

Introduction to Computing

Lecture 3

Dr. Naveed Anwar Bhatti

Webpage: naveedanwarbhatti.github.io

Introduction to C++ Programming

Machine
Language

Assembly
Language

High-Level
Language

Machine Language:

- Only language computer directly understands
- Defined by hardware design
 - Machine-dependent
- Generally consist of 0s and 1s (**Binary Language**)
- Instruct computers to perform elementary operations
 - One at a time
- Cumbersome for humans
- Example:

```
10100111010
01011101001
```

Assembly Language:

- English-like abbreviations representing elementary computer operations
- Clearer to humans
- Incomprehensible to computers
 - Translator programs (assemblers)
 - Convert to machine language
- Example:

```
LOAD  BASEPAY
ADD   OVERPAY
STORE GROSSPAY
```

High-Level Language:

- Similar to everyday English, use common mathematical notations
- Single statements accomplish substantial tasks
 - Assembly language requires many instructions to accomplish simple tasks
- Translator programs (compilers)
 - Convert to machine language
- Example:

```
grossPay = basePay + overtimePay
```



History of C and C++

- History of C

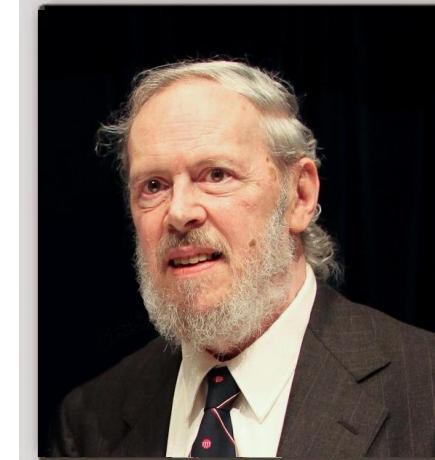
- Evolved from other programming language 'B'
- Dennis Ritchie (Bell Laboratories)
 - Added data typing, other features
- Hardware independent

- History of C++

- Extension of C
- Early 1980s: Bjarne Stroustrup (Bell Laboratories)
- Provides capabilities for object-oriented programming
 - Objects: reusable software components
 - Model items in real world
 - Object-oriented programs
 - Easy to understand, correct and modify

Both Died October 2011

To Be Remembered



Dennis Ritchie
September 9, 1941 – October 12, 2011

the creator of
C programming language
and
Unix Operating System



Steve Jobs
24 February 1955 – October 5, 2011

the founder of
Apple & Pixar
and
every great thing
they made



- C++ programs
 - Built from pieces called classes and functions
- C++ standard library
 - Rich collections of existing classes and functions



Basics of a Typical C++ Environment

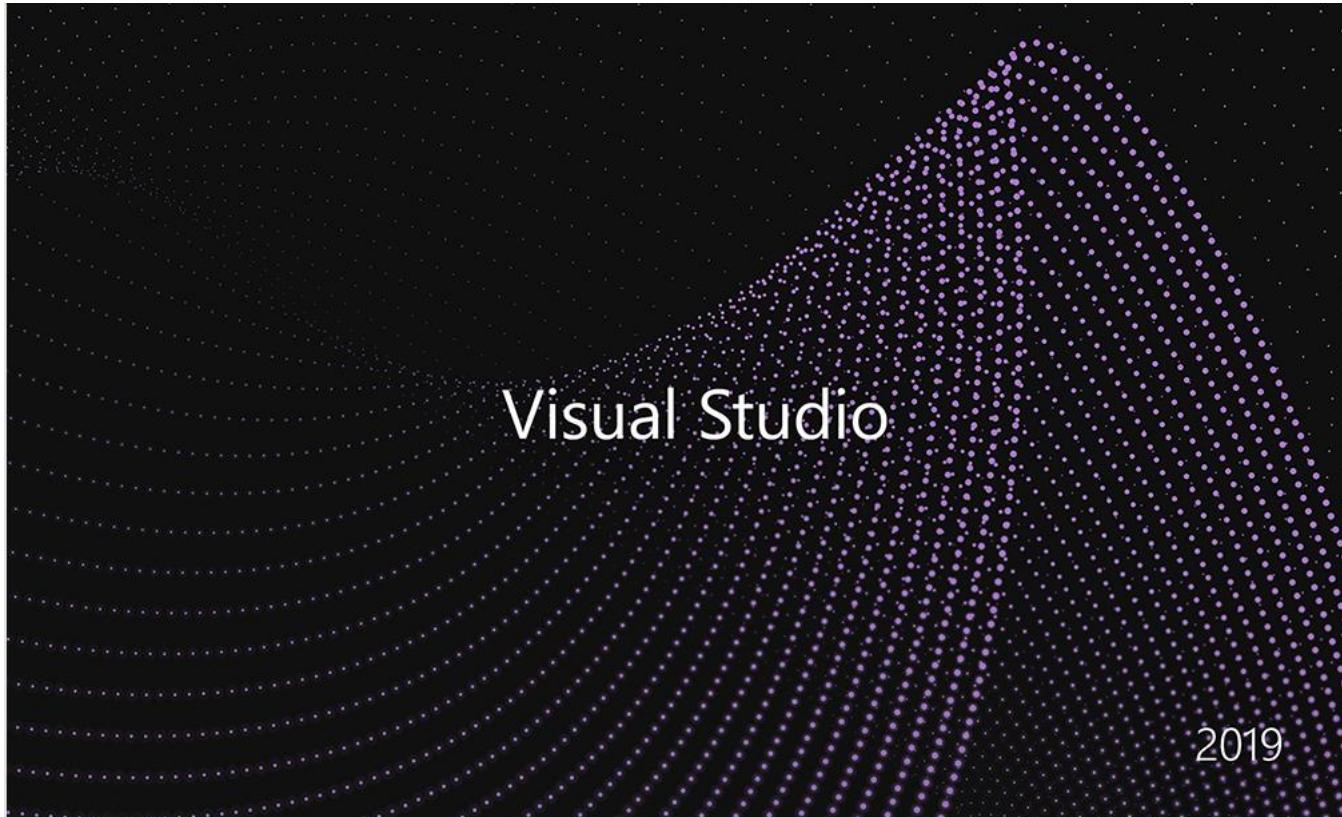
- C++ systems
 - IDE (Integrated Development Environment)
 - Language
 - C++ Standard Library
- C++ program extension
 - .cpp
 - .h (header file)



