Visual Studio Session

Youngsup Kim
idebtor@gmail.com

Contents

Before you start

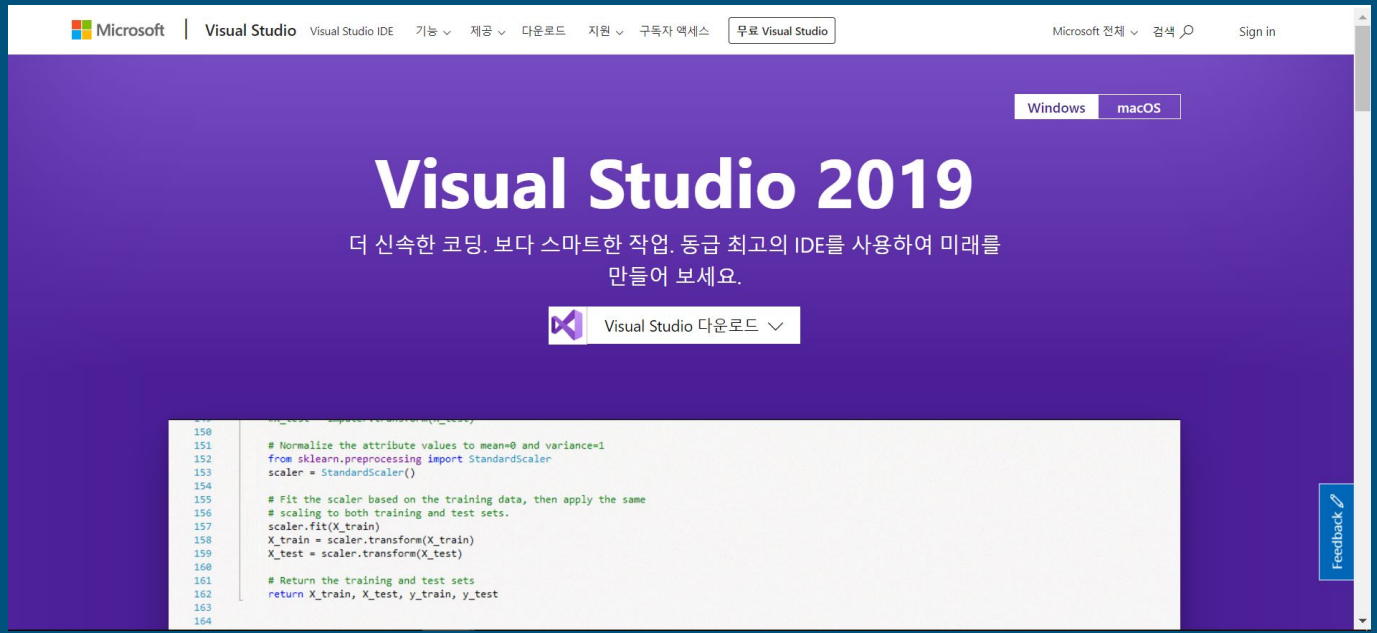
Clone this Repository!

https://github.com/SukJinKim/19-1-Visual-Studio-Session

How to install Visual Studio Professional 2019

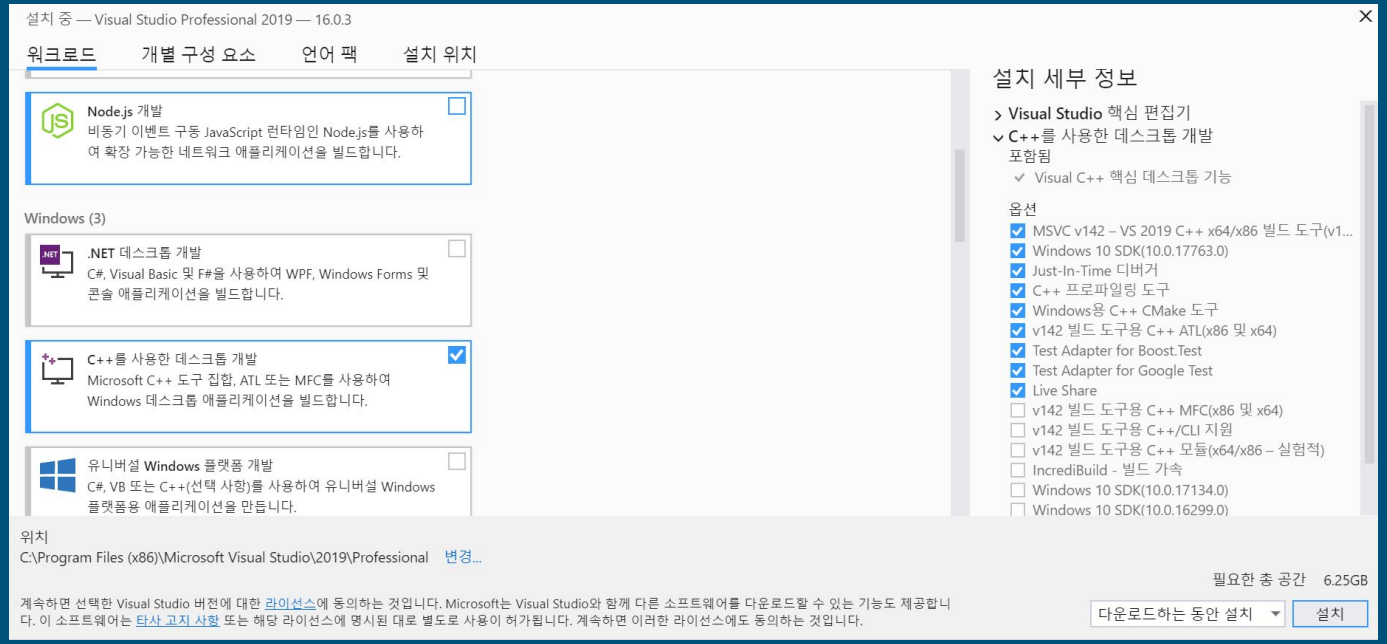
How to install Visual Studio Professional 2019

1. Download Visual Studio Professional 2019



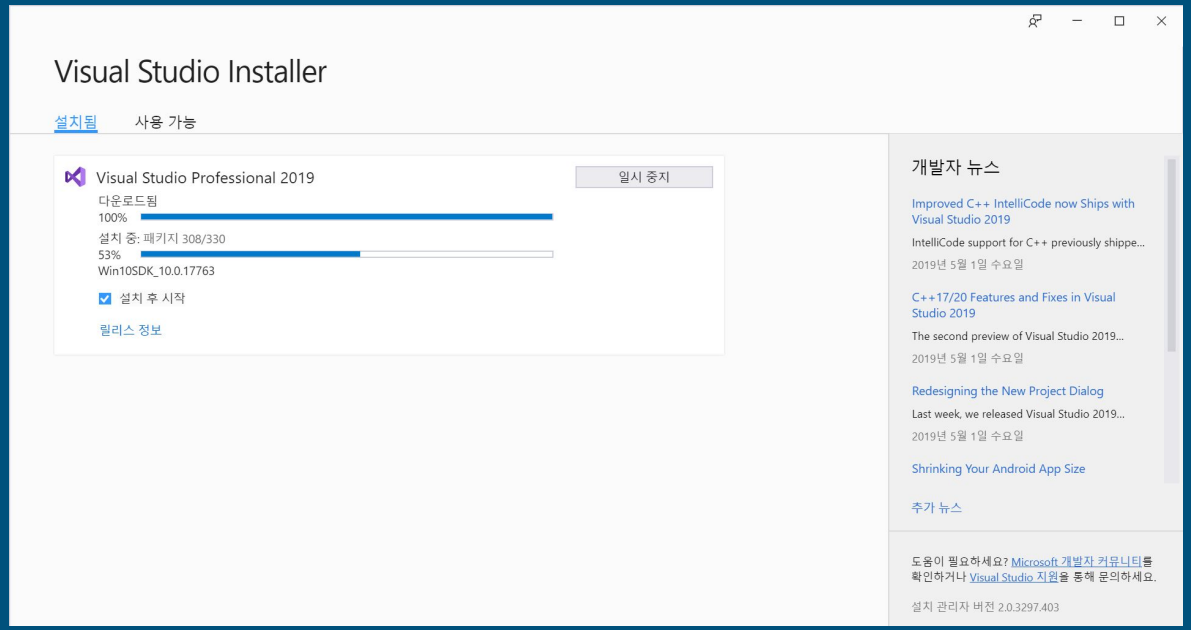
How to install Visual Studio Professional 2019

2. Check the box labeled desktop development using c++.



How to install Visual Studio Professional 2019

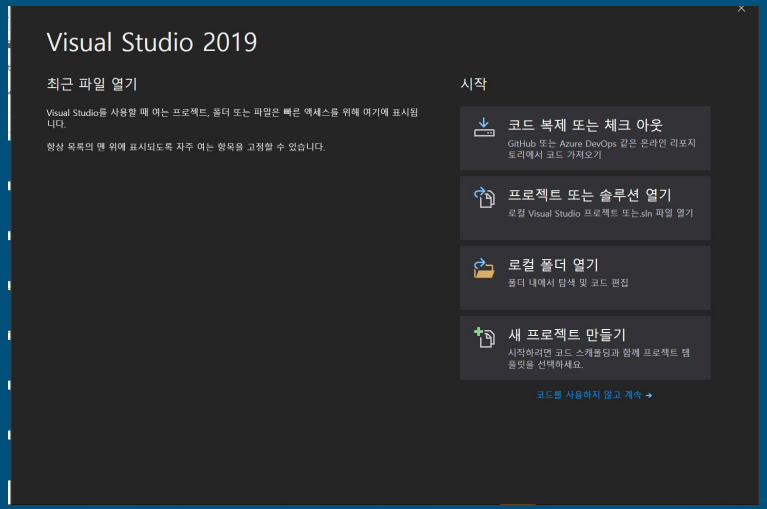
3. Wait until the installation is completed.

