

Programming for Engineers I

Creating a New Project in Visual Studio

Attique Dawood

September 25, 2012

1 Solution and Projects

In Visual Studio 2003–2010 a *project* must be created before the code can be compiled into an executable (exe). A project will produce exactly one exe therefore every project must contain at least one code/.cpp file and a `main()` function. Projects are contained in something called a *solution*. A solution may contain more than one projects. Projects *must* be part of a solution even if there is only one project in a solution. Projects or code/.cpp files cannot be compiled separately.

2 Creating a New Solution and Project

2.1 New Project Dialog

In Visual Studio navigate to **File > New > Project**. A dialog box containing additional options will appear.

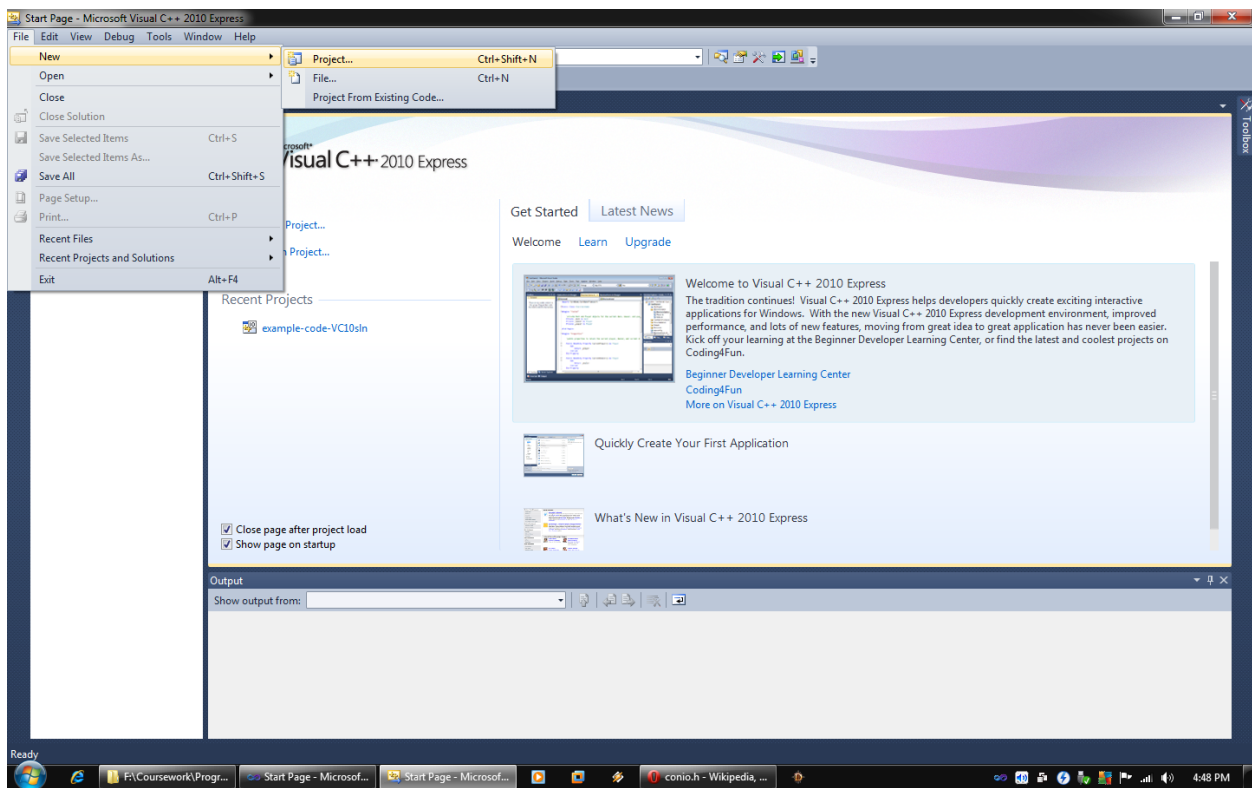


Figure 1: Creating a new project and solution

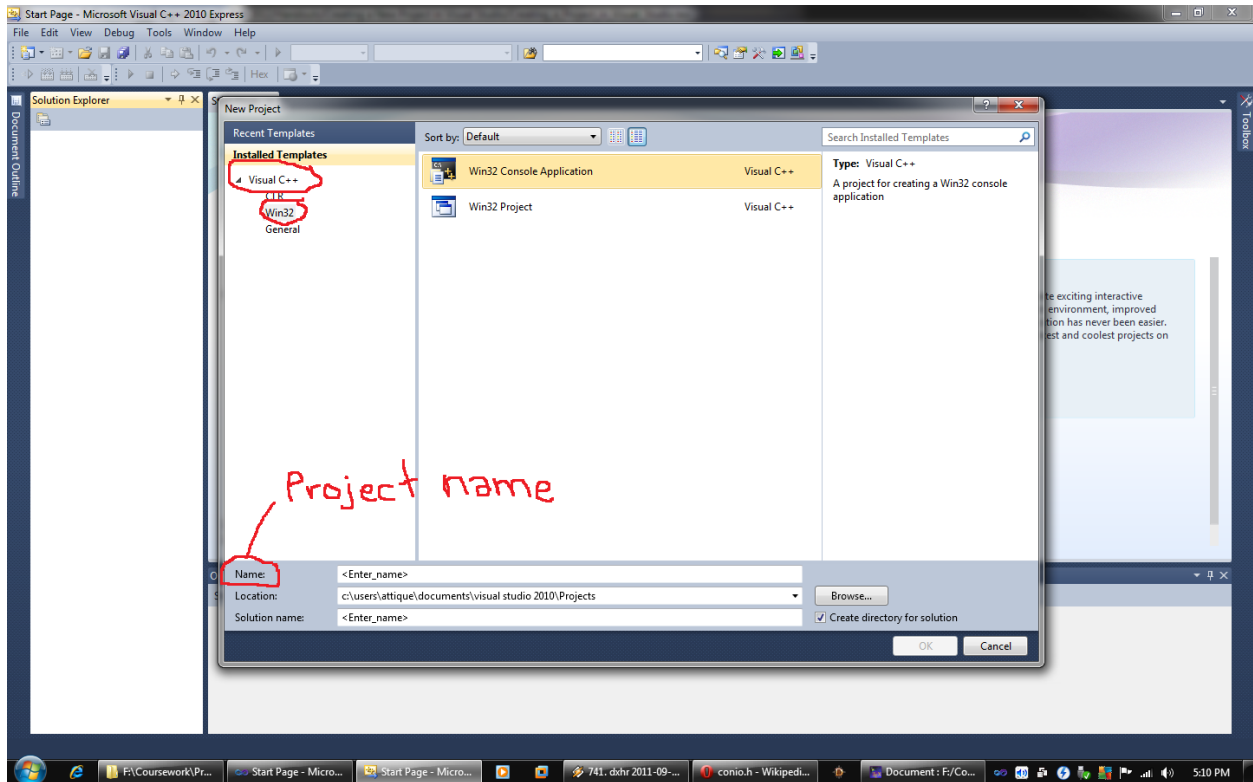


Figure 2: New Project Dialog Box

Make sure you select **Visual C++ > Win32** from the template options on the left and **Win32 Console Application** from the additional list of projects.

2.2 Naming Your Solution and Project

Most assignments require you to submit multiple programs or code/.cpp files for each exercise or question. It is a good idea to create a single solution for assignment. For each exercise or question in an assignment you can create a separate project. You have the option to build (or compile) projects in a solution at once. Also, when you open a solution file (.sln) all projects within that solution will be visible and can be managed.

For this example, we will assume that we are required to submit three programs (or exercises) from Assignment No. 2. We will name our project as **Question No 1** and the solution will be called **Assignment 2**. Notice the location or path for the solution. Your solutions are by default created in the **My Documents** folder.

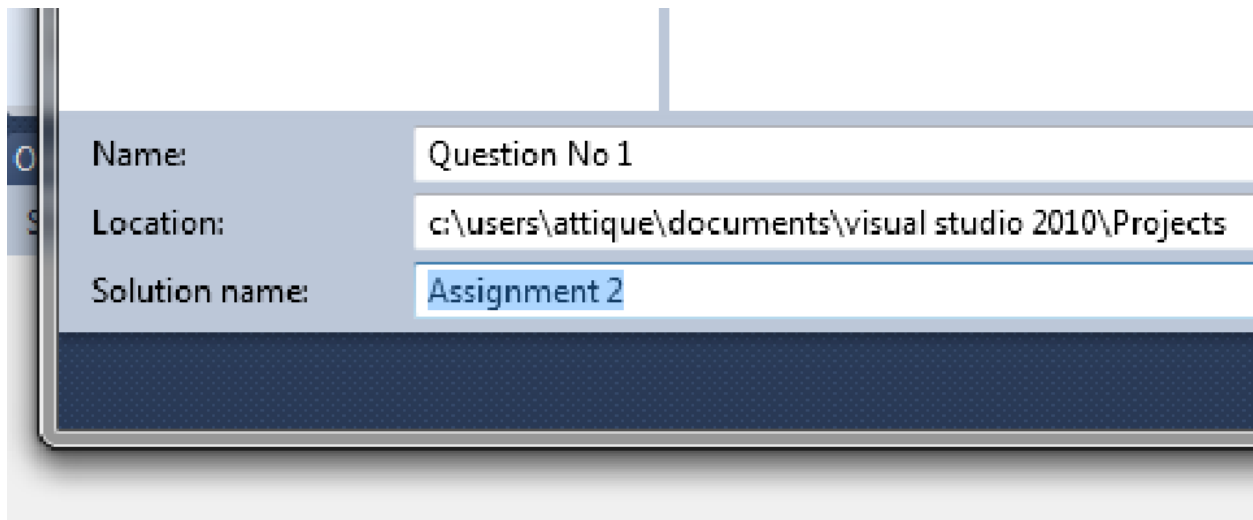


Figure 3: Project name, solution name and path/location

2.3 Application Settings

Click OK and Win32 Application Wizard will open. From here you *must* check Empty Project before proceeding. In the Application Type make sure Console Application is checked. Click Finish and your solution will be created.

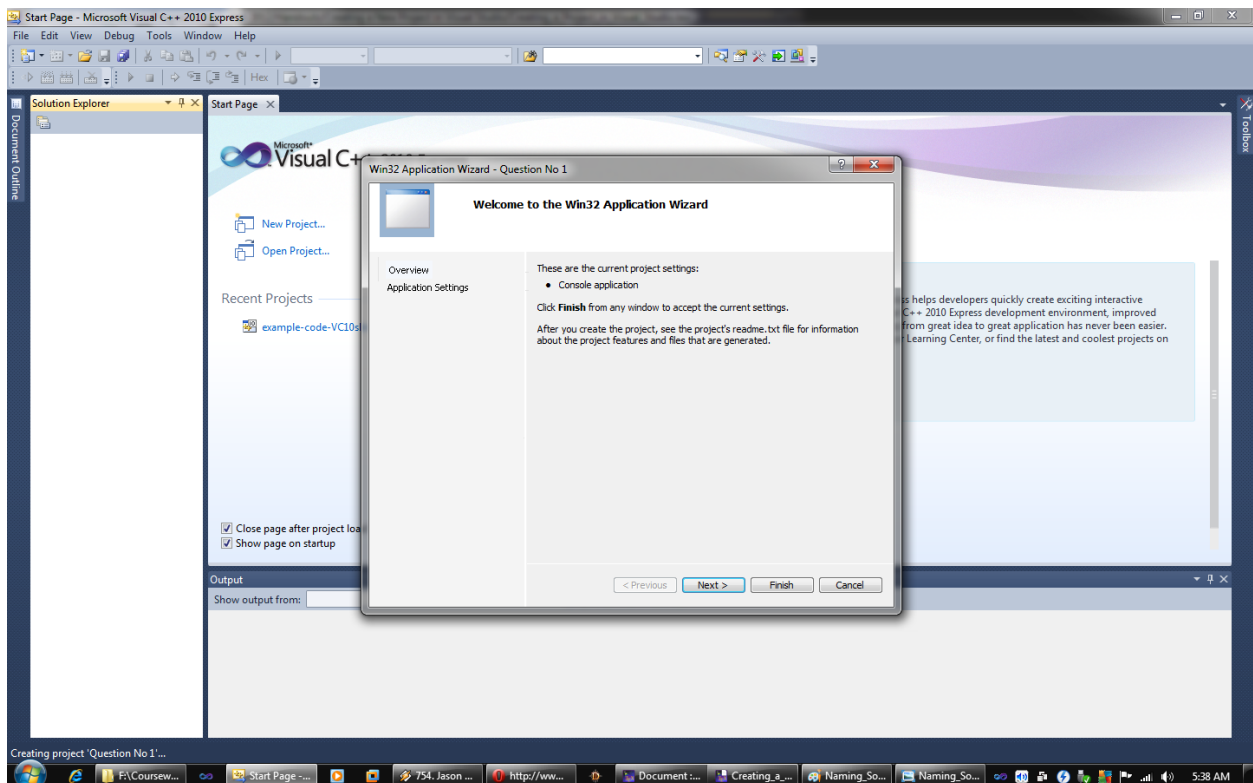


Figure 4: Win32 Application Wizard

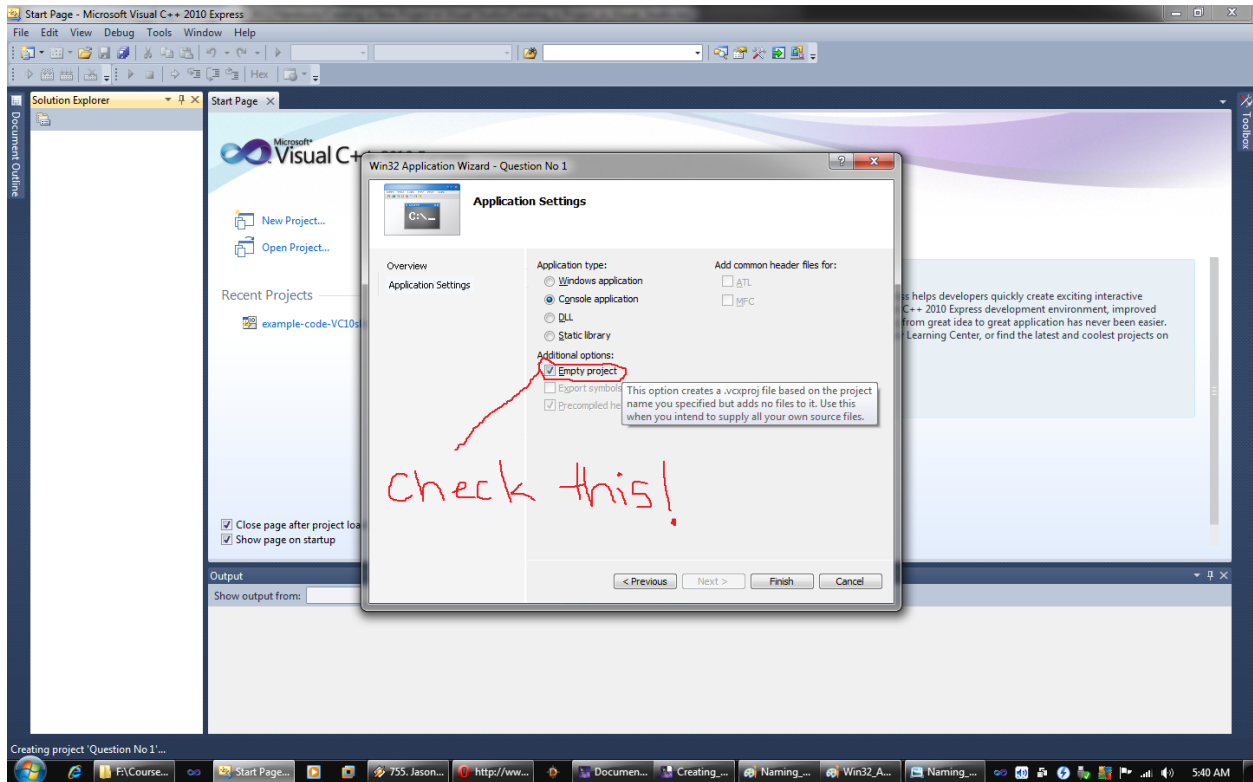


Figure 5: Creating an empty project

3 Building/Compiling and Running Your Program

3.1 Solution Explorer

After creating your solution/project make sure that **Solution Explorer** is visible. If it is hidden, enable it by navigating to **View > Other Windows > Solution Explorer**¹ or with the short-cut key **Ctrl+Alt+L**. Solution Explorer gives an overview of your solution. You can view all the projects and code files contained in your solution space and navigate between different files and projects.

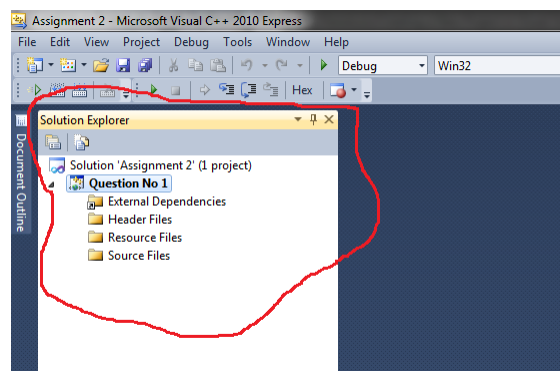


Figure 6: The Solution Explorer Window

¹The instructions may vary depending on the layout of Visual Studio. By default Visual C++ 2010 Express Edition has *Basic Layout*. You can change that to *Expert Layout* by navigating to **Tools > Settings > Expert Settings**.

3.2 Adding Files to Project

Before you start writing your code you need to add the .cpp source file to your project. Right-click the **Source Files** folder from **Solution Explorer** and **Add a New Item**. A new dialog box will appear. Make sure you select **C++ File (.cpp)** from the list of items. Name your file and click **Add**.

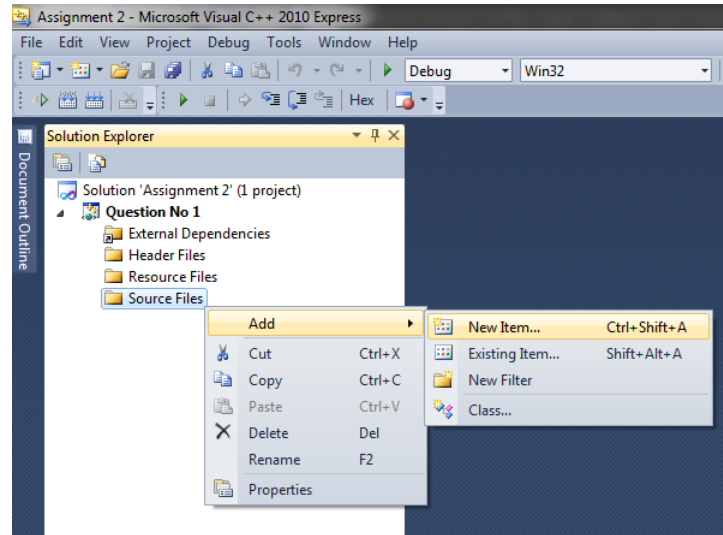


Figure 7: Adding new items to project