Web-based IDE's
can work as well,
but functionality
is limited

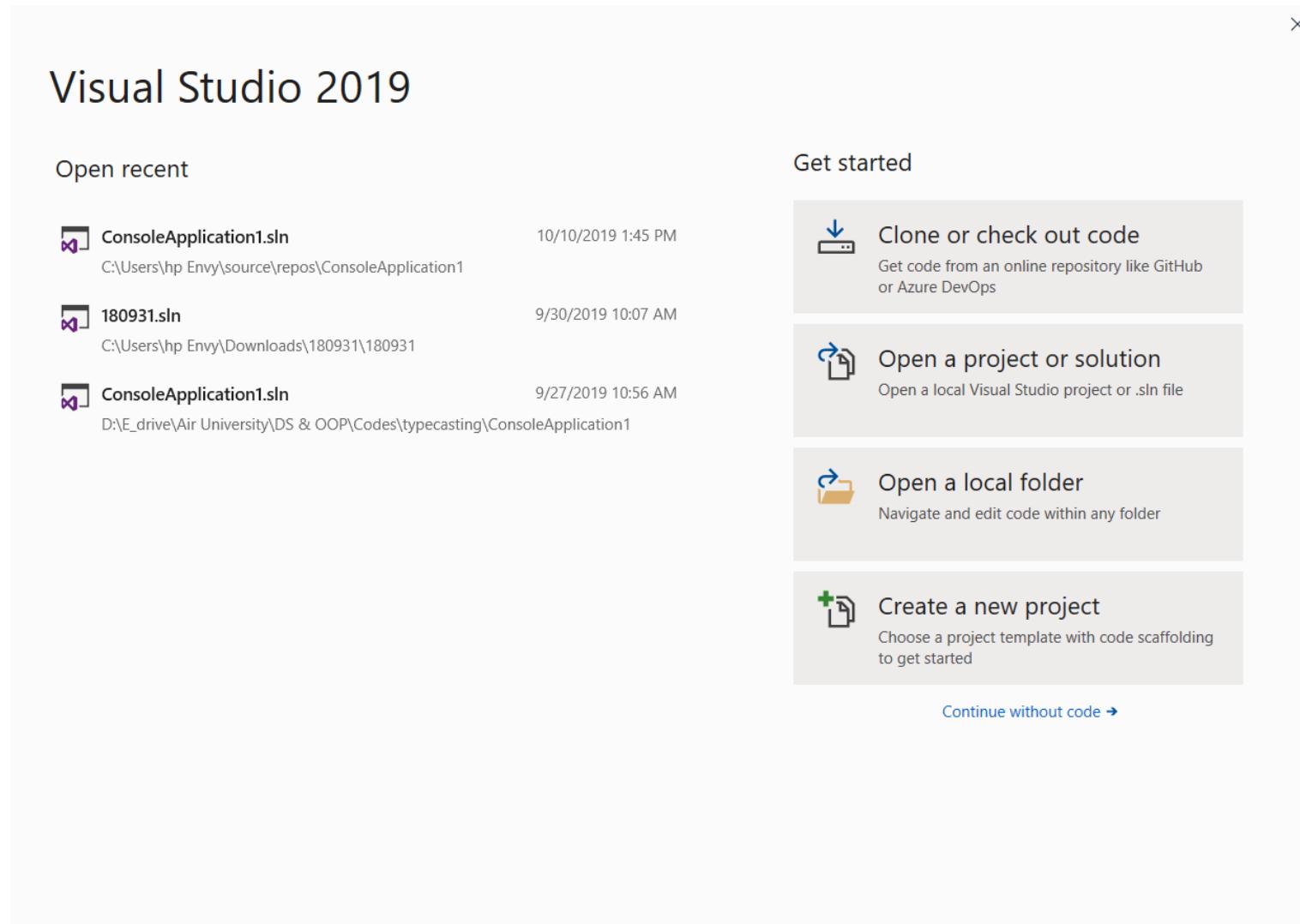


Basics of a Typical C++ Environment





Basics of a Typical C++ Environment





Basics of a Typical C++ Environment

Recent project templates

Console App C++

Search for templates (Alt+S) Language Platform Project type

 **Console App (.NET Core)**
A project for creating a command-line application that can run on .NET Core on Windows, Linux and MacOS.
C# Linux macOS Windows Console

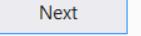
 **WPF App (.NET Framework)**
Windows Presentation Foundation client application
C# Windows Desktop

 **Class Library (.NET Standard)**
A project for creating a class library that targets .NET Standard.
C# Android iOS Linux macOS Windows Library

 **Empty Project**
Start from scratch with C++ for Windows. Provides no starting files.
C++ Windows Console

 **Console App**
Run code in a Windows terminal. Prints "Hello World" by default.
C++ Windows Console

 **Windows Desktop Wizard**
Create your own Windows app using a wizard.
C++ Windows Desktop Console Library



Basics of a Typical C++ Environment

Create a new project

Recent project templates

Console App C++

Search for templates (Alt+S) Language Platform Project type

Language dropdown menu:

- All Languages
- C#
- C++ **(selected)**
- F#
- Java
- JavaScript
- Python

Platform dropdown menu:

- Windows, Linux and macOS
- C# Linux macOS

Project type dropdown menu:

- Console
- Query Language
- TypeScript
- Visual Basic

Console App (.NET Core)

A project for creating a console application that can run on .NET Core on Windows, Linux and macOS.

C# Linux macOS

WPF App (.NET Framework)

Windows Presentation Foundation application

C# Windows Desktop

Class Library (.NET Standard)

A project for creating a class library that targets .NET Standard.

C# Android iOS Linux macOS Windows Library

Empty Project

Start from scratch with C++ for Windows. Provides no starting files.

C++ Windows Console

Console App

Run code in a Windows terminal. Prints "Hello World" by default.

C++ Windows Console

Windows Desktop Wizard

Create your own Windows app using a wizard.

