

# Your First C++ Program

September 2, 2009

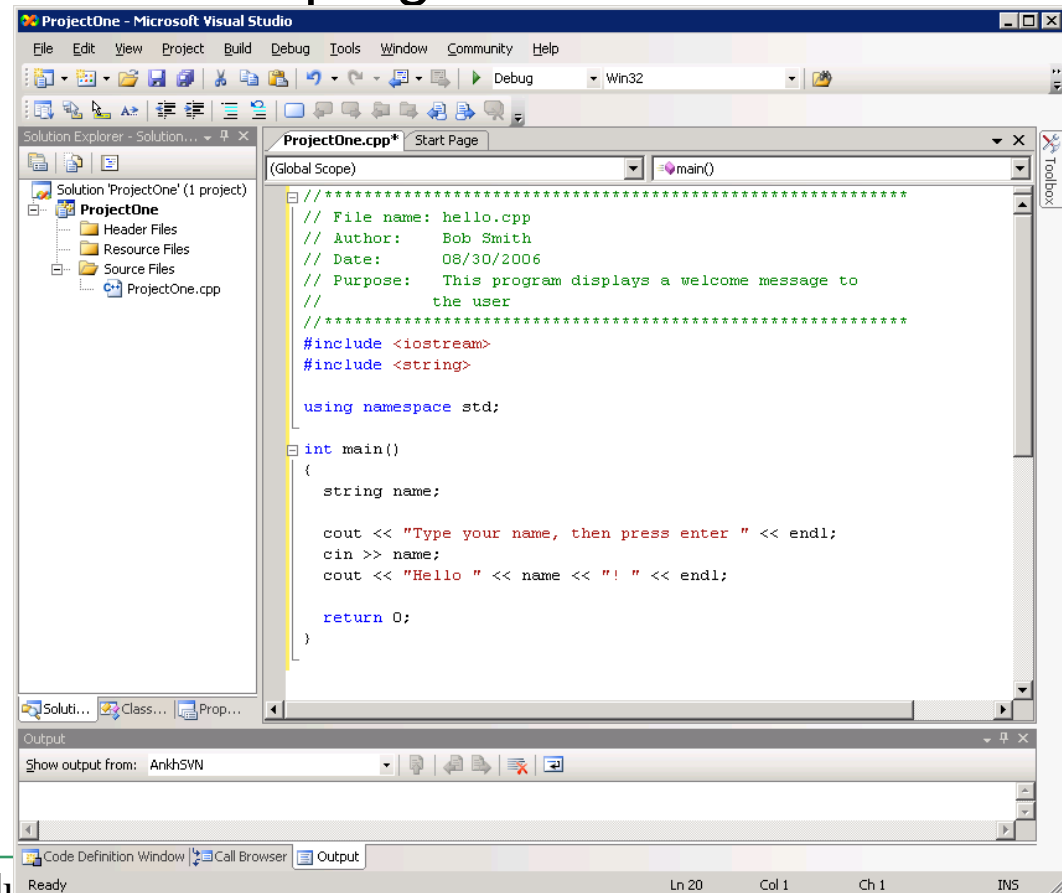
# C++

---

- Based on the C programming language
- One of today's most popular programming languages
  - used extensively in industry
- Language + Libraries
  - Libraries: bits of programs you can use

# Visual Studio

- Microsoft Tools
  - Integrated **D**evelopment **E**nvironment
  - all the tools you need to write a program!
- Source code editor
  - highlights source code as you type
- Debugger
- Extra Libraries



# Problem

---

- Programs are written to solve problems
  - source code
  - process data
- Imagine that you have been asked to solve the following problem
  - Write a program that asks the user to enter their name and display a personalized welcome message

