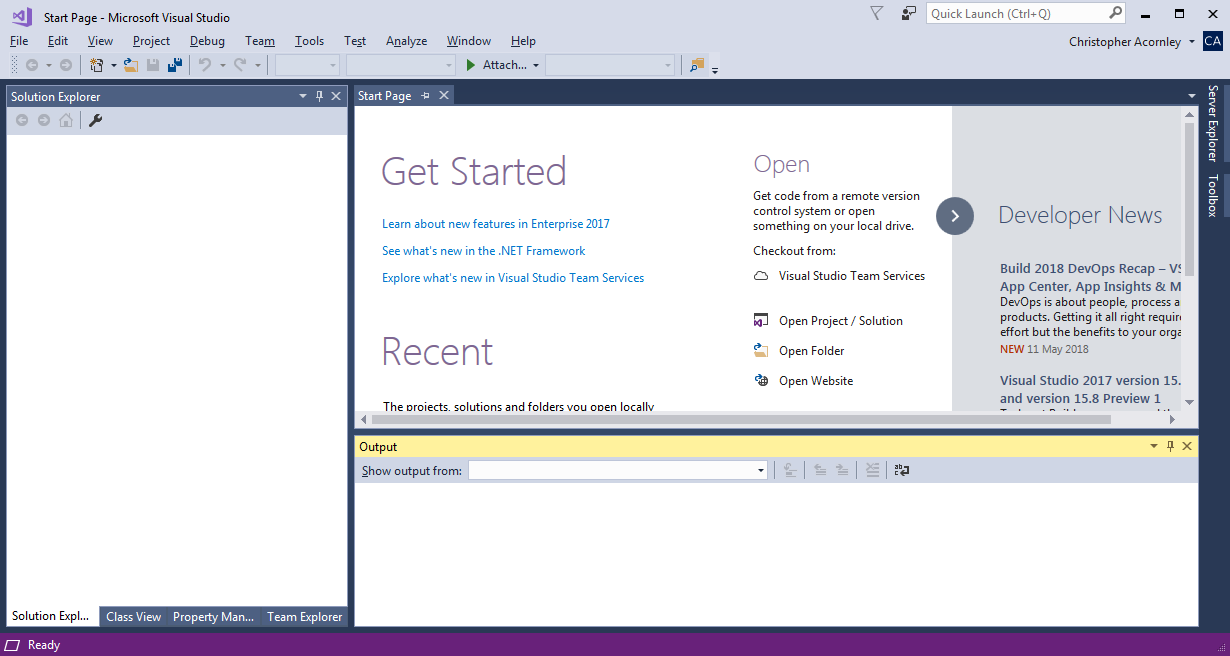
**Setting up a Visual Studio Project**

For the content in this package, we will need a C++ compiler. The most common one used by Abertay University is Microsoft Visual Studio. We recommend getting the latest version Visual Studio 2017.

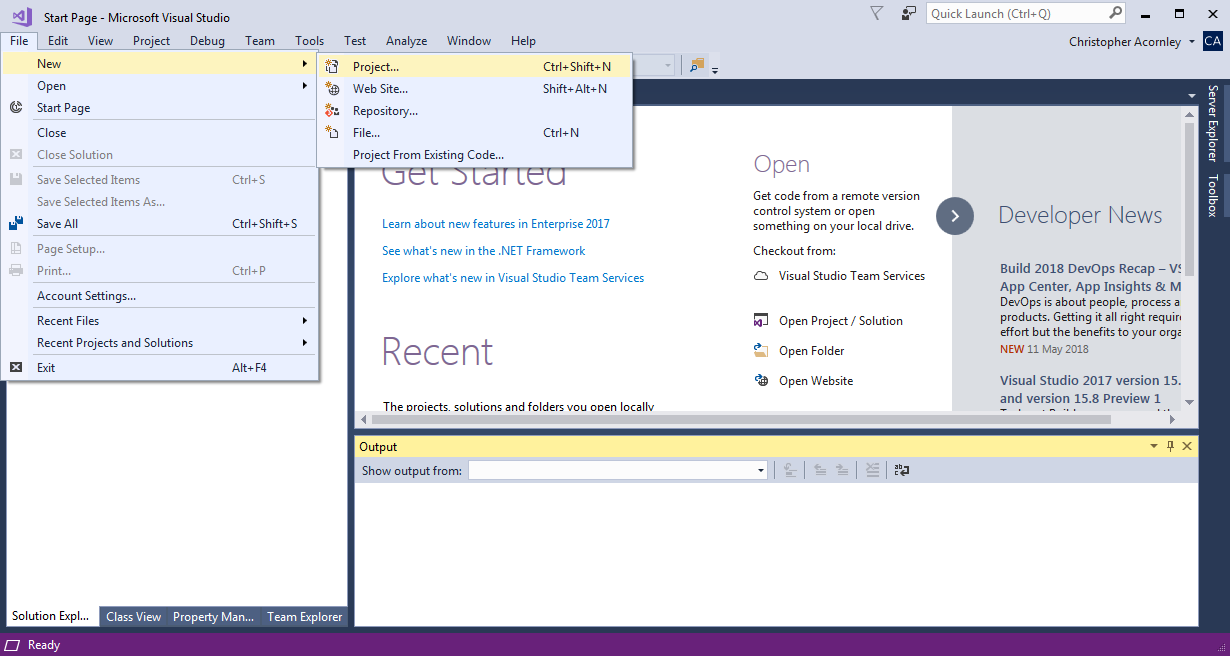
The basic interface when the program is started is the following:



From here we can create, edit and open projects and solutions. The hierarchy of files in Visual Studio means you create a single solution for each problem or application, you can then create multiple projects within that solution, with each project having their own code files (can be one or more likely more than one).

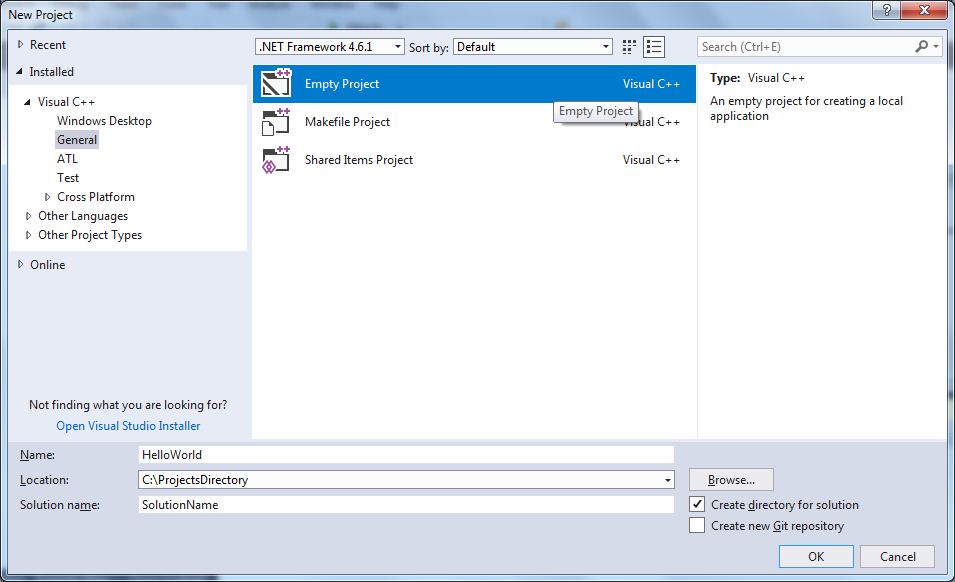
**Creating a Solution**

To create a solution, go the **File** -> **New** -> **Project**. This will create a solution for your application/problem and a single project within that solution.

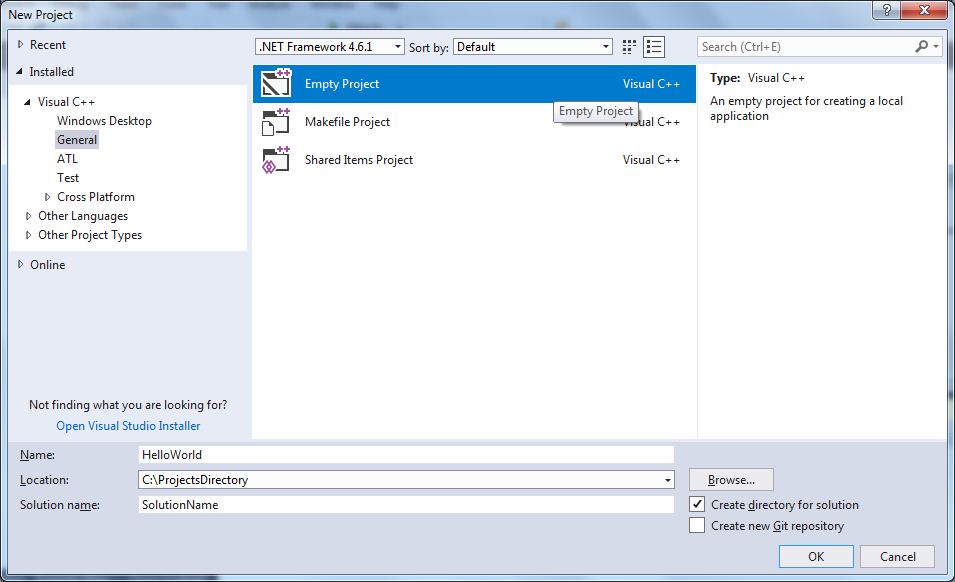


Visual Studio usually has a large number of different projects you can create, depending on the language or specifics of the project. For all examples in this package, make sure you select

**Visual C++** -> **General** -> **Empty Project**



We can now customise the project slightly by changing the Solution Name, Project Name and Save Location.



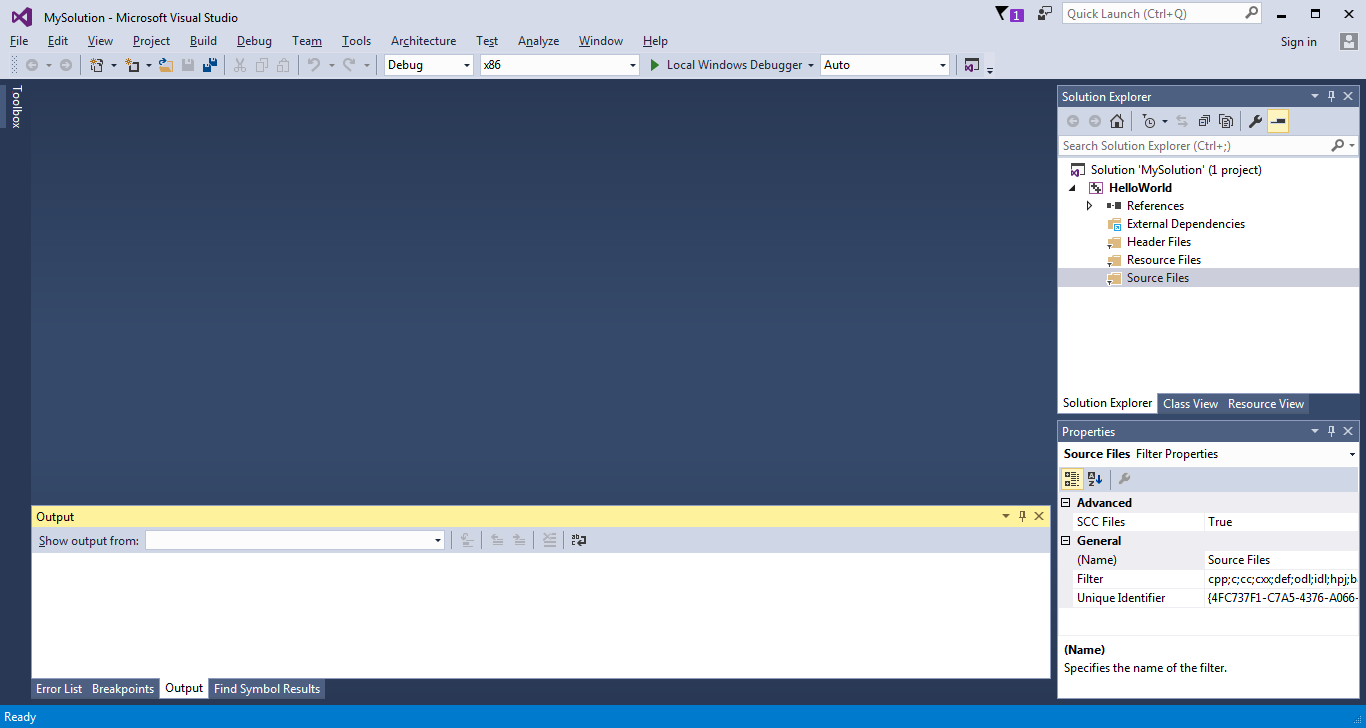
Project Name

Save Location

Solution Name

Once you are satisfied with this, click **OK**.

Once it creates the project, Visual Studio should transition to the Solution View:



**Adding Code Files**

In C++ there are two types of code files; **Headers** and **Source** files. Technically, you can put all your code in both file types and the compiler will not complain. However the intended purpose for these files is very different.

**Header** files normally contain the *Declarations* of a variable or class. If you want to create a new class, along with its own variables and functions, you would do so in a Header file (.h, .hpp, etc.)

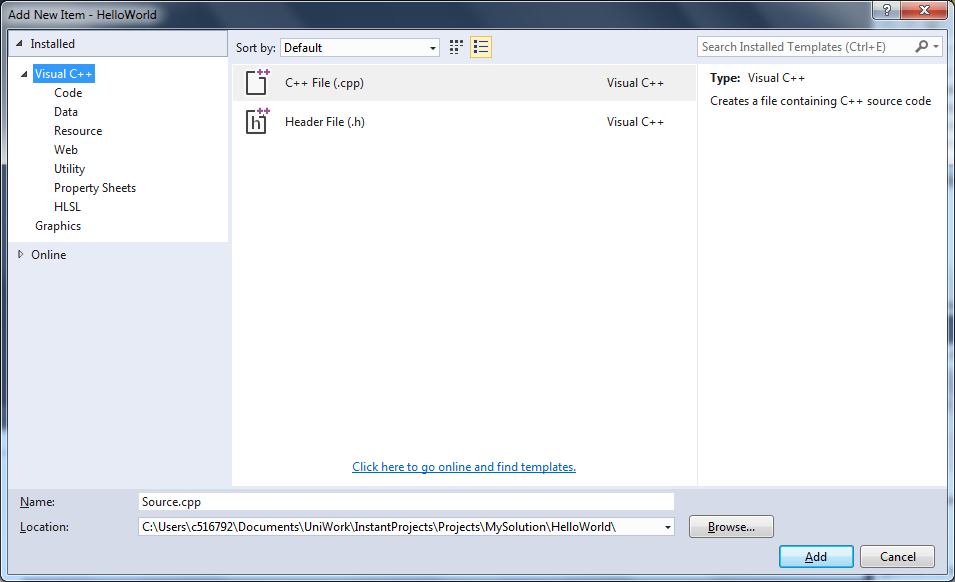
**Source** files contain the *Definitions* of your code. What it does and how it works with other aspects of your program (.c, .cpp, etc.)

An easy way to think about this is the **Header** files contain what is in your program while **Source** files tell these items how to interact.

For this example we only need a single **Source** file. Follow these steps to add a file to your project:

1. Mouse over the Solution Explorer
2. Select and Right-Click the Source Folder
3. Select **Add -> New Item**

The following dialogue box should appear.



Select **Visual C++** -> **C++ File (.cpp)**

At the bottom of the Dialogue Box, you can name the .cpp file. For now, you can leave it as the default name.

Click the **ADD** button.

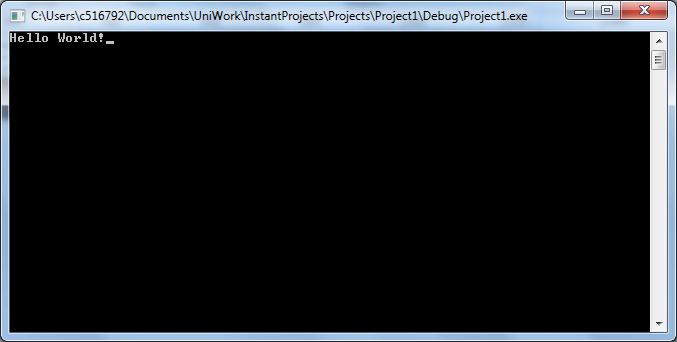
In the Solution Explorer, you should now see the file you’ve just created. If it has not done so already, double click on the file in the Solution Explorer to open it.

**Let’s Add Some Code**

Copy and paste the following code segment into your new **Source** file.



Once you have added the code, press **F5**. This will compile the program and run it.



Now that we have some C++ running, we can move on to the next lesson where we will deconstruct the HelloWorld program and see what each line of code is doing.

**Exercises**:

* 1. Create an empty Visual Studio C++ Project. Follow the steps above.

**Summary:**

* Aside from a few cases, it’s generally easier to create a new solution for each piece of work you undertake.
* To create a new solution in Visual Studio go to: **File -> New -> Project**
* The examples in this package will mainly focus on Empty Projects, though feel free to play around with the others.
* Add your code files by Right-Clicking the Source Folder in your Solution Explorer and selecting **Add -> New Item**
* The **main** function is the most important function in any **C++** program. Always include it.