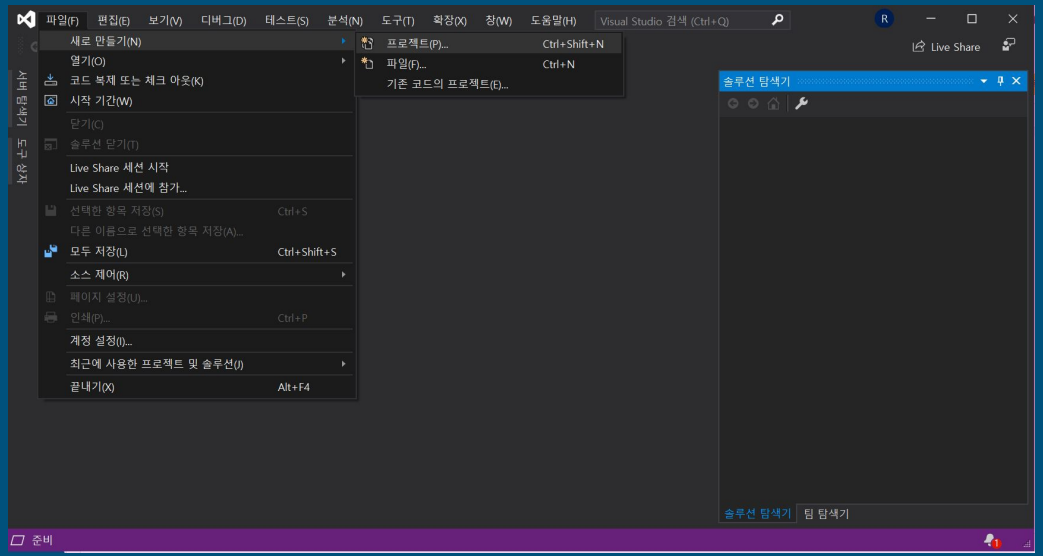


Let’s build your first c++ project and debug it!

Your first C++ console project - Create a project

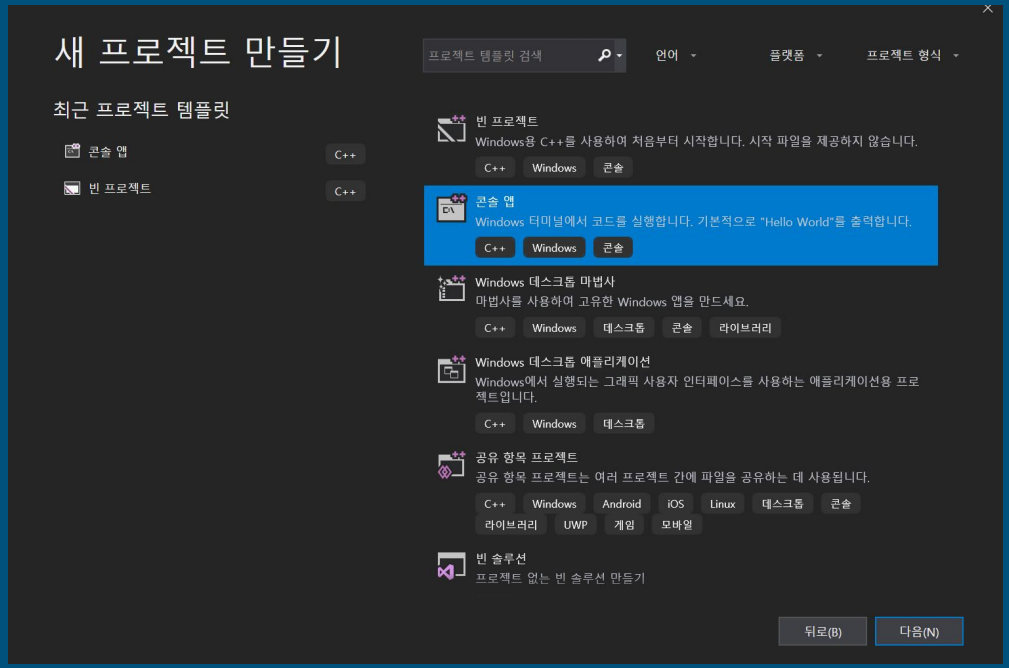
1. Choose **Create a new project** to get started. Otherwise, on the menubar in Visual Studio, choose **File > New > Project**. The Create a new project window opens.





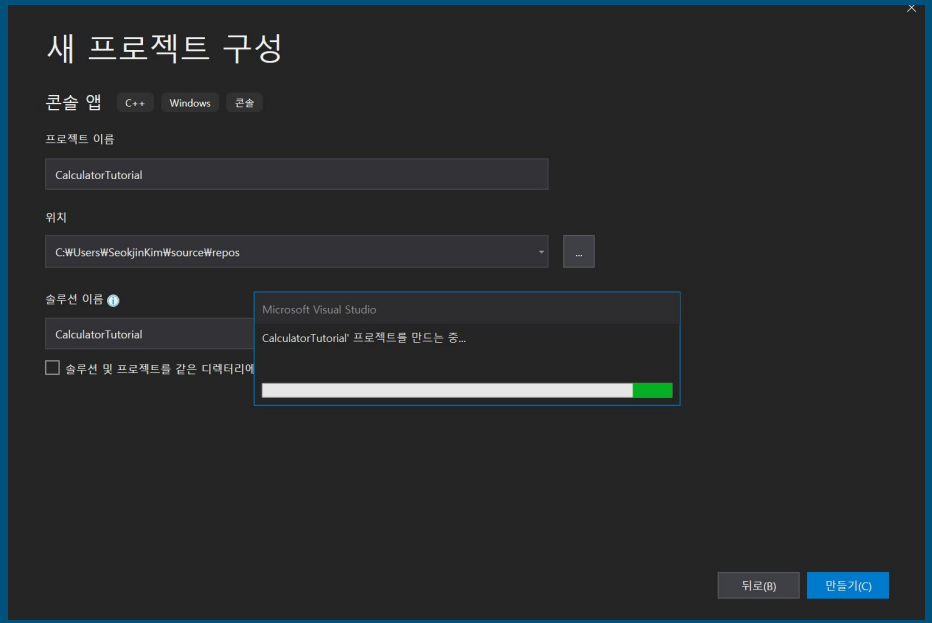
Your first C++ console project - Create a project

2. In the list of project templates, choose **Console App**, then choose Next

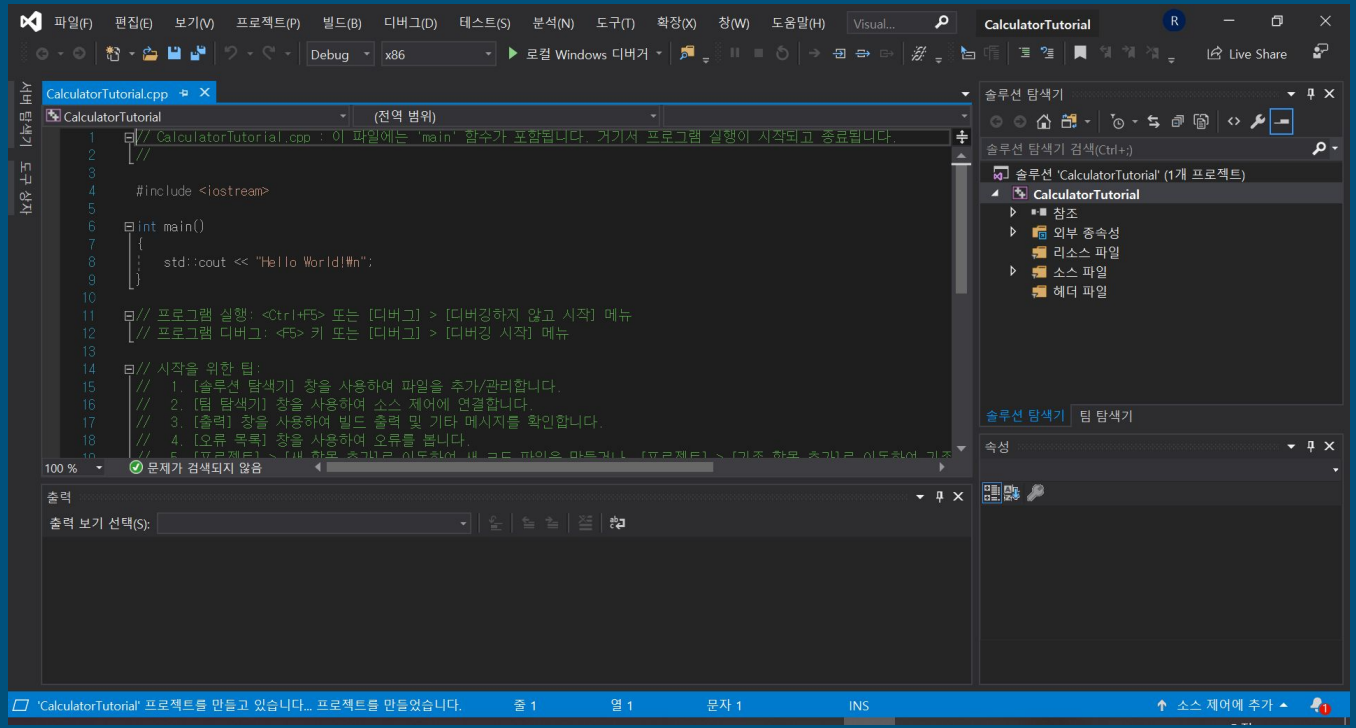


Your first C++ console project - Create a project

3. In the Configure your new project dialog box, select the Project name edit box, name your new project ***CalculatorTutorial***, then choose Create.