# Your First C++ Program

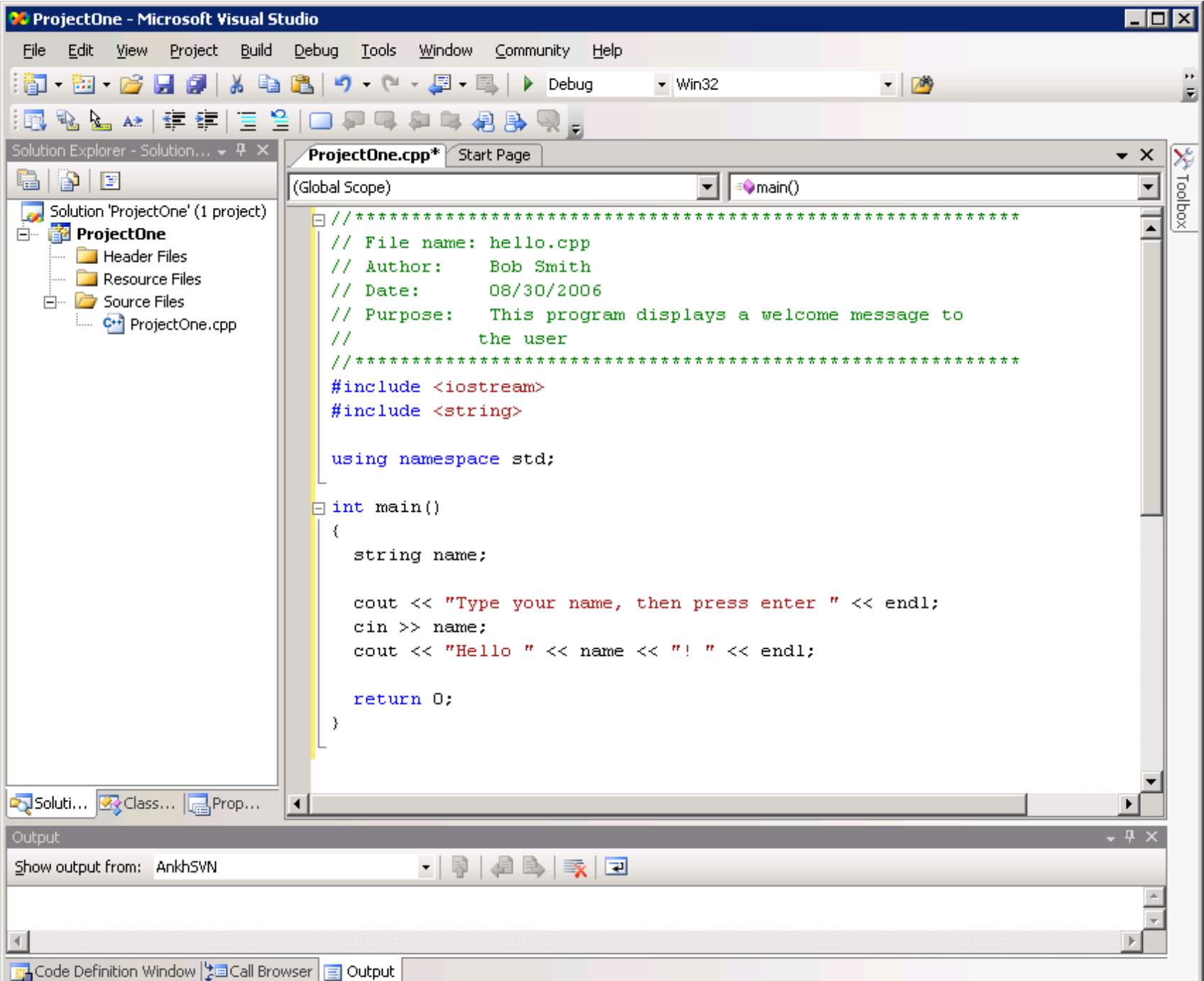
---

```
//*****  
// File name: hello.cpp  
// Author:    Bob Smith  
// Date:     08/30/2009  
// Purpose:  This program displays a welcome message to  
//           the user  
//*****  
#include <iostream>  
#include <string>  
  
using namespace std;  
  
int main()  
{  
    string name;  
  
    cout << "Type your name, then press enter " << endl;  
    cin >> name;  
    cout << "Hello " << name << "! " << endl;  
  
    return 0;  
}
```

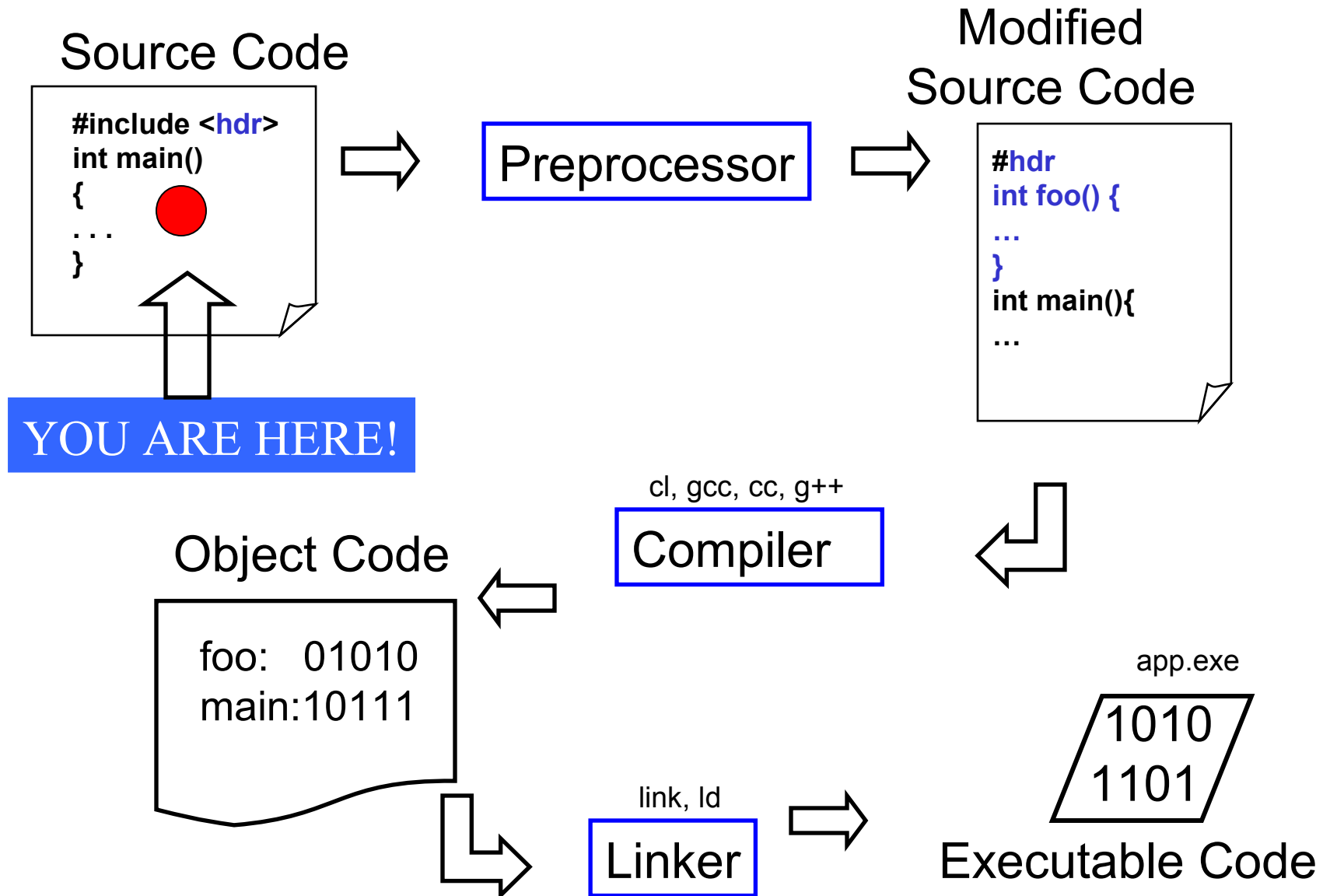
---

*Program Output:*

```
Type your name, then press enter  
Doug  
Hello Doug!
```



# Building an Application



# Language Elements

---

- Key Words
  - Have special meaning in C++
  - `using namespace int`
- Programmer-Defined Identifiers
  - Names made up by the programmer
  - `name`
- Operators
  - Perform operations
  - `* =`
- Punctuation
  - Used to mark the beginning and end of the program
  - `;`



