USING VISUAL STUDIO FOR PYTHON AND C

Dr. Siamak Sarmady (Urmia University of Technology)

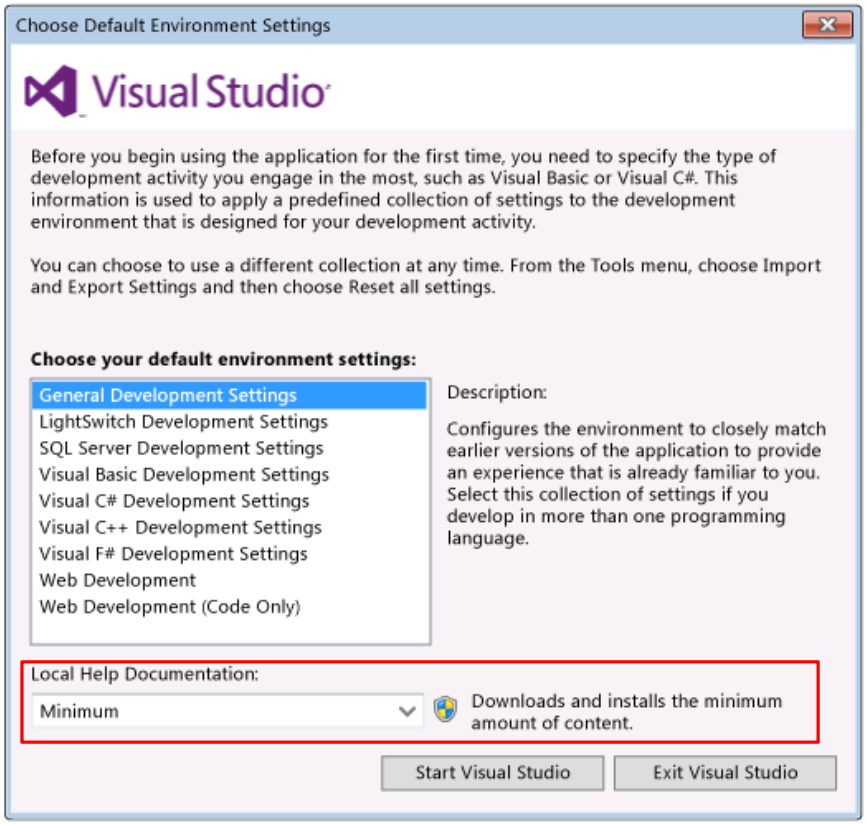
Installing Visual Studio

Installing Visual Studio

- Run the Visual studio installer and remove all the options. We don't need any of these to run Python and C/C++ codes.

Launching for the first time

- After install has been finished, launch Visual Studio.
- Change the combo box named "Local Help Documentation to "None" (otherwise Visual Studio will download 1G of help files!
- The IDE will show you a list of languages and will ask you to select your default language. Select C++.