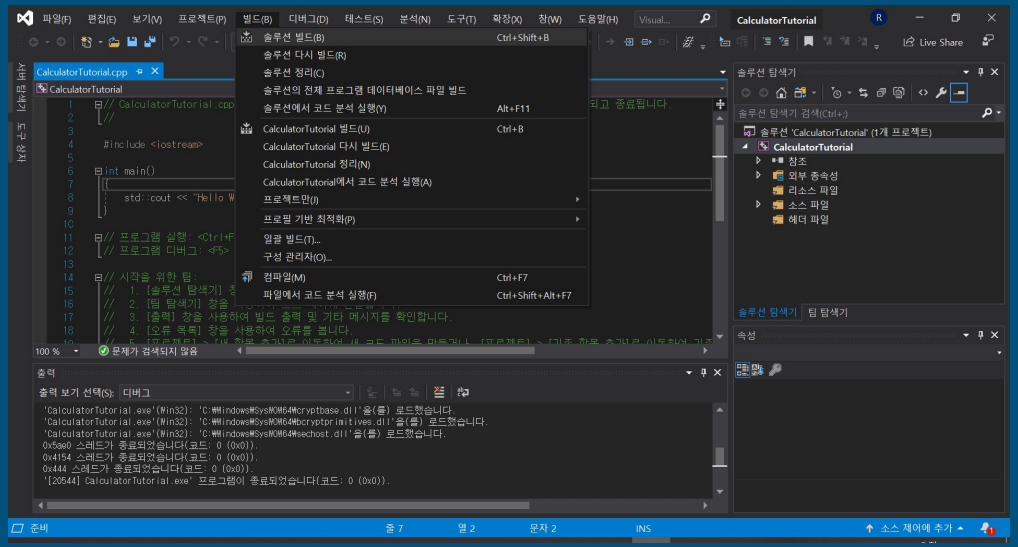
Your first C++ console project - Create a project



Your first C++ console project - build and run

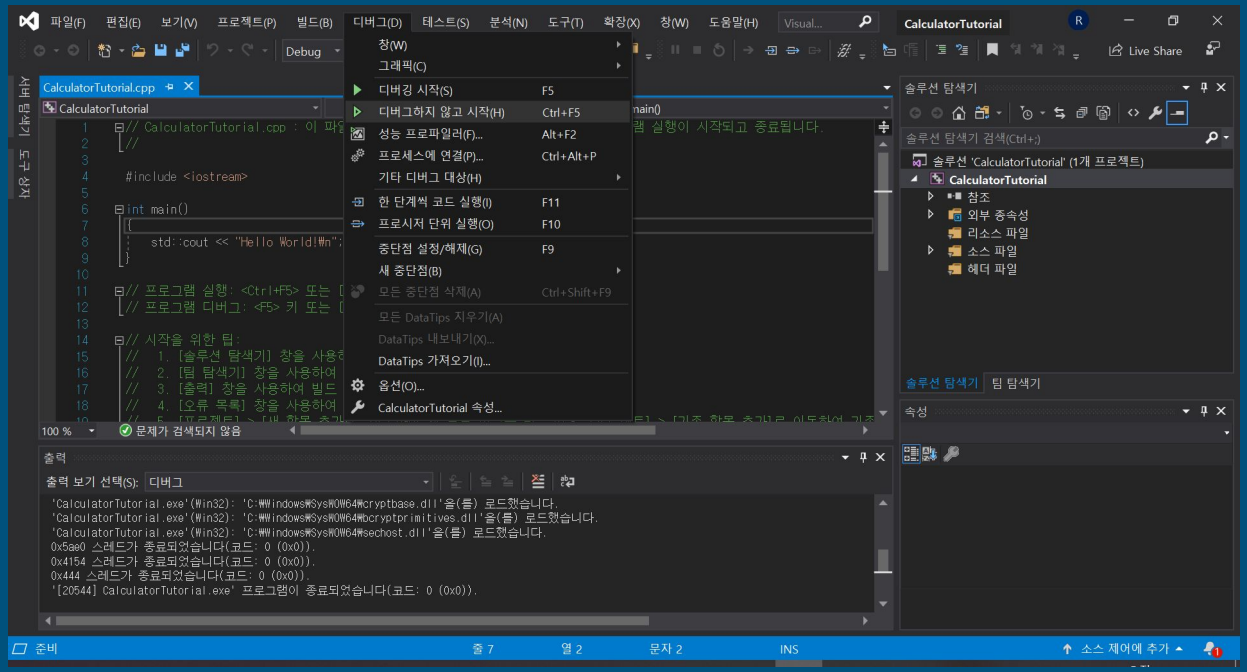
Let’s verify that your new app builds and run!

1. To build your project, choose Build Solution from the Build menu. The Output window shows the results of the build process.



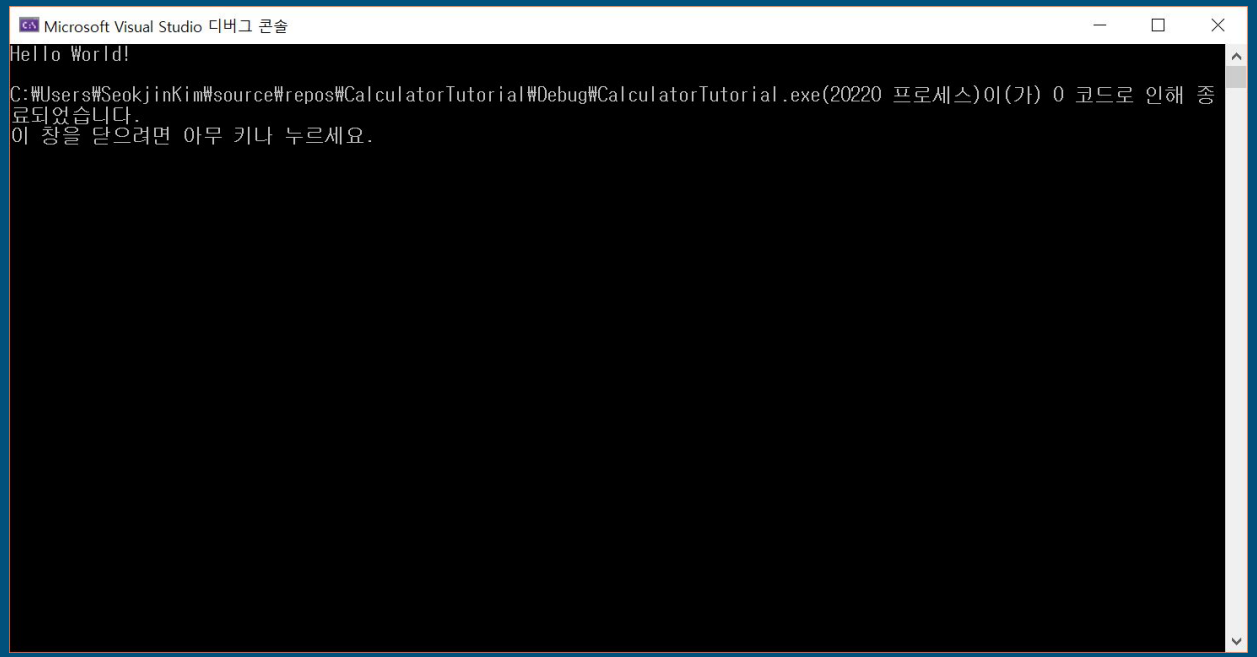
Your first C++ console project - build and run

2. To run the application, press Ctrl+F5 or go to the Debug menu and choose Start Without Debugging.



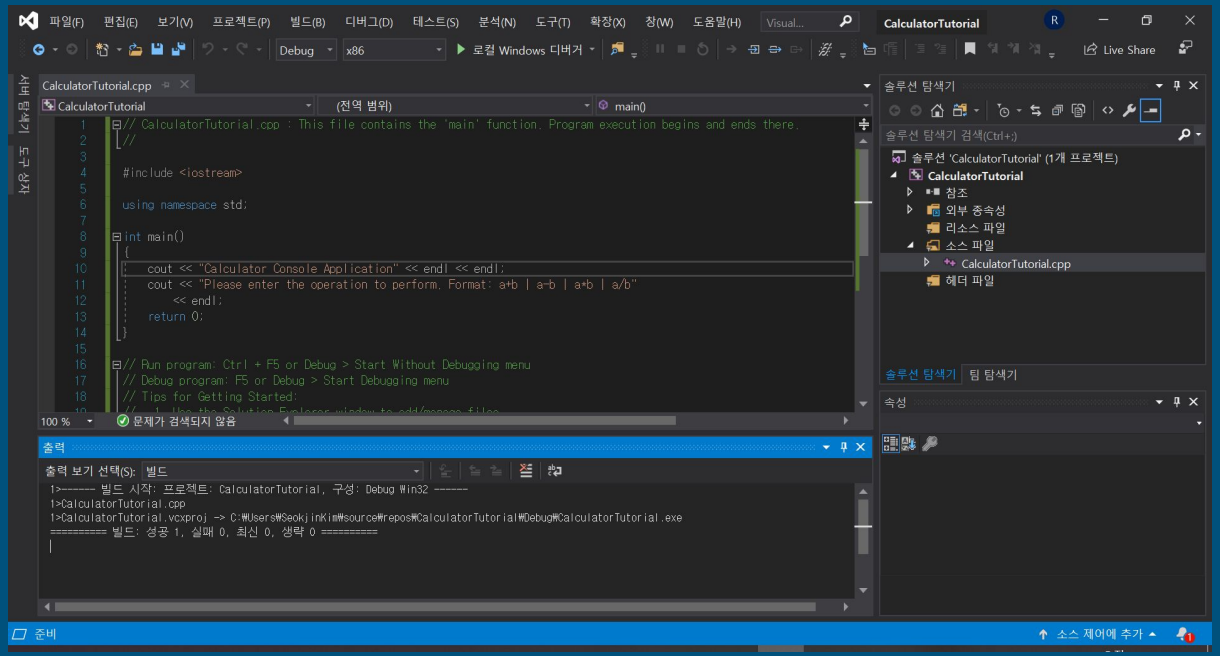
Your first C++ console project - build and run

2. To run the application, press Ctrl+F5 or go to the Debug menu and choose Start Without Debugging.



Your first C++ console project - Edit the Code

1. Edit the code to match ***CalculatorTutorial1.cpp*** in github.

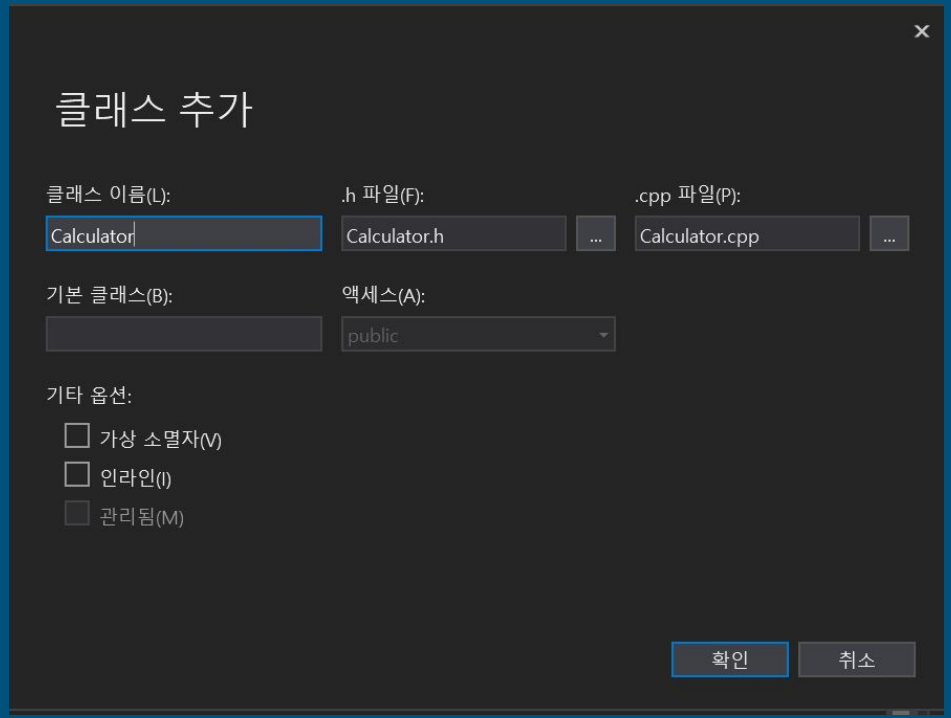


Your first C++ console project - Add code to do some math.