C++ Windows Desktop Console Library

Back Next



Basics of a Typical C++ Environment

X

Create a new project

Recent project templates

Filtering by: C++ [Clear filter](#)

 Console App	C++
 Empty Project	Start from scratch with C++ for Windows. Provides no starting files.
 Console App	Run code in a Windows terminal. Prints "Hello World" by default.
 Windows Desktop Wizard	Create your own Windows app using a wizard.
 Windows Desktop Application	A project for an application with a graphical user interface that runs on Windows.
 Shared Items Project	A Shared Items project is used for sharing files between multiple projects.
 Dynamic-Link Library (DLL)	

[Back](#) [Next](#)



Basics of a Typical C++ Environment

X

Configure your new project

Empty Project C++ Windows Console

Project name

Project1

Location

C:\Users\hp Envy\source\repos



Solution name ?

Project1

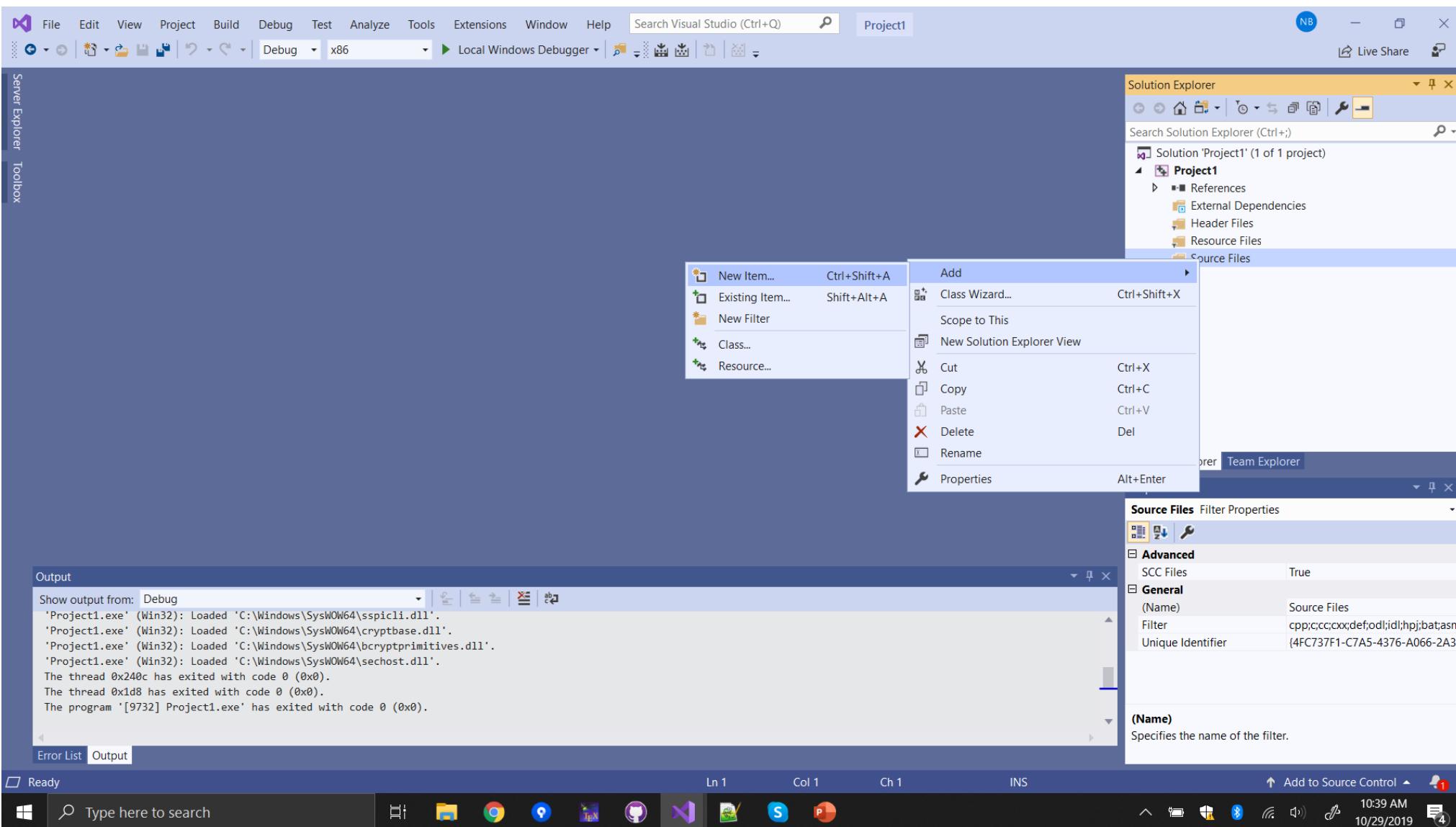
Place solution and project in the same directory

Back

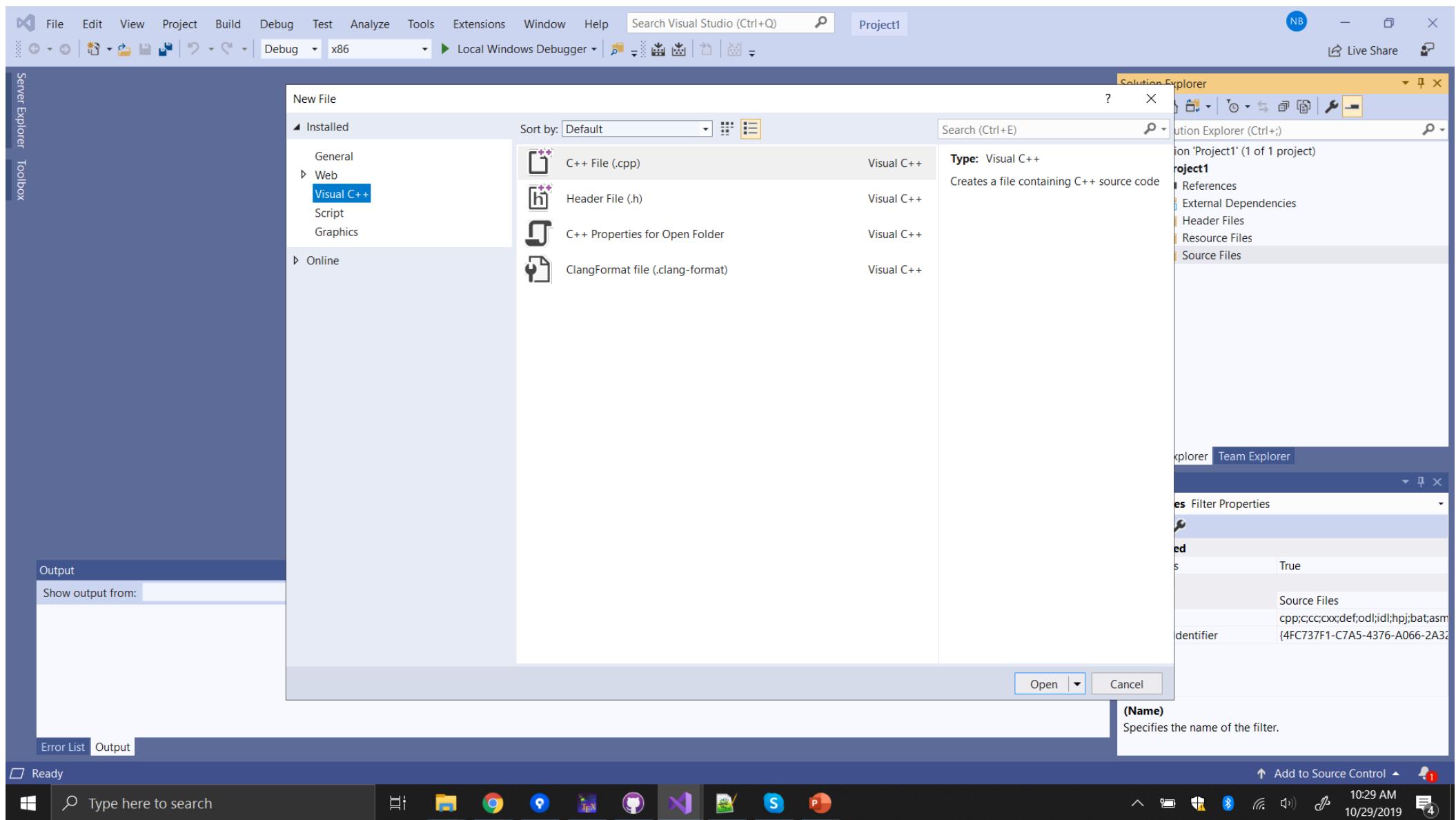
Create



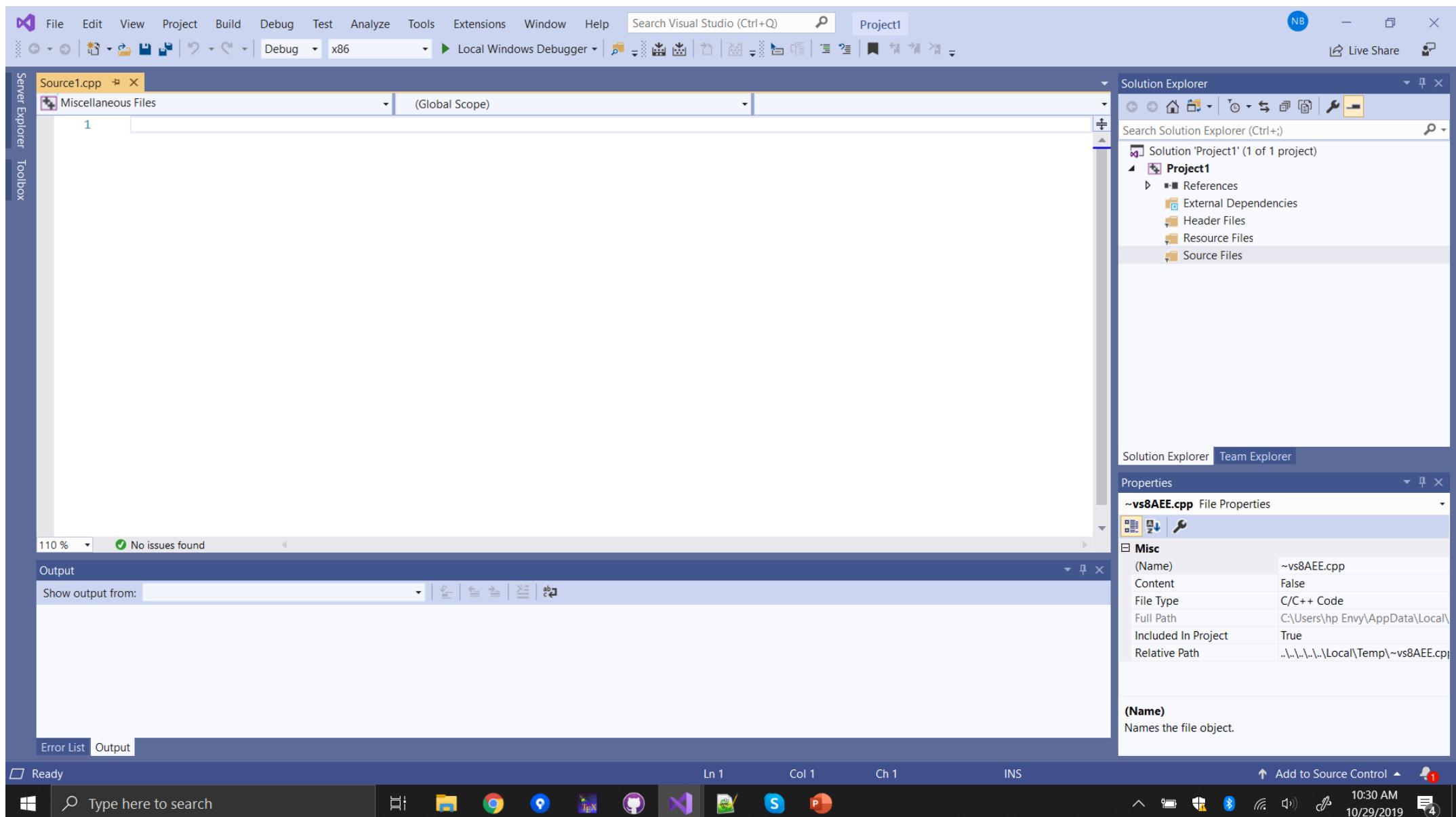
Basics of a Typical C++ Environment



Basics of a Typical C++ Environment



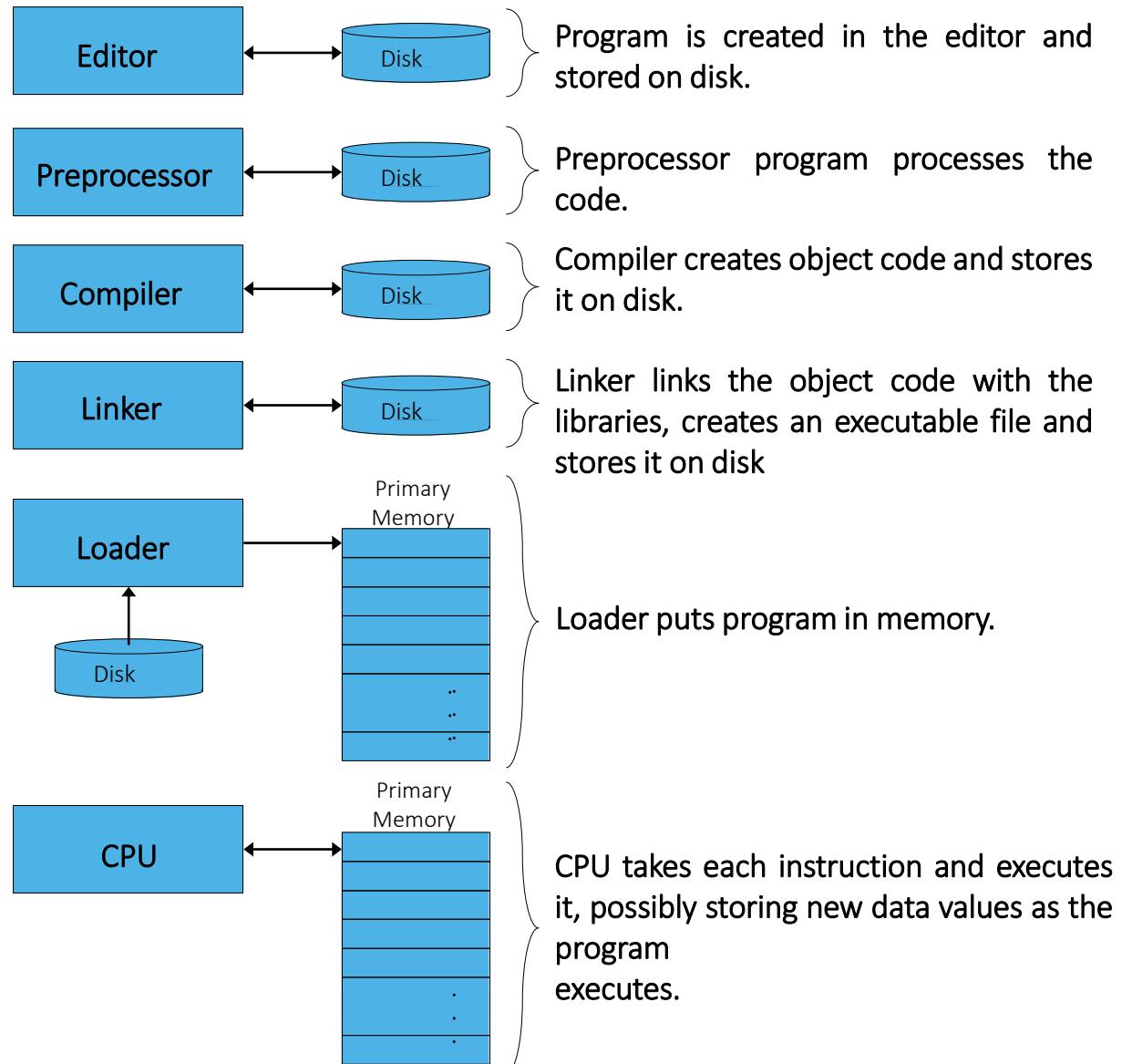
Basics of a Typical C++ Environment



Basics of a Typical C++ Environment

Phases of C++ Programs:

1. Edit
2. Preprocess
3. Compile
4. Link
5. Load
6. Execute





- Common Input/output functions

- **cin**

- Standard input stream
 - Normally keyboard

- **cout**

- Standard output stream
 - Normally computer screen



A Simple Program: Printing a Line of Text

- Before writing the programs
 - Comments
 - Document programs
 - Improve program readability
 - Ignored by compiler
 - Single-line comment
 - Use C's comment /* .. */ OR Begin with // or
 - Preprocessor directives
 - Processed by preprocessor before compiling
 - Begin with #



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.
2. #include <iostream>
3. using namespace std;
4. // function main begins program execution
5. int main()
6. {
7.     cout << "Welcome to C++!\n";
8.
9.     return 0; // indicate that program ended successfully
10.
11. } // end function main
```

Single-line comments.

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Preprocessor directive to include input/output stream header file **<iostream>**.

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

using namespace std
means that we can use names
for objects and variables from
the *standard library*

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Function **main** returns an integer value

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Function **main** appears
exactly once in every C++
program

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Left brace { begins function
body

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\\n";  
8.     // cout is the standard output stream  
9.     return 0; // indicate the program has completed  
10.  
11. } // end function main
```

Standard output stream

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.     return 0; // indicate that program ran successfully
```

Stream insertion operator

10.

11. } // end function main

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Another way to insert a new line, is with the `endl` manipulator

Special character "\n" is used. "n" means new line

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Statements end with a
semicolon ;

Welcome to C++!



A Simple Program: Printing a Line of Text

```
1. // A first program in C++.  
2. #include <iostream>  
3. using namespace std;  
4. // function main begins program execution  
5. int main()  
6. {  
7.     cout << "Welcome to C++!\n";  
8.  
9.     return 0; // indicate that program ended successfully  
10.  
11. } // end function main
```

Welcome to C++!

Keyword **return** is one of several means to exit function; value **0** indicates program terminated successfully



A Simple Program: Printing a Line of Text

- Standard output stream object
 - “Connected” to screen
 - `<<`
 - Stream insertion operator
 - Value to right (right operand) inserted into output stream
- Escape characters
 - `\`
 - Indicates “special” character output

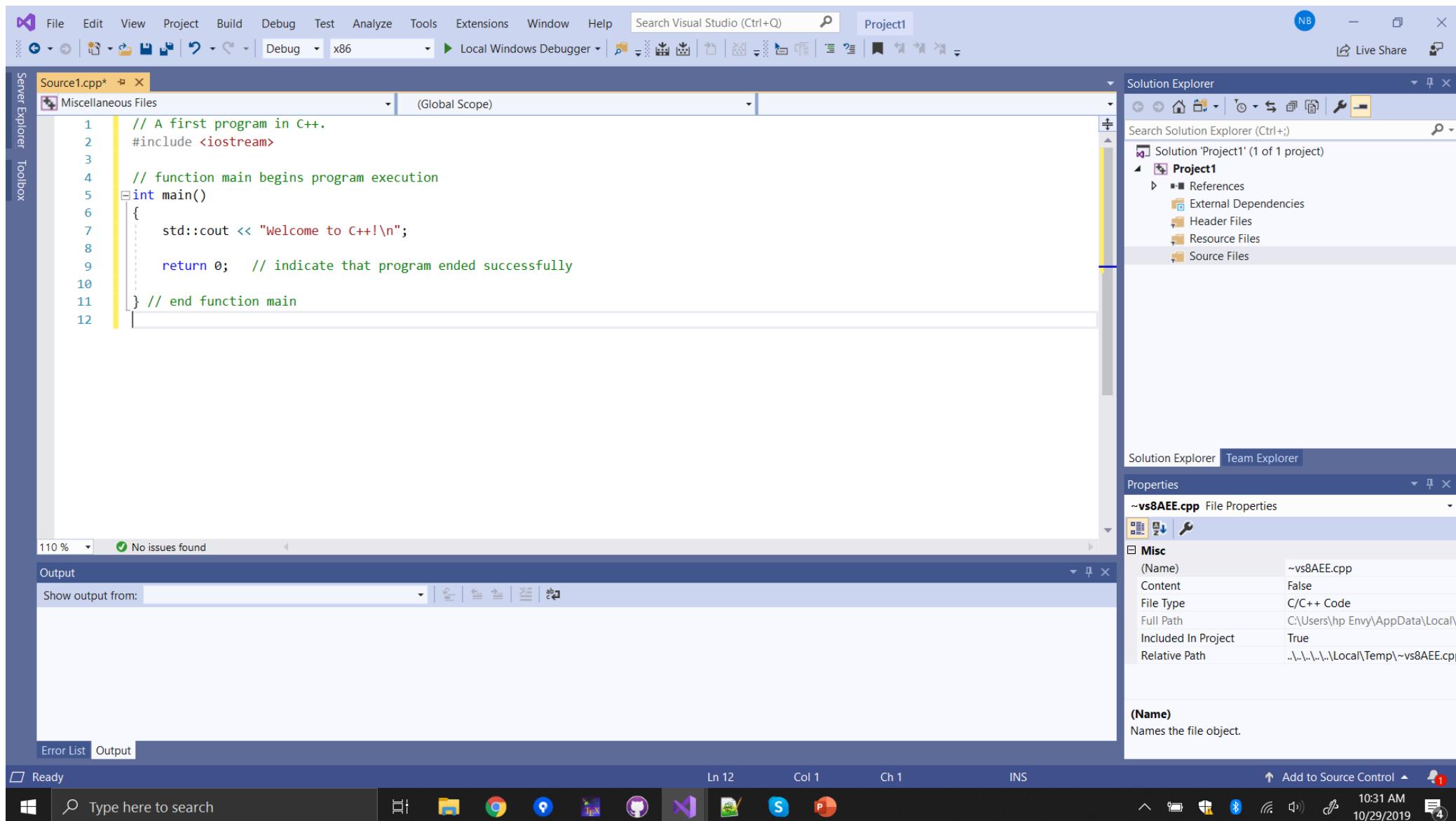


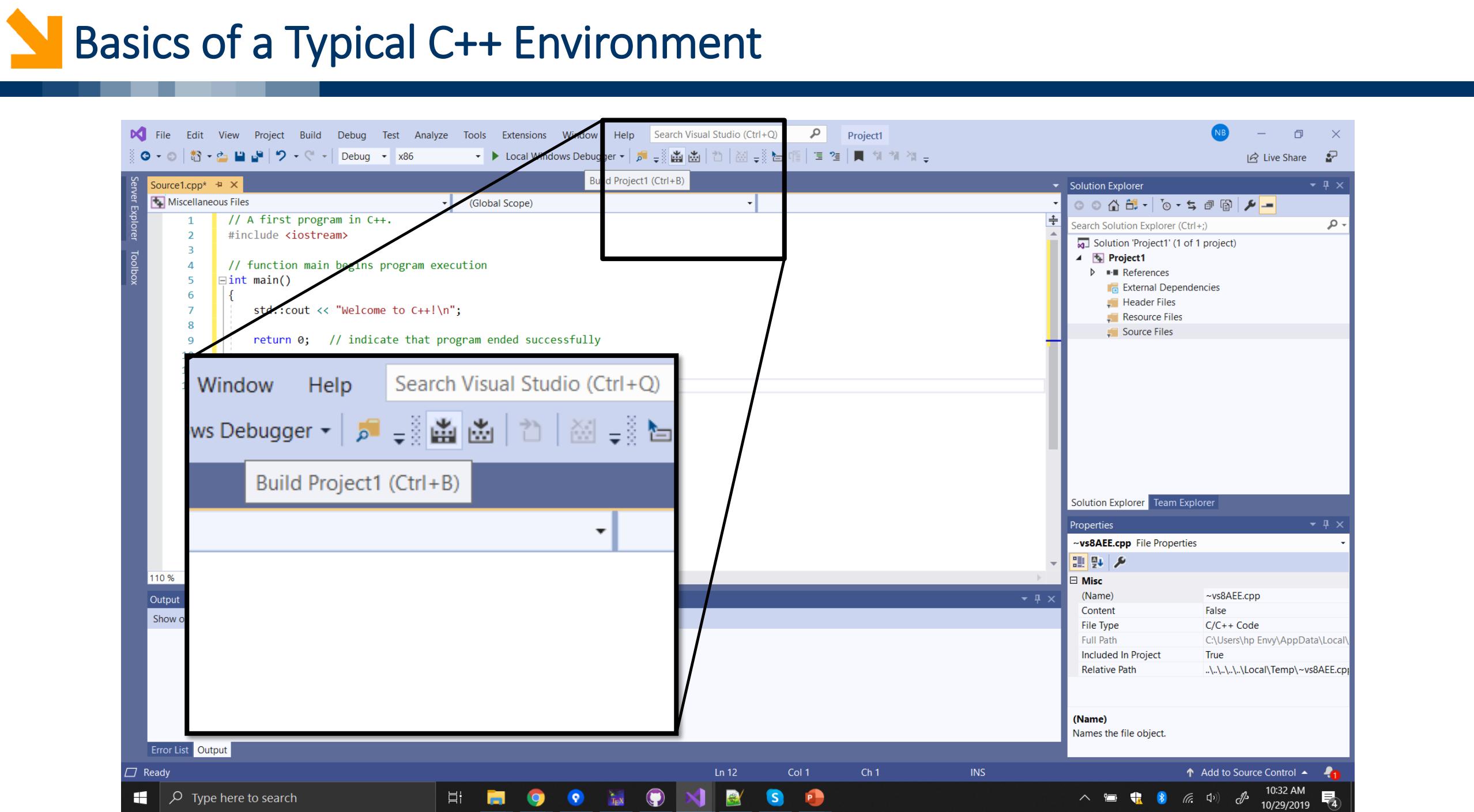
A Simple Program: Printing a Line of Text

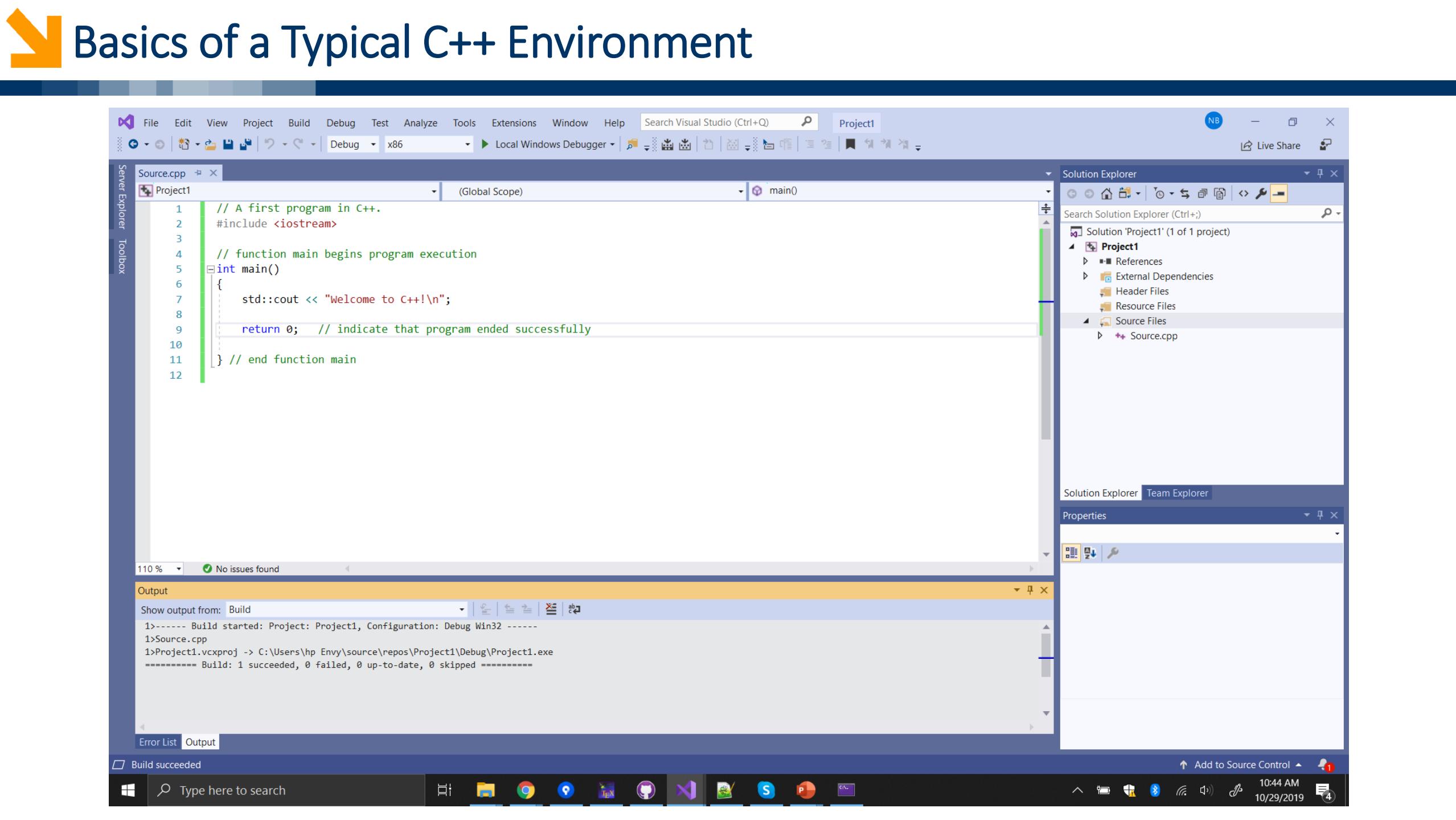
Escape Sequence	Description
\n	Newline. Position the screen cursor to the beginning of the next line.
\t	Horizontal tab. Move the screen cursor to the next tab stop.
\\"	Backslash. Used to print a backslash character.
\"	Double quote. Used to print a double quote character.



Basics of a Typical C++ Environment

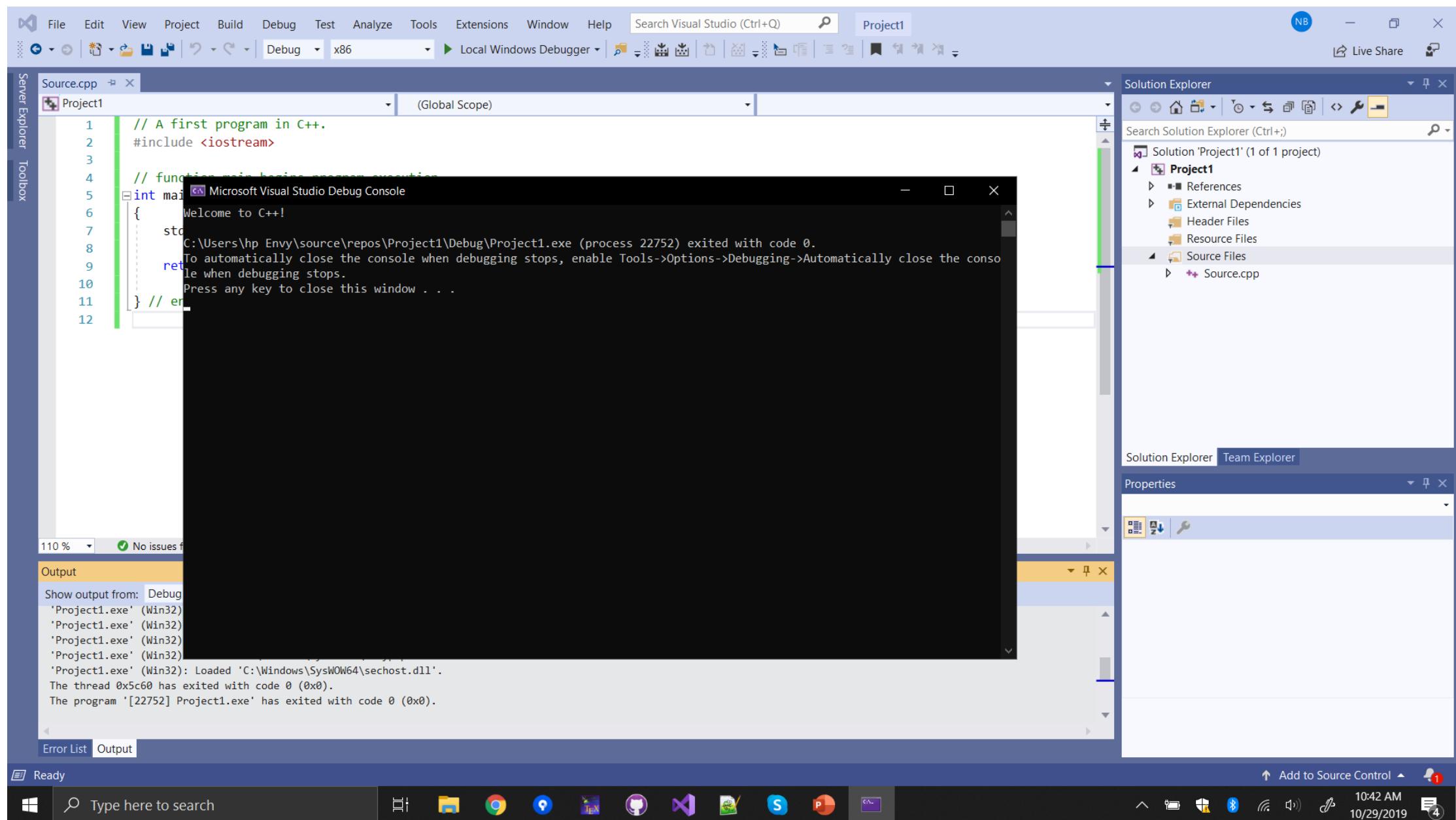




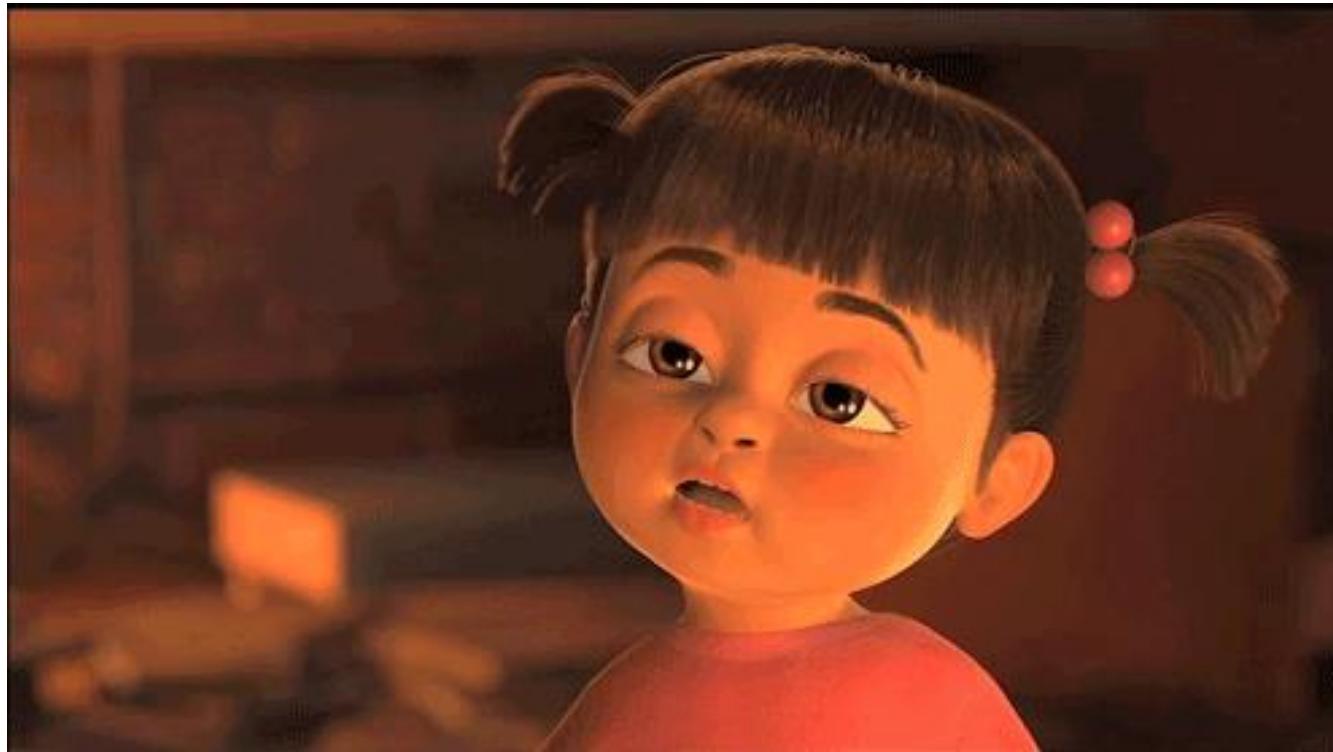




Basics of a Typical C++ Environment



Thanks a lot



If you are taking a Nap, **wake up.....Lecture Over**