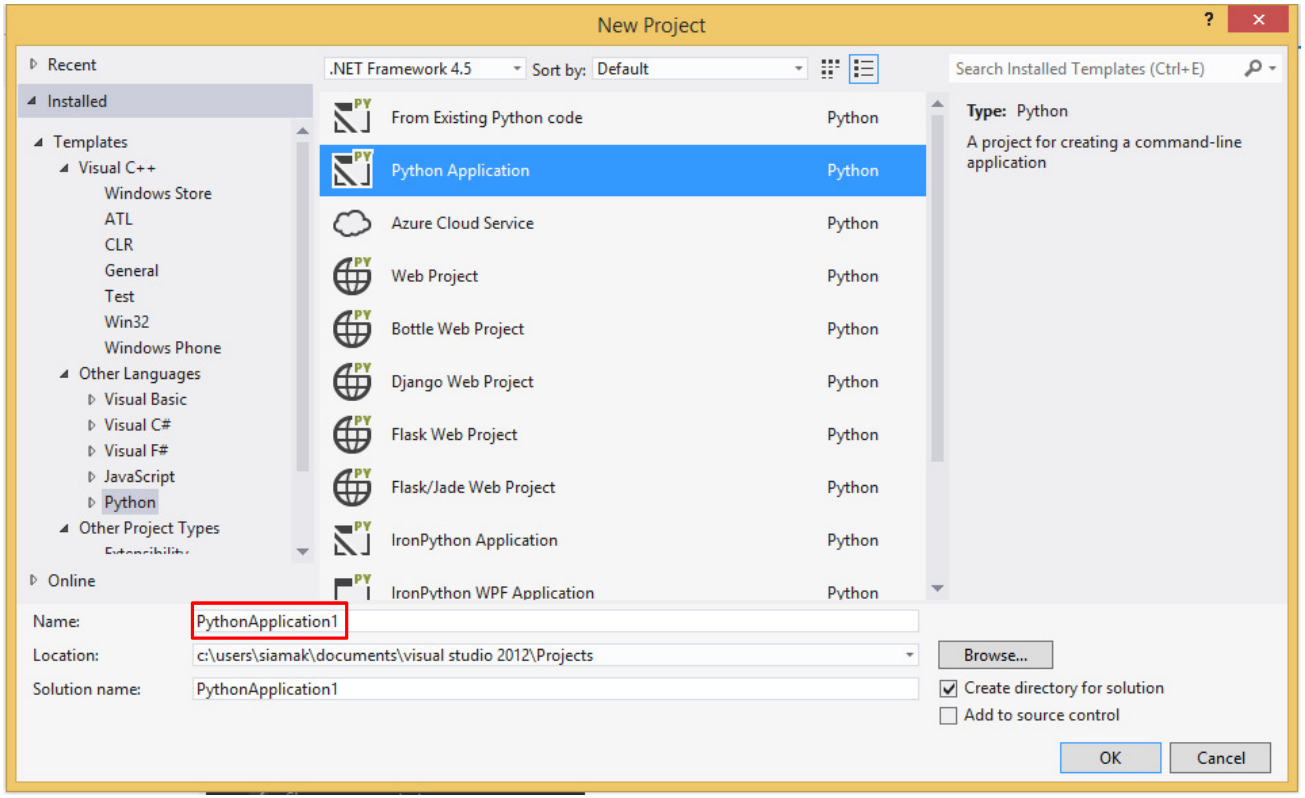
Running Python

Installing and Using Python Tools for Visual Studio (PTVS)

- Go to below address:
  https://pytools.codeplex.com/releases/view/109707
- Download "PTVS v2.1" for visual studio 2012 (the file "PTVS 2.1 VS 2012.msi")
- Note that PTVS 2.2 cannot be installed on VS2012.
- Close Visual Studio first and then install the extension.

Creating Python Projects

- Now you can create a new python project.
- Go to File -> New project -> Python and select "Python Application"
- Select a name for your python project (soloution) name
- You may also change the directory where the project is saved.
- Press Ok

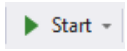


Adding the program

- As soon as the project is created a Python file with the same name as the project is also created. For example enter the following code:

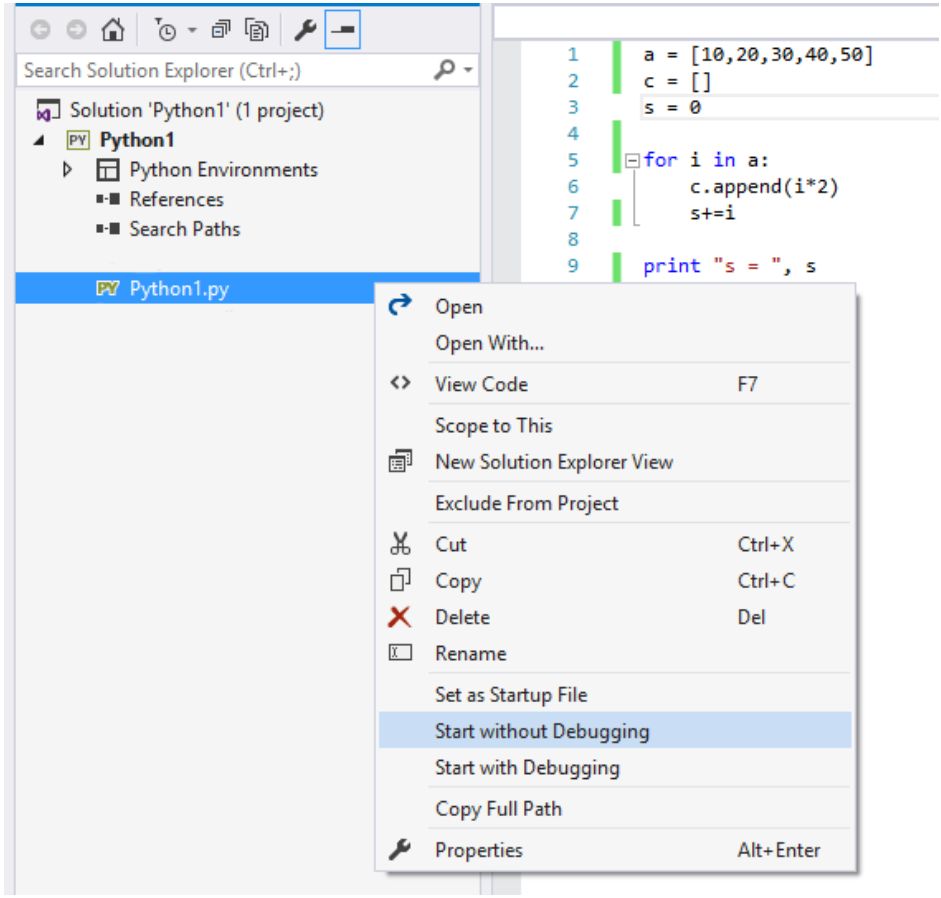
```
a = [10,20,30,40,50]
c = []
s = 0

for i in a:
    c.append(i*2)
    s+=i

print "s = ", s
print "c = ", c
```

- In order to run the project either press the "run" button on the toolbar



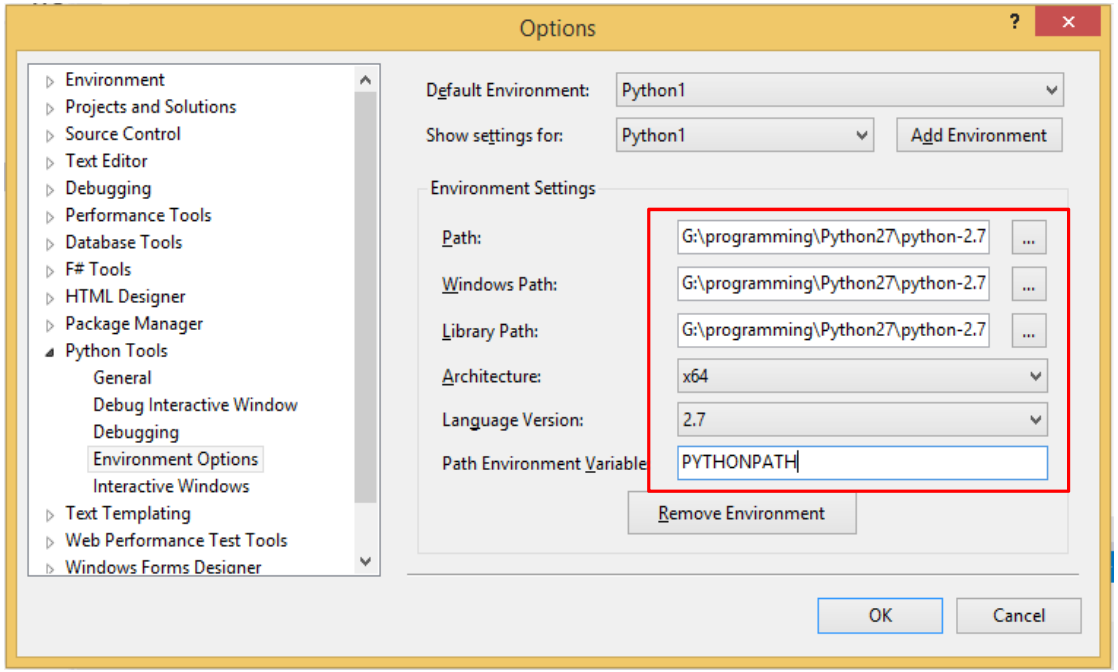
or

- Right click on the python file and select "Run Without De



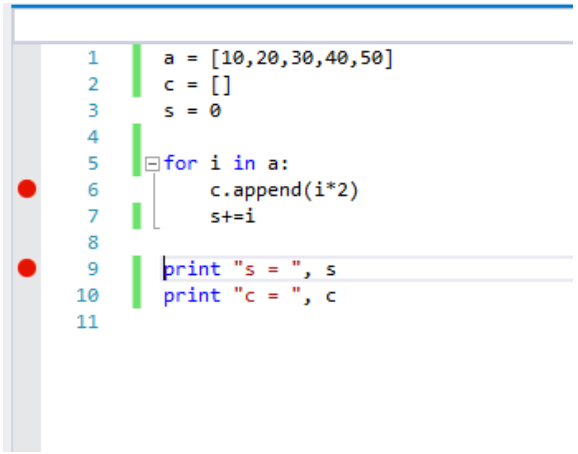
Python Configurations

- If Visual Studio cannot find your python installation, you need to configure it. Either modify the settings, or add a new environment and put the address of "python.exe" in the Path and Windows Path. In the 3rd box put your python "Lib" path.
- Go to Tools->Options->Python Tools -> Environment Settings



Debugging the Code

- In order to debug the code, place breakpoints in desired lines (click on the space before the line).

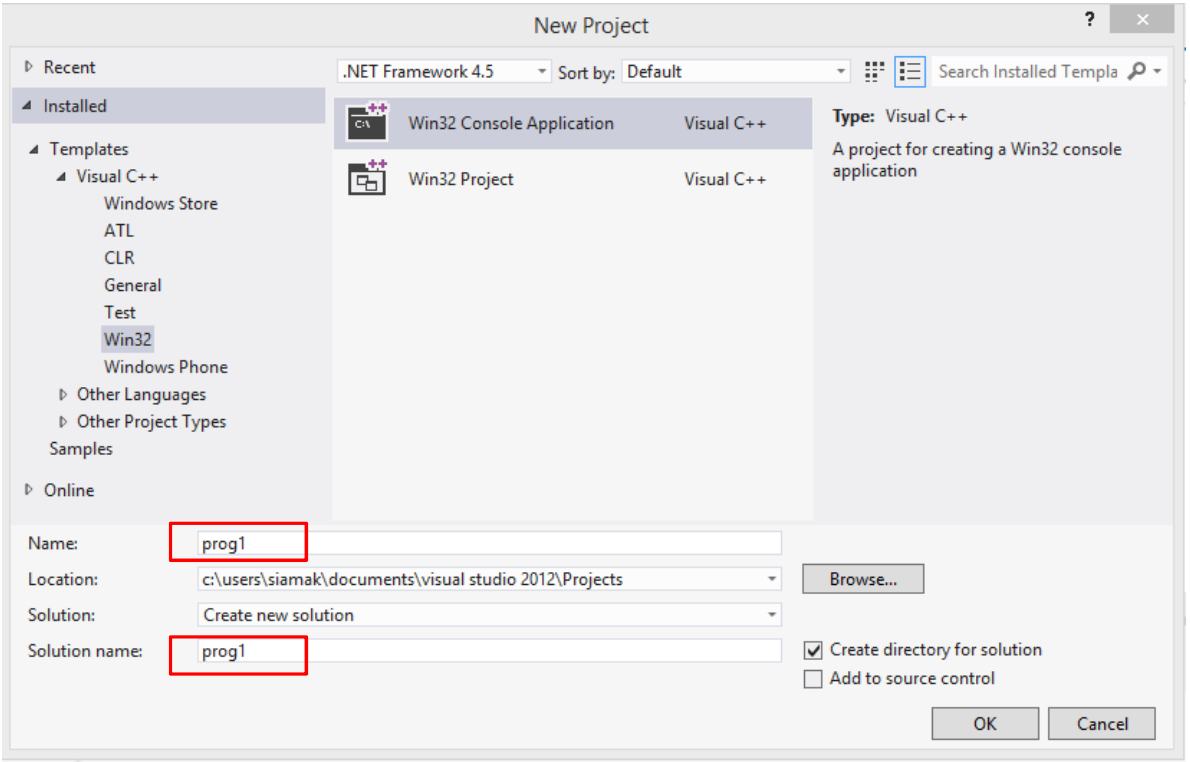


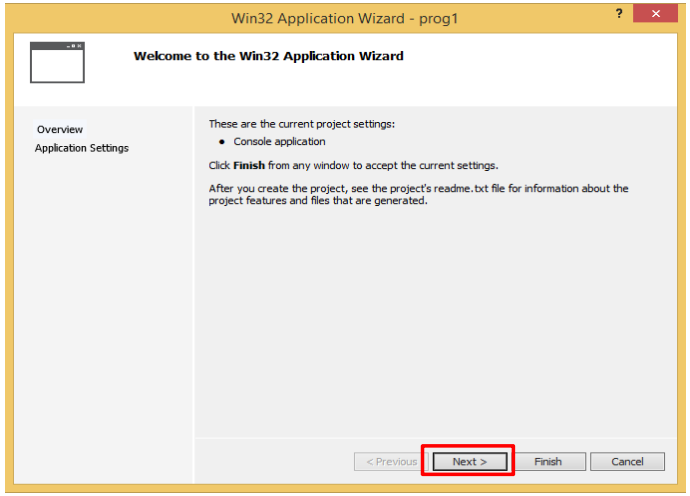
- In order to start debug process, either right click on the "file name" and select "Start with debugging" or from "Debug" menu select "Start Debug"
- Use toolbar buttons to "run step by step", "step into functions" or "step over" function names

Running C++

Creating Project

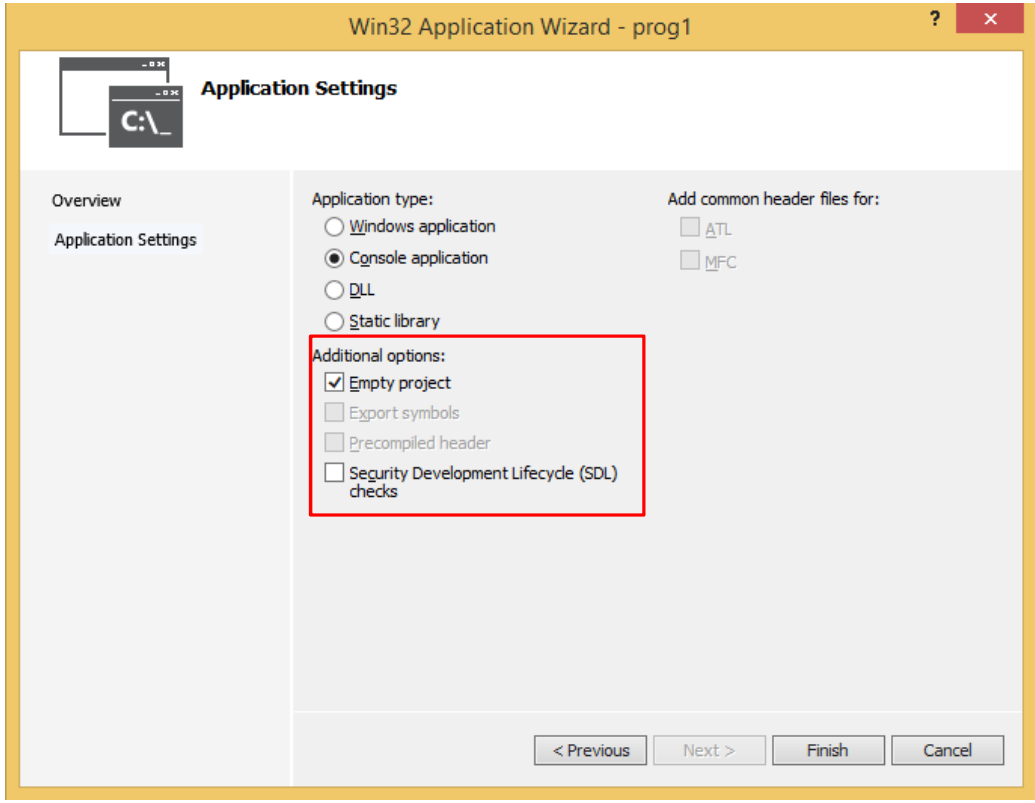
- File -> new project -> C++ -> Console Application
- Select a name (for project and solution).
- On the Wizard that appears select "next"





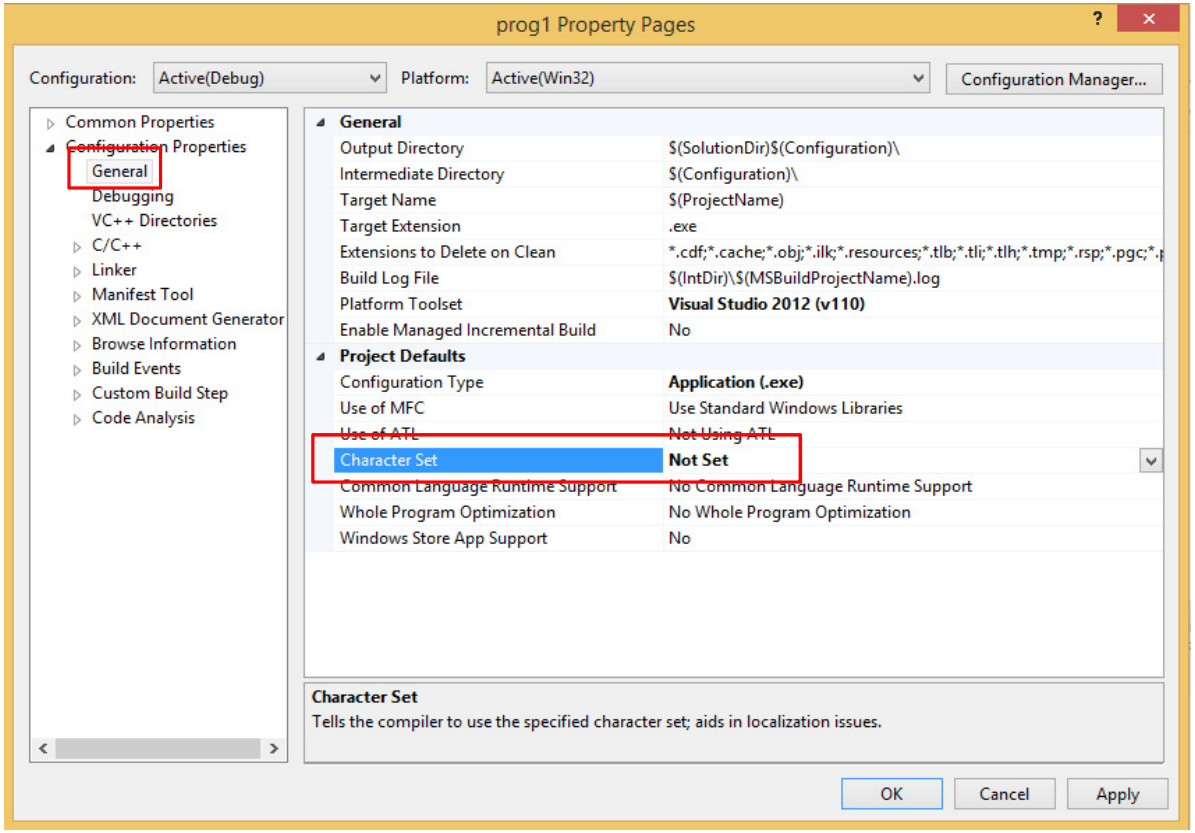
Application Setings

- In additional options, select "Empty Project"
- Make sure the settings match the shown options.



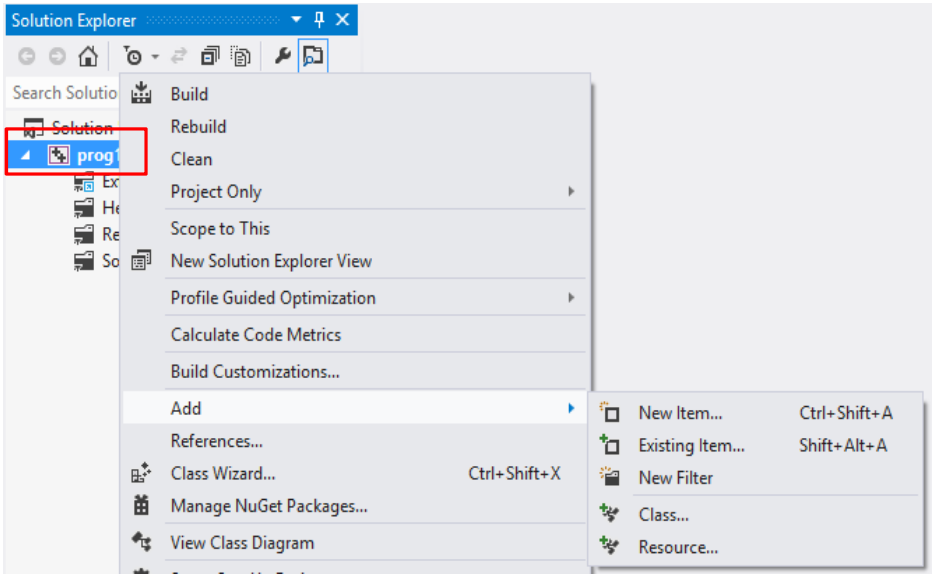
Modify Project Settings

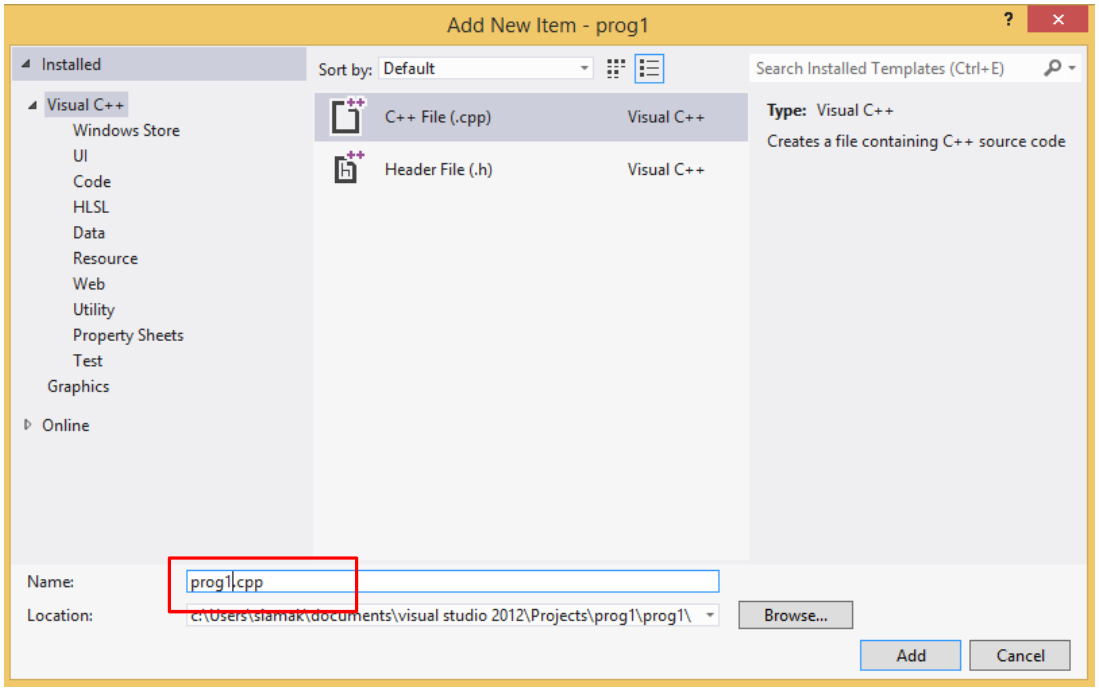
- Right Click on the project name, select properties
- In the General section, set "Character Set" to "Not Set"



Add a C file

- Right click on the project name and select add->New Item…
- Now enter a name for the C/C++ file name (keep the .cpp file extension)



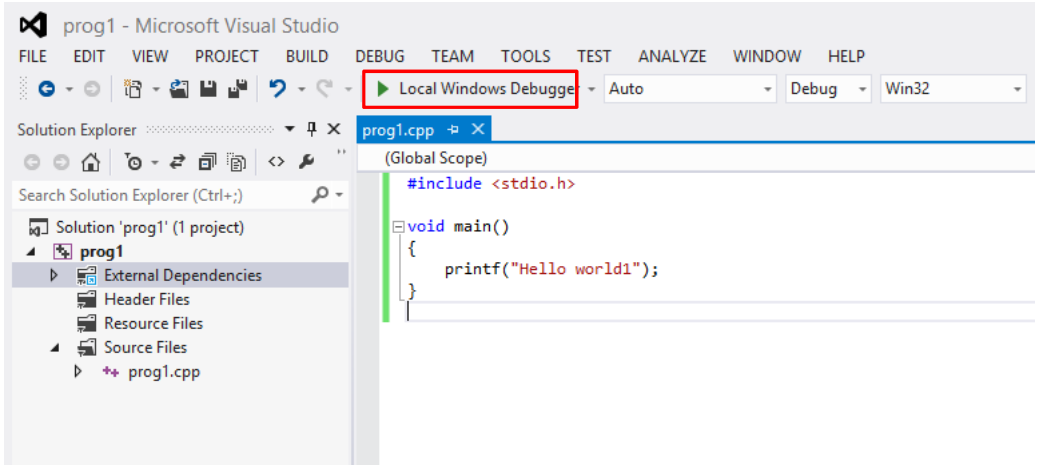


Adding the code and running the code (in debag mode)

- In the C/C++ file you created, enter the following code.

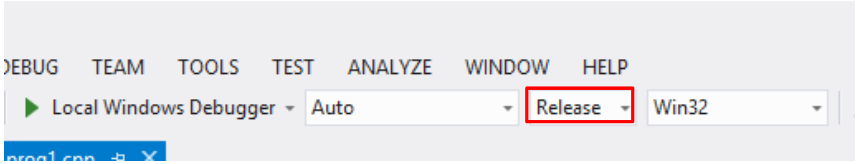
```
#include <stdio.h>

void main()
{
        printf("Hello world1");
        getchar();
}
```

- In order to run the code in debug mode, click on the "local windows debugger button"



Running the Code (in debag mode)

- After you have solved bugs of your application, change the run mode to Release. Visual studio will produce a much faster code that does not have debug checks in it.



- If your output console disappears and you want to keep the console from disappearing, from debug menu, select "Start without debug" (or Ctrl+F5)