1. Go to the Project menu and choose Add Class. In the Class Name edit box, enter *Calculator*. Choose OK.

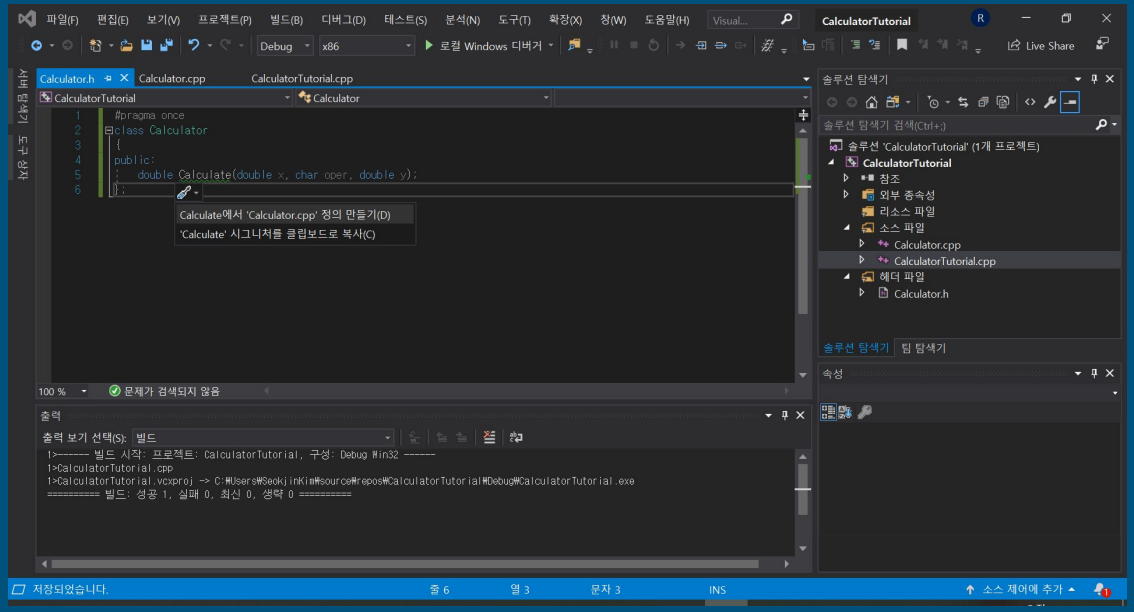
   Two new files get added to your project. To save all your changed files at once, press Ctrl+Shift+S. It's a keyboard shortcut for File > Save All. There's also a toolbar button for Save All, an icon of two floppy disks, found beside the Save button. In general, it's good practice to do Save All frequently, so you don't miss any files when you save.

Your first C++ console project - Add code to do some math.



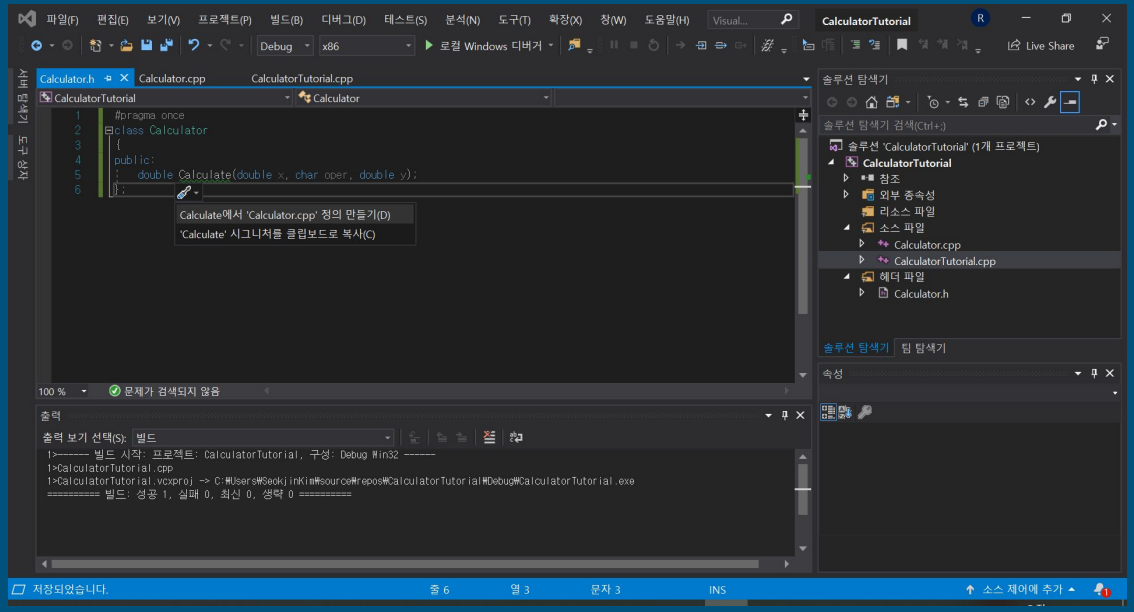
Your first C++ console project - Add code to do some math.

3. You'll see a green squiggle appear under Calculate. It's because we haven't defined the Calculate function in the .cpp file. Hover over the word, click the lightbulb (in this case, a screwdriver) that pops up, and choose Create definition of 'Calculate' in Calculator.cpp.



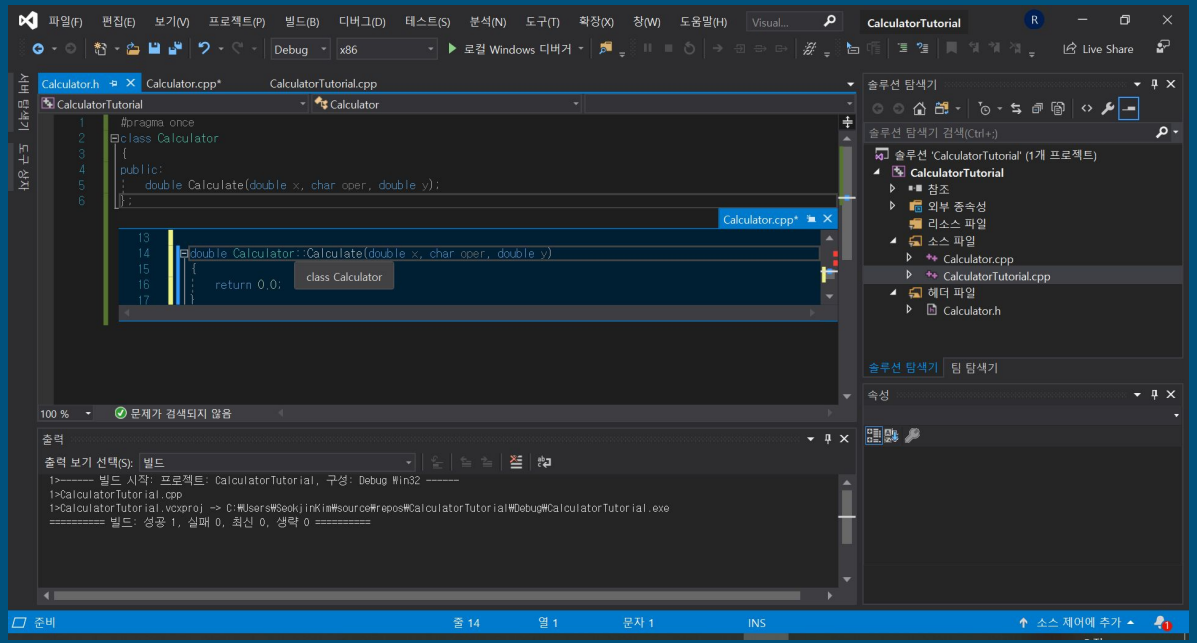
Your first C++ console project - Add code to do some math.

3. You'll see a green squiggle appear under Calculate. It's because we haven't defined the Calculate function in the .cpp file. Hover over the word, click the lightbulb (in this case, a screwdriver) that pops up, and choose Create definition of 'Calculate' in Calculator.cpp.



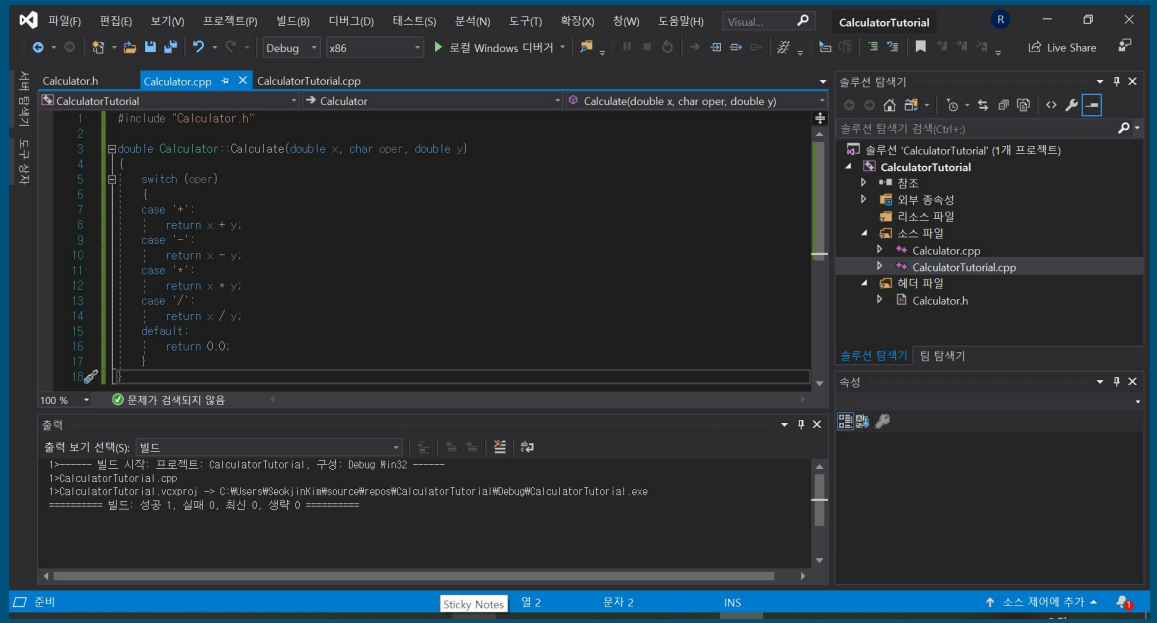
Your first C++ console project - Add code to do some math.

