## String class & the Online Help

## The goal for this exercise is to learn how to look up existing classes using an online help tool, and to be able to use methods built into the string class to manipulate text.

## Let's say that you're looking for more information about a certain topic within Visual Studio, such as the String class (which allows us to quickly & easily deal with text in C#). You've got at least a couple of options:

1. Looking in Visual Studio's dedicated, local help application
2. Looking online, at MSDN.microsoft.com
3. Looking online generally, using a search engine

## Let's look at each of these:

**Looking in Visual Studio's dedicated, local help application**

## You can get to the Help feature via the Help menu item (all the way at the right of the menu items). Once there, you've got a number of options for help – if you know what you're looking for, the Index is a good option:



Once you select that item (or press Control and F1 at the same time, followed by I), a separate window will open. It's left-most window will look something like this:



You should select the TextBox under "Look for:", then type in String. Once you've done that, click on the "strings [C#}" option (you may need to adjust the **Filtered by:** drop-down menu – **Visual C# Express Edition** works, as should the **(unfiltered)** option).



You will then be shown (in the right-hand panel) a page that gives an overview of the String class in C#. This page explains a lot about the String class – don't worry about most of it, but if you're curious, do read through it. Towards the top there's a link to the System.String page, which the list of Properties and Methods that the String method has.



From there there's also a link ("For a list of all members of this type, see [String Members](http://msdn.microsoft.com/en-us/library/system.string_members%28VS.80%29.aspx).") that will describe methods of the String class.

**Looking online, at MSDN.microsoft.com**

Another good option is to look online for help. While using a general web search engine (Google, Bing, etc) can work well, Microsoft has a site dedicated specifically to providing information about just its products. It's a great reference if want to look up (say) methods that the String class provides. You can either go directly to it (<http://msdn.microsoft.com/en-us/default.aspx>), or you can search for it using a search engine. Once you're there, type ".Net string" or "C# String" into the search box & you should find several useful links amongst the top several. One thing to keep in mind is that MSDN provides information about all of Microsoft's languages, so it's good to limit the search to either C# or the .Net platform (just so you don't get stuff about Visual Basic, F#, etc)

**Looking online generally, using a search engine**

You can, of course, search the web in general. This tends to be more useful for tracking down specific problems that you're having, and sometimes leads to useful 'overview' pages. Adding the term "tutorial" to whatever you're looking for tends to give good 'overview' type results (for example, searching for "C# string tutorial"), but your mileage may vary.

 **What you need to do for this exercise:**

1. You should have read everything above, understood everything above, and tried out all the code, on your own, so that you’re clear about how it works. You are also encouraged (as you always are) to ‘play around’ with this, as well – try stuff that isn’t specifically called for above, just to test your knowledge, and just to see if you can.
2. In addition, in the starter project that has been provided to you, you should create a method (on a new class that you will name **StringBasics**) that asks the user to type in some text (unless it has already been provided to you). Store a reference to that text using a **string** variable – for Console apps, this would look like:

string input;
Console.WriteLine("Type in some text!");
input = Console.ReadLine();
3. Then, print an uppercase version of the string using the **ToUpper** method.
4. Also, print a lowercased version using the **ToLower** method.