# Syntax (Grammar)

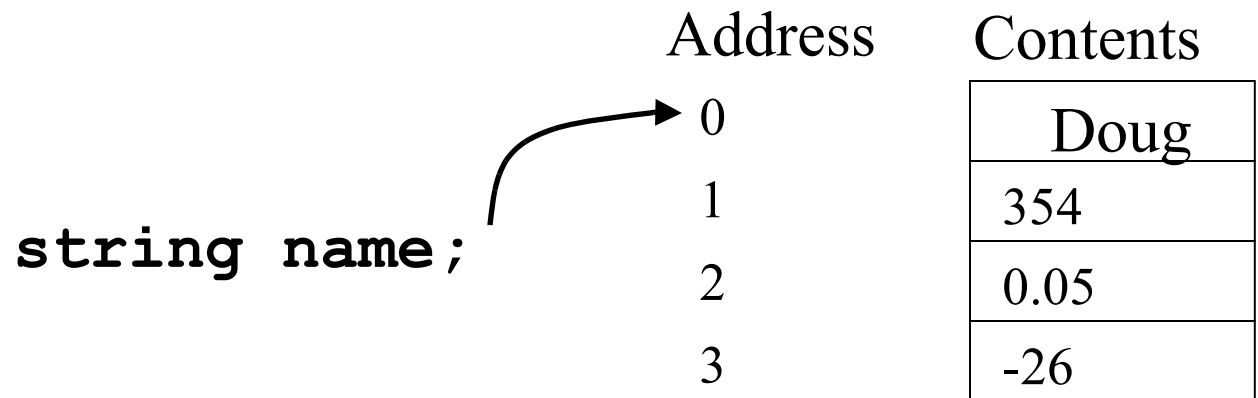
---

- Rules that must be followed when constructing a program
- Controls the use of key words, programmer-defined identifiers, operators, and punctuation

# Variables

---

- Names storage location in the computers memory
- Holds **data**
- The data can change



# Program Components

---

- The C++ program on the previous slide consists of the following elements:
  - Comments
  - Preprocessor directives
  - Standard namespace
  - **main** function
  - Declaration statements
  - Executable statements

# Comments

---

- Comments are
  - How you explain in English what the different parts of your program do
  - Ignored by the compiler
  - Very important for you and me
- The editor in Visual Studio will colour code your comments. They will be green

# Comments

---

- There are two ways to write comments
  - `// I am a comment`
    - Anything after `//` to the end of the line will be a comment
  - `/* I am another comment */`
    - You must start the comment with `/*` and end it with `*/` in this style of comment

# Preprocessor directives

---

- `#include <iostream>`
- `#` signifies preprocessor directive
- Processed before program translation
- `#include` tells the preprocessor to look for libraries
- `<>` signifies part of standard C++ libraries
- We'll see other examples of preprocessor directives later

# Preprocessor directives

---

- **`iostream`** is the input/output stream library
- It is needed to output data to the screen and read in data from the keyboard
- **`#include`** takes the contents of the library file and places them in the current program

# Namespace std

---

- `using namespace std;`
- Indicates that we will be using objects (`cout` & `cin`) that are named in a region called `std`
  - predefined objects C++ provides
- The statement ends in a semicolon
- The statement appears in all our programs



# main Function

---

```
int main()  
{  
    // program statements  
    return 0;  
}
```

- Where the start of your program execution begins
- **return 0;** ends the main function and indicates that the program terminated successfully
- Everything within the double braces `{}` should be indented
  - Style!

# Program Statements

---

- There are two types of statements that you can write inside the main (or any other) function
  - Declaration statements
    - Specify the **data** that is needed by the program
  - Executable statements
    - Perform **operations**
- All statements must end with a semicolon;

# Program Statements

---

- Declaration statements

```
string name;
```

- Executable statements

```
cout << "Type your name, then press enter" << endl;
```

```
cin >> name;
```

```
cout << "Hello " << name << "!" << endl;
```

# Program Skeleton

---

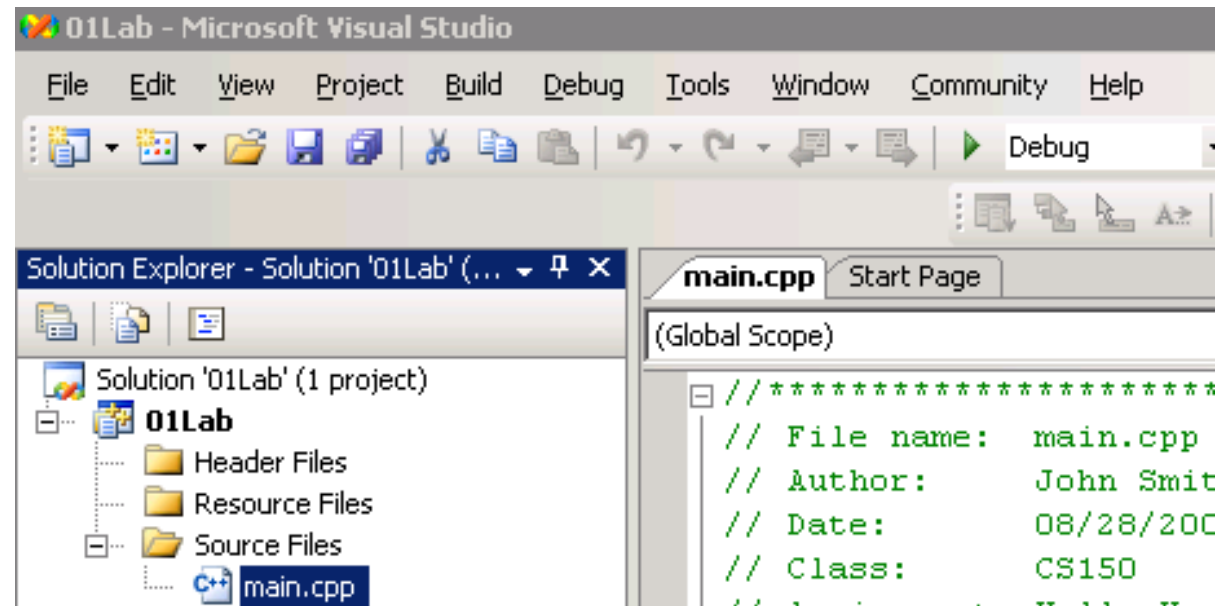
- All programs in C++ should have the following skeleton

```
//*****  
// File name: filename.cpp  
// Author:    Your Name  
// Date:     09/01/2009  
// Purpose:   Description about what the program does  
//*****  
#include <iostream>  
  
using namespace std;  
  
int main()  
{  
    // declaration statements  
  
    // executable statements  
  
    return 0;  
}
```

# Do try this at home (or in the lab) !

---

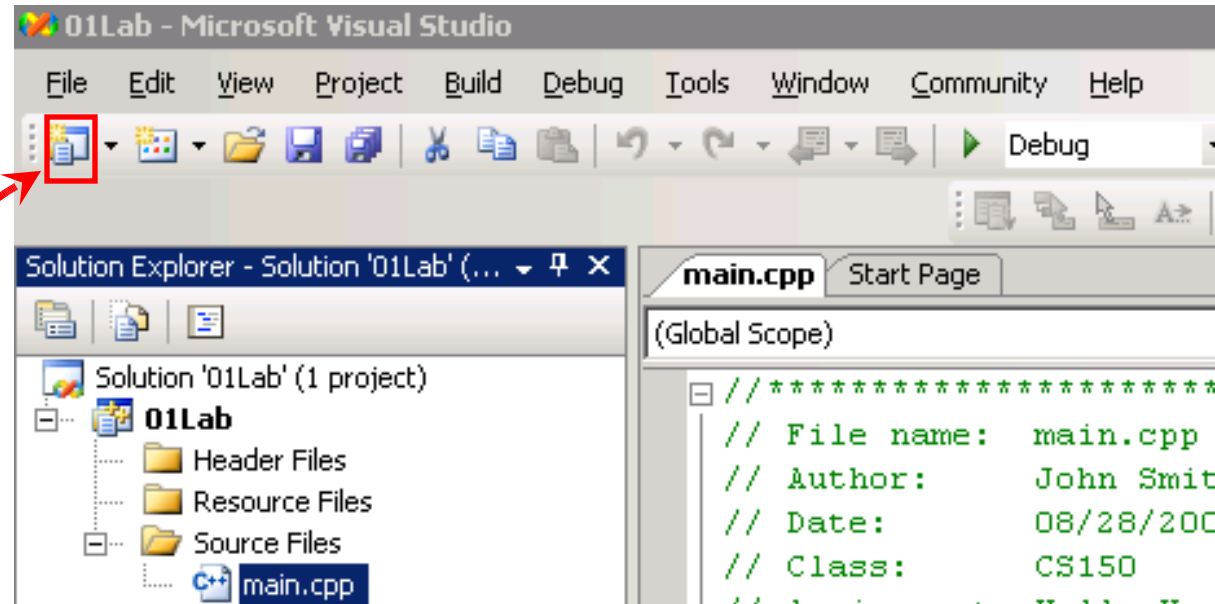
- Add New Project
  - Visual C++ | General | Empty Project
- Add New Item
  - C++ Source File



# Do try this at home (or in the lab) !

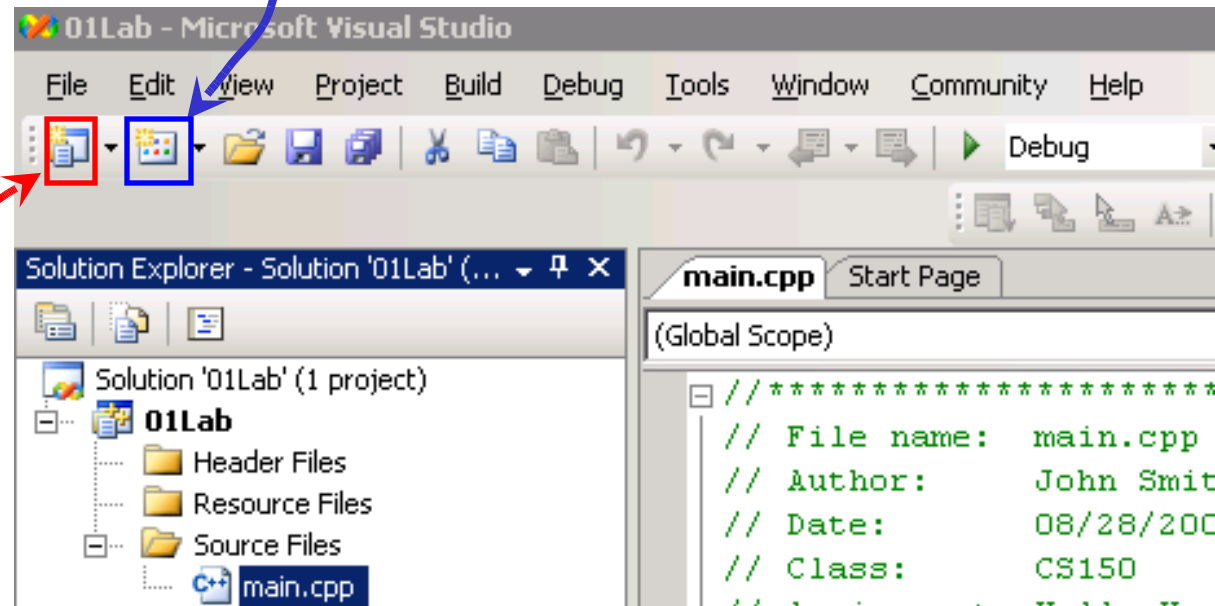
---

- Add New Project
  - Visual C++ | General | Empty Project
- Add New Item
  - C++ Source File



# Do try this at home (or in the lab) !

- Add New Project
  - Visual C++ | General | Empty Project
- Add New Item
  - C++ Source File



# Summary

---

- Today we
  - Wrote our first C++ program
  - Introduced the basic components of a C++ program
- We covered p. 12 - 33 from your textbook