4. A pop-up appears that gives you a peek of the code change that was made in the other file. The code was added to *Calculator.cpp*. Currently, it just returns 0.0. Let's change that. Press Esc to close the pop-up.



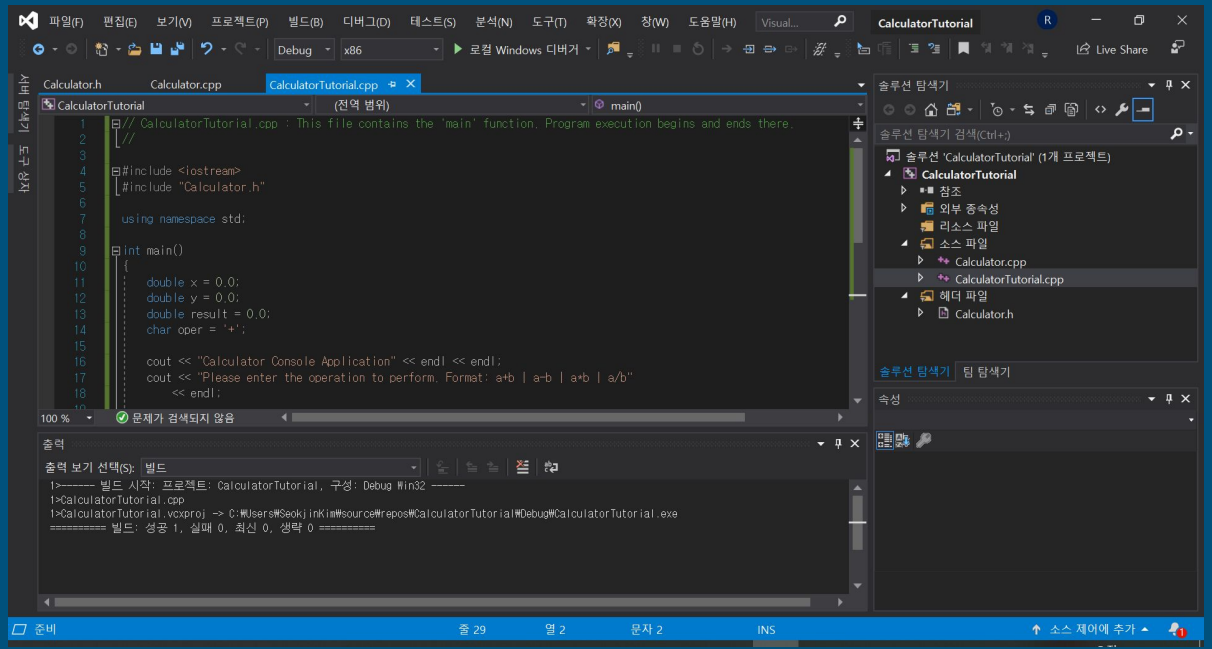
Your first C++ console project - Add code to do some math.

5. Switch to the ***Calculator.cpp*** file in the editor window. Remove the Calculator() and ~Calculator() sections (as you did in the .h file) and edit the code to match ***Calculator.cpp*** in github.



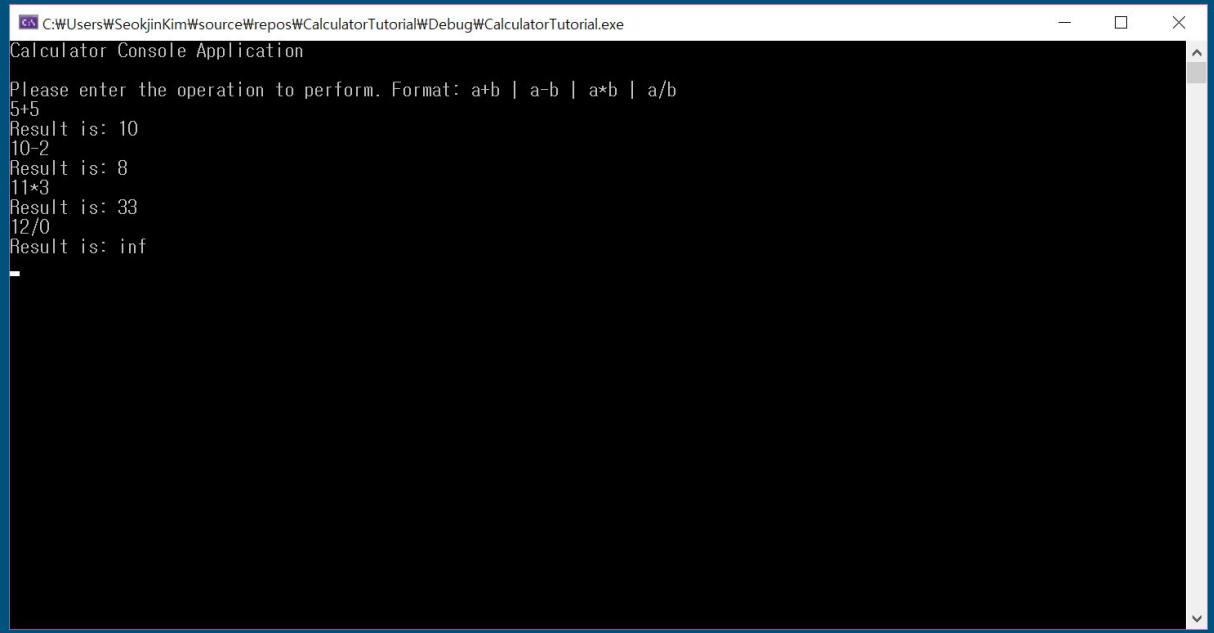
Your first C++ console project - Call the Calculator class member functions

1. Now let's update the main function in *CalculatorTutorial.cpp* to match CalculatorTutorial2.cpp in github.



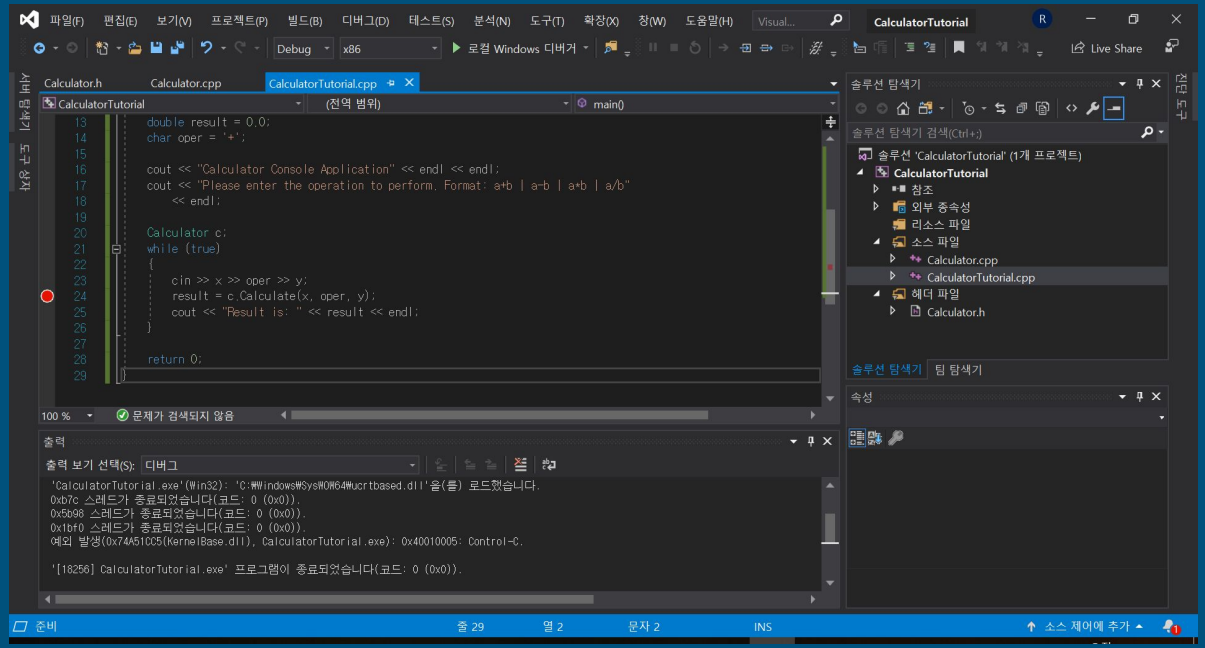
Your first C++ console project - Call the Calculator class member functions

2. Build and test the code! Check the result when entered ***12/0!*** It's definitely a bug!!!



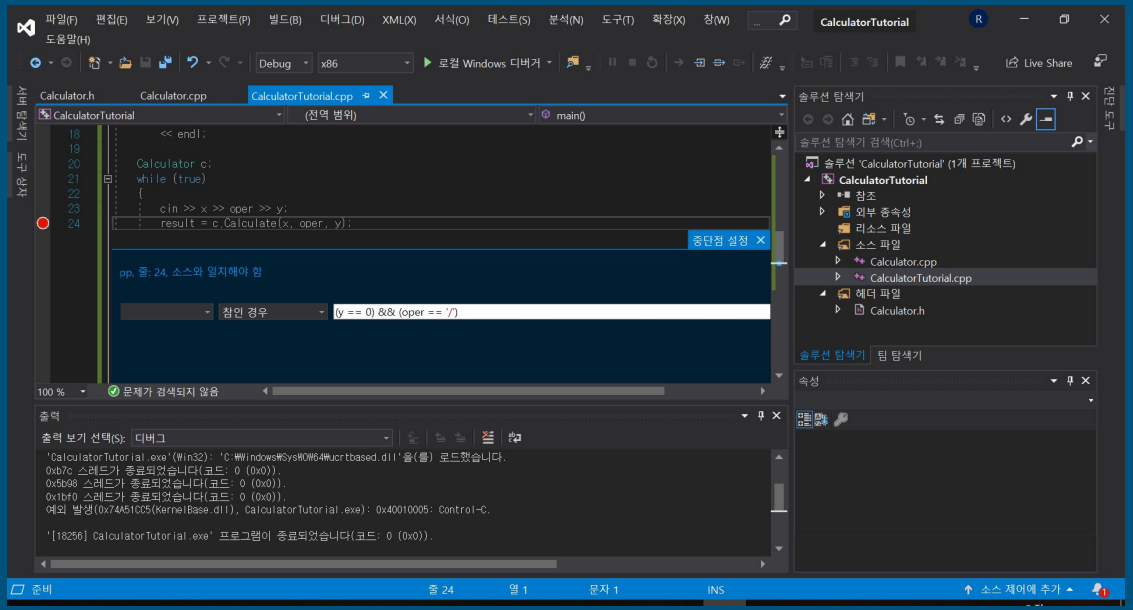
Your first C++ console project - Debug the app

1. Set a breakpoint on the ***result = c.Calculate(x, oper, y);*** line, just after the user was asked for input. To set the breakpoint, click next to the line in the gray vertical bar along the left edge of the editor window. A red dot appears.



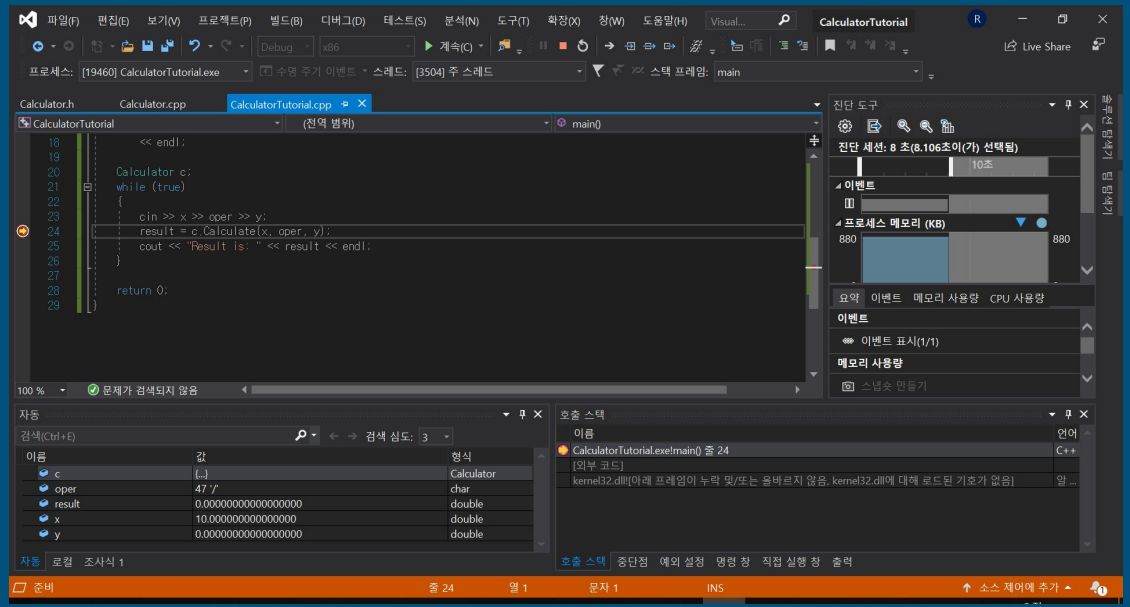
Your first C++ console project - Debug the app

2. Right-click the red dot that represents the breakpoint, and choose ***Conditions***.In the edit box for the condition, enter ***(y == 0) && (oper == '/')***. Choose the Close button when you're done. The condition is saved automatically. Now we pause execution at the breakpoint specifically if a division by 0 is attempted.



Your first C++ console project - Debug the app

3. To debug the program, press F5, or choose the Local Windows Debuggertoolbar button that has the green arrow icon. In your console app, if you enter something like "5 - 0", the program behaves normally and keeps running.However, if you type "10 / 0", it pauses at the breakpoint.



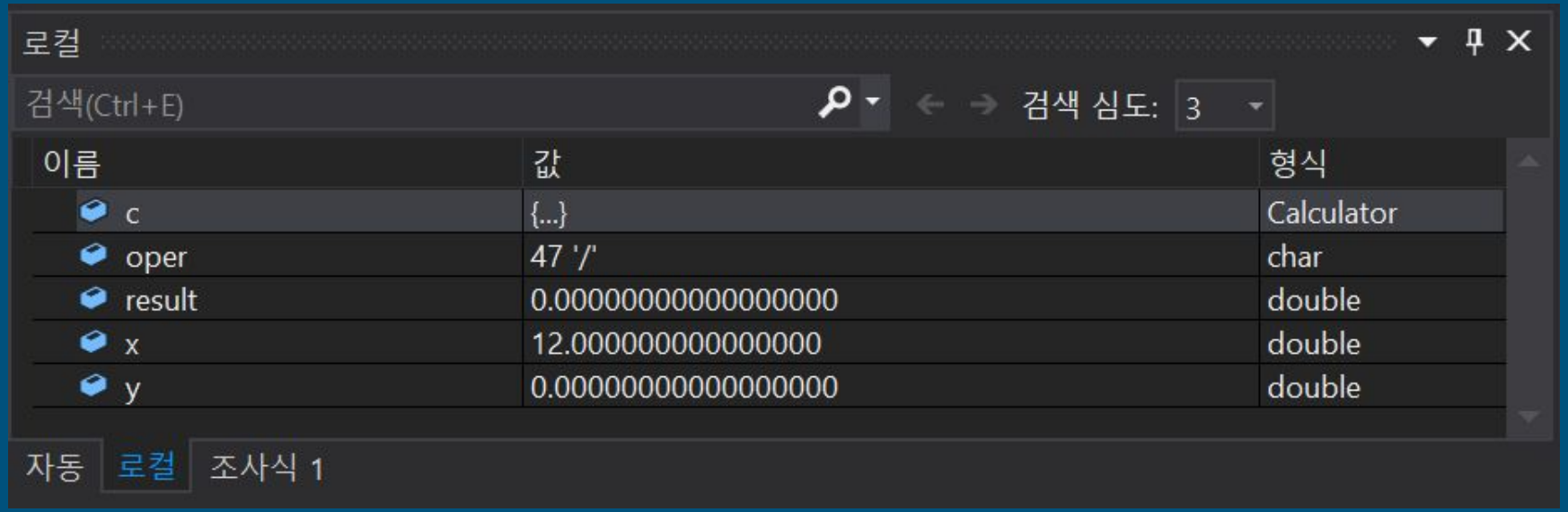
Your first C++ console project - Useful windows in the debugger

Whenever you debug your code, you may notice that some new windows appear.These windows can assist your debugging experience.

Take a look at the ***Autos*** window. The Autos window shows you the current values of variables used at least three lines before and up to the current line.

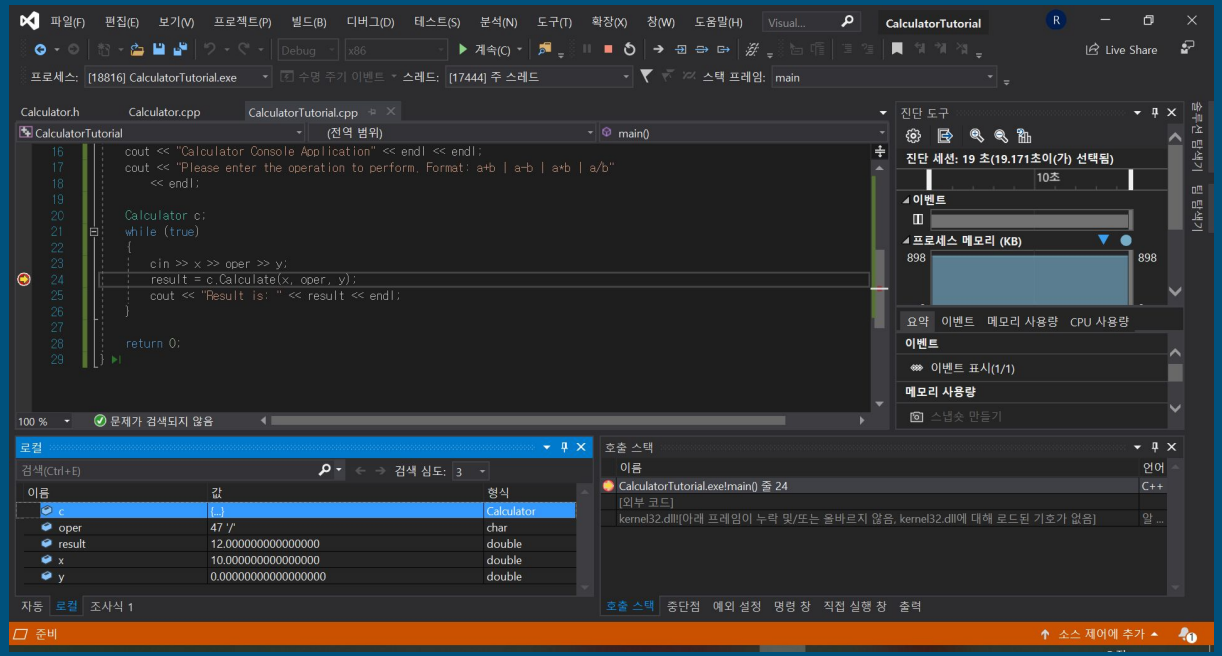
To see all of the variables from that function, switch to the ***Locals*** window. You can actually modify the values of these variables while debugging, to see what effect they would have on the program.

Your first C++ console project - Useful windows in the debugger



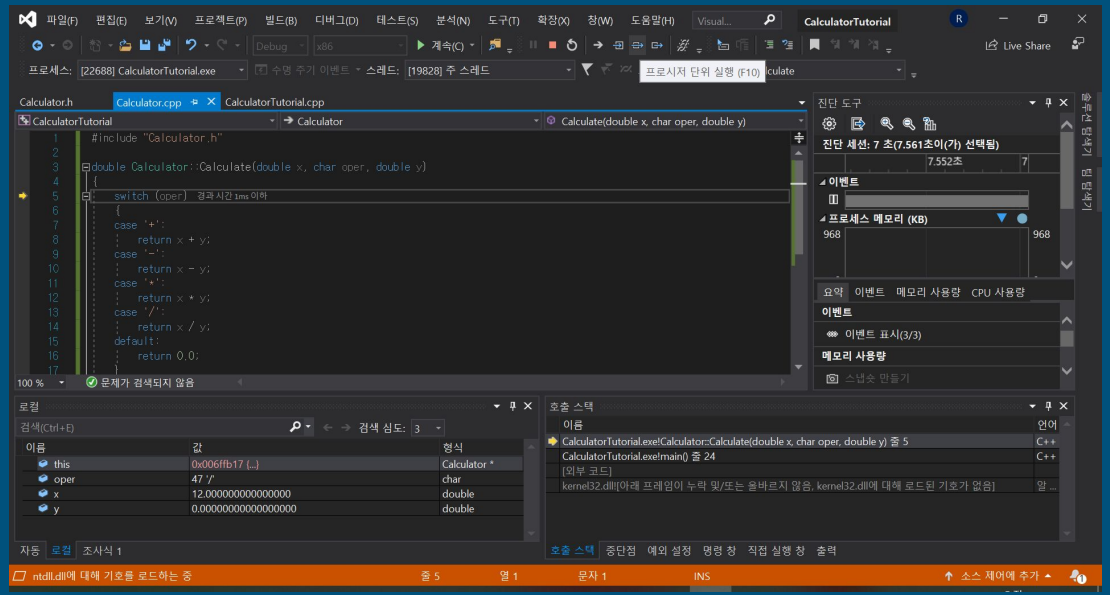
Your first C++ console project - To continue debugging

1. The yellow line on the left shows the current point of execution. The current line calls Calculate, so ***press F11*** to Step Into the function. You'll find yourself in the body of the Calculate function.



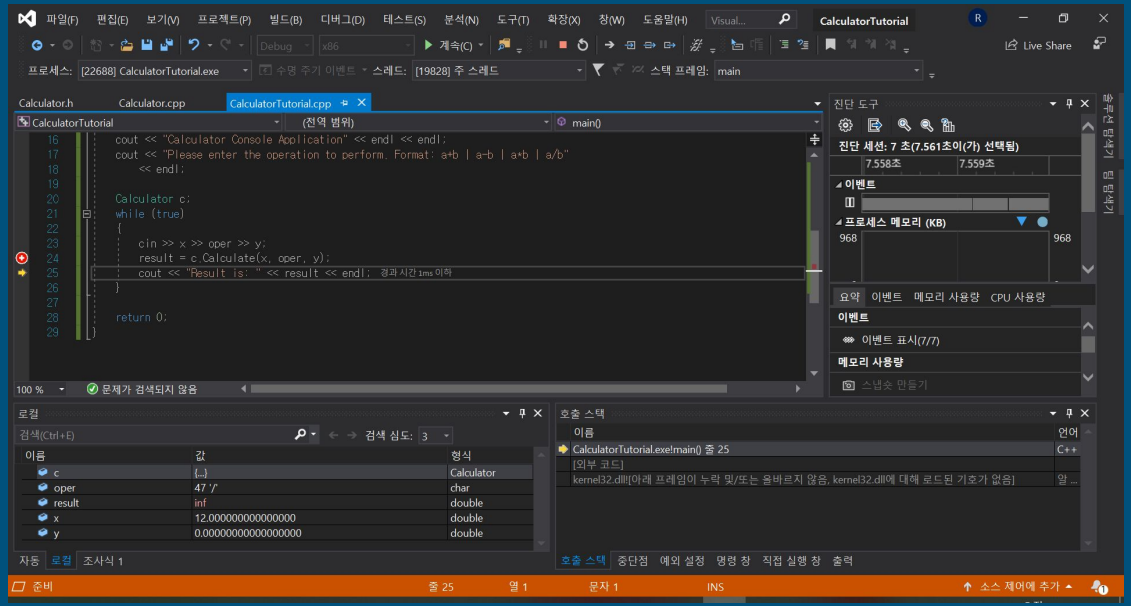
Your first C++ console project - To continue debugging

2. Now that the point of execution is at the start of the Calculate function, press F10 to move to the next line in the program's execution. F10 is also known as Step Over. You can use Step Over to move from line to line, without delving into the details of what is occurring in each part of the line.



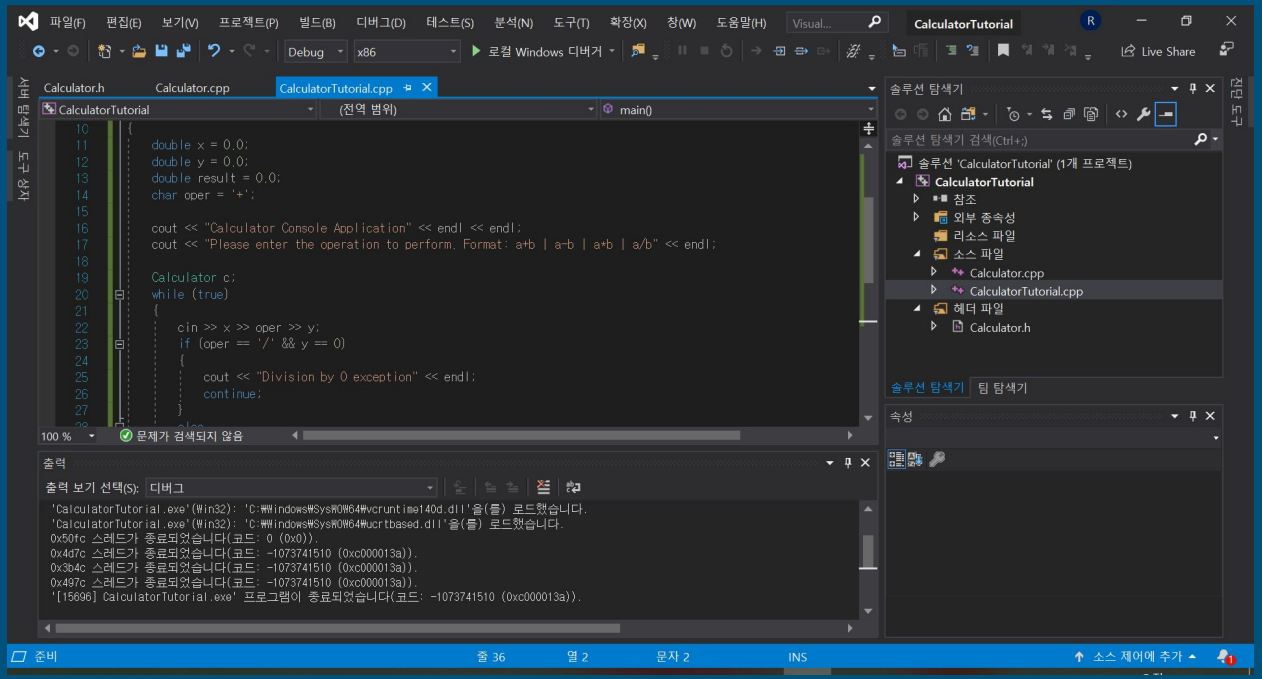
Your first C++ console project - To continue debugging

3. Continue using ***F10*** to Step Over each line until you get back to the main()function in the other file, and stop on the cout line. You'll see its value is listed as "inf", which doesn't look right, so let's fix it.



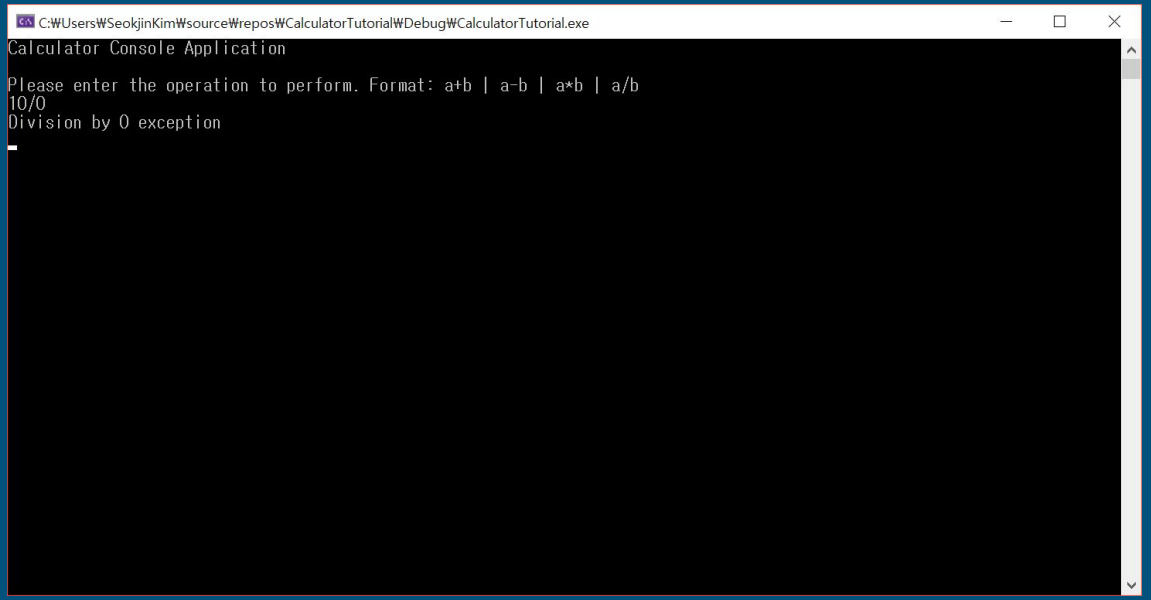
Your first C++ console project - To fix the "divide by zero" error

1. Edit the code to match ***CalculatorTutorial3.cpp*** in github.



Your first C++ console project - To fix the "divide by zero" error

2. Now you can see that error message is printed when entered 10/0!

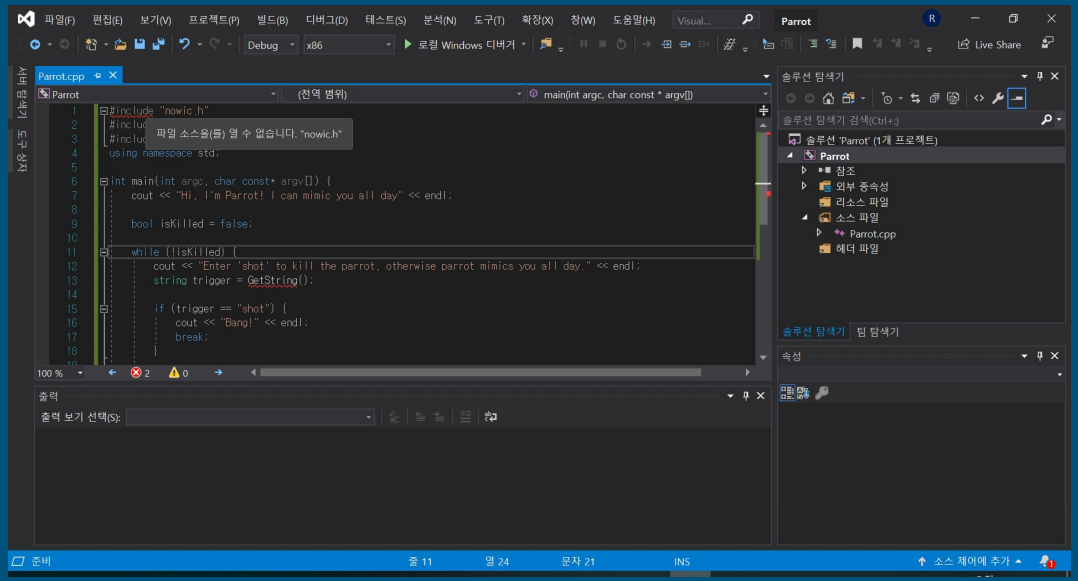


Happy Coding :~)

How to handle external library in Visual Stuio

How to handle external library in VS

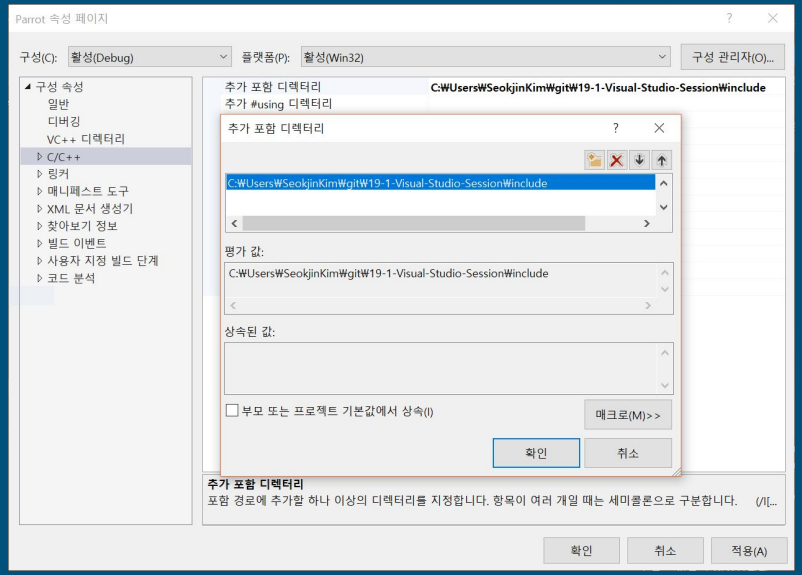
1. Create a new console project named ‘Parrot’.
2. Edit the ***Parrot.cpp*** to match ***Parrot.cpp*** in github.
3. You'll see a red squiggle appear under #include and GetString().



How to handle external library in VS

4. You must enter the path to the header file

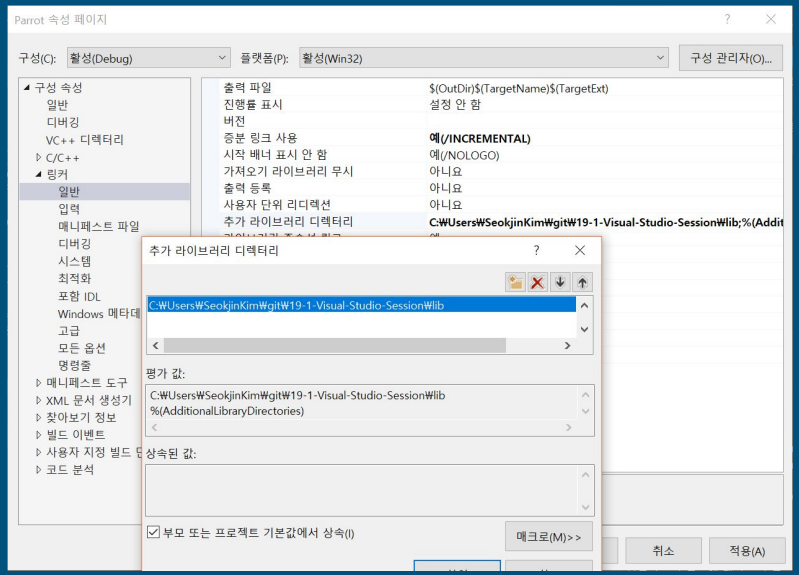
Solution Explorer >> Right click on the project >> setting >> C/C++ >> general >> additional include directories



How to handle external library in VS

5. You must enter the path to the static library

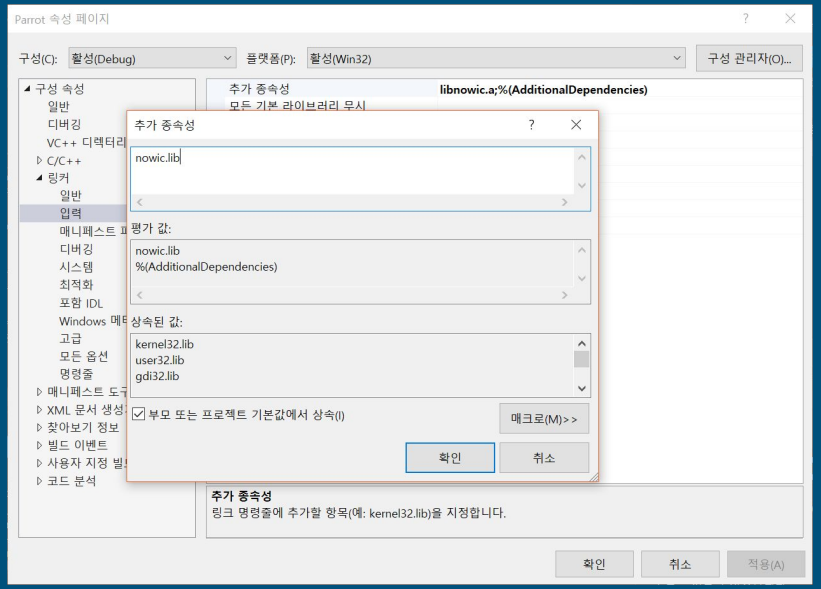
Solution Explorer >> Right click on the project >> setting >> Linker >> general >> additional library directories



How to handle external library in VS

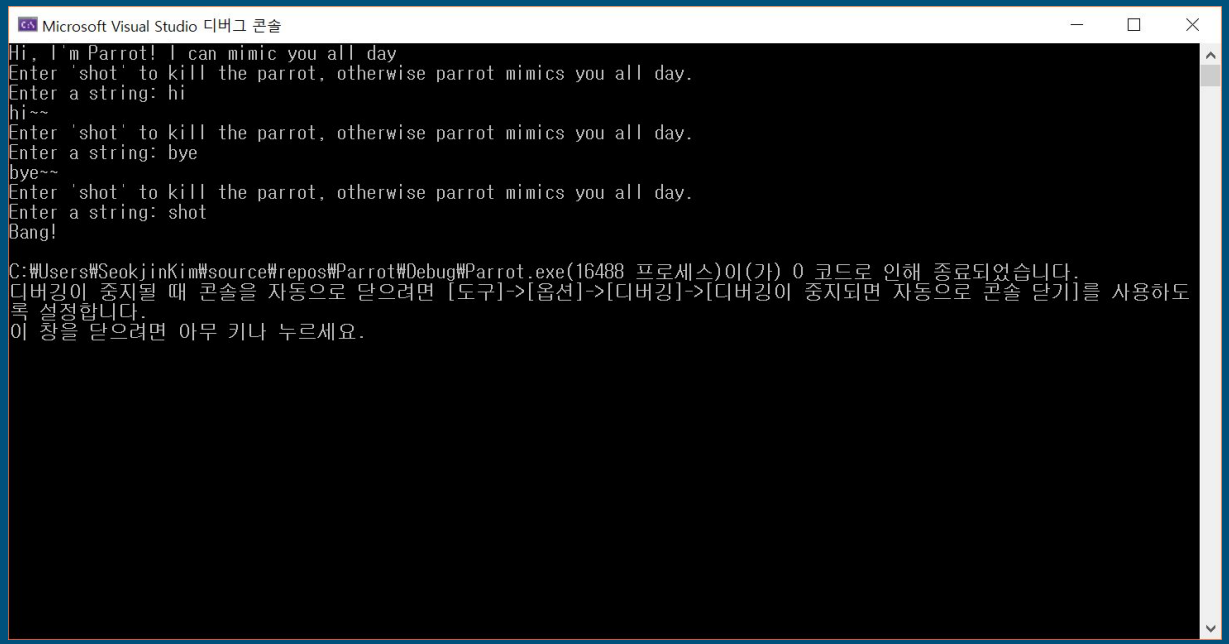
5. You must enter the name of static library

Solution Explorer >> Right click on the project >> setting >> Linker >> general >> additional dependencies



How to handle external library in VS

7. Build and test Parrot!



References

Create a C++ console app project

https://docs.microsoft.com/ko-kr/cpp/get-started/tutorial-console-cpp?view=vs-2019

Visual Studio IDE Introduction

https://docs.microsoft.com/ko-kr/visualstudio/get-started/visual-studio-ide?view=vs-2019

How to handle external library in Visual Studio

https://wnsgml972.github.io/setting/dll_lib.html