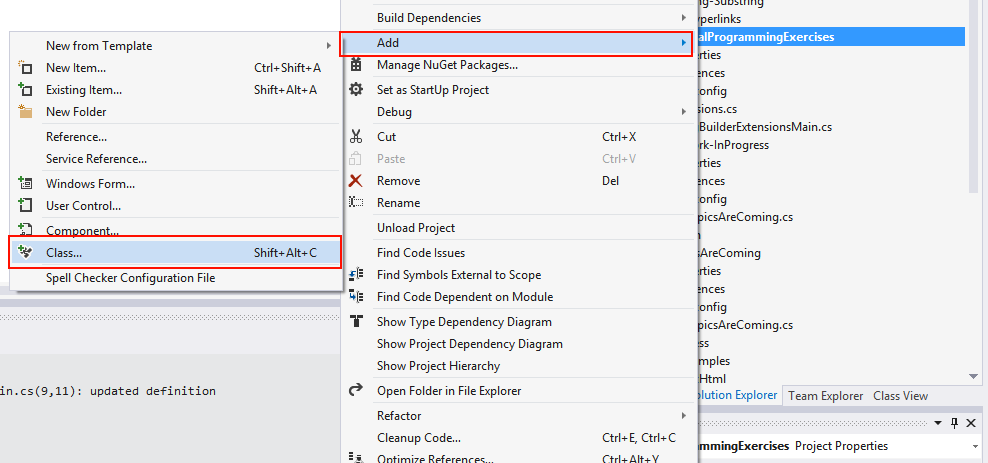
This document defines in-class exercise problems from the [“Advanced C#“ Course @ Software University](http://softuni.bg/courses/advanced-csharp/). You are presented with some problems and certain steps you need to take in order to accomplish the tasks.

## Problem 1. Extend the StringBuilder Class

The StringBuilder class has many useful methods, however, it doesn’t contain a Substring method. That’s a shame. But you can write your own.

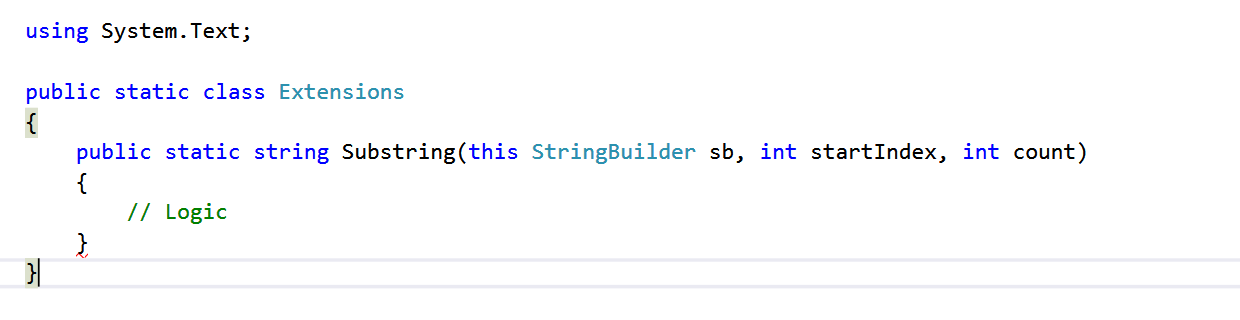
If you don’t remember all the steps, we’ve outlined them for you.

### Step 1. Create a public static class

You need to create a new class to hold the extension method in addition to the class holding you Main method. Right-click on your project and select **Add -> Class**. Choose a meaningful name, like Extensions, or StringBuilderExtensions.Remove all unnecessary using directives and make the class **public** and **static** (just add the "public" and "static" keywords in front of the class name).

### Step 2. Create the Extension Method

Within the class, create a **public static** method called Substring. As a first parameter, you should specify the type you want to extend, preceded by the keyword "this". Then come the other parameters – the stat index and count of symbols. The signature of the method should look like this:



### Step 3. Implement the Method

You need to create a string which takes **count** characters from the StringBuilder starting from **startIndex**. Remember to return the result as a string. We leave the implementation details to you.

### Step 4. Test the Method

In your main program, create some StringBuilder variables and call your Substring extension method. Extension methods are static, but are called like instance methods. Here is a sample code you may execute: