

CSCI 1301 - Lab 05

Clément Aubert

January 29, 2018

Deadlines: This lab needs to be completed before taking Lab 06.

Dependencies: This lab can be done in almost any order, but the recommended order is of course the one below.

Shortcut of the day: Open the source code of a project, and hit CTRL + k, then CTRL + d. What happened? If nothing changed (which is, from a certain perspective, a good sign), then try to find the answer at <http://visualstudioshortcuts.com/2015/>. Keep this handy cheatsheet bookmarked!

Part 0 - Get Your Quiz Back

If time allows, I'll grade your quiz and make personal feedback during the lab. If not, you'll have it back next Tuesday, and are encouraged to meet with me if something isn't clear.

Part I - Read From the User

- a. Download the following project: PersonalizedWelcomeMessage.zip.
- b. Extract it, and open it in VS.
- c. Compile and execute it.
- d. You will be prompted with the message

Please, enter your first name, followed by "Enter":

Enter your first name, followed by Enter ☐. You just witnessed an interaction between a program and an user!

- e. Read the source code, and make sure you understand all of it.
- f. Change the code, so that the program would also ask the user's last name, and print both their first and last name.

Part II - Variables Types: From String to Integer

- a. Create a new project
- b. Write two statements, one that declares a variable of type `int` named `intVar`, one that declares a variable of type `string` named `stringVar`.
- c. Assign the value 3 to `intVar`, and "4" to `stringVar`.
- d. Print the values of `intVar` and `stringVar`.
- e. Write a statement that assign the value of `stringVar` to `intVar`. Why is the compiler complaining?
- f. Copy the following statement, to "convert" the string value of `stringVar` into an integer value, and assign it to `intVar`:

```
intVar = int.Parse(stringVar);
```

- g. Using <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/types/how-to-convert-a-string-to-a-number>, try to understand what just happened.
- h. Change the value of `stringVar` to be " 6" (note the spaces) and to assign it to `intVar` using `int.Parse` as previously shown. Does it work like you expected?
- i. Re-do the previous item three times, with the values "6 ", " 6 ", and "6 2", instead of " 6".

Part III (Optional) - Pushing Further

The following are two independent tasks, to widen your understanding of this class, and to prepare you for the next labs.

- a. In Lab 04's "Pushing Further", part b., you saw how one could remove the `using System;` statement, provided you used `System.Console.WriteLine` instead of just `Console.WriteLine`. To understand this behavior, have a look at <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/namespaces/index>
- b. Looking back at Part II, note that we can also go from `int` to `string`. Use a statement like

```
stringVar = intVar.ToString();
```

to "convert" the integer value of `intVar` into a string, and assign it to `stringVar`. You're not supposed to understand all the details of the explanation at https://docs.microsoft.com/en-us/dotnet/api/system.int32.tostring?view=netframework-4.7.1#System_Int32_ToString, but looking at it could be an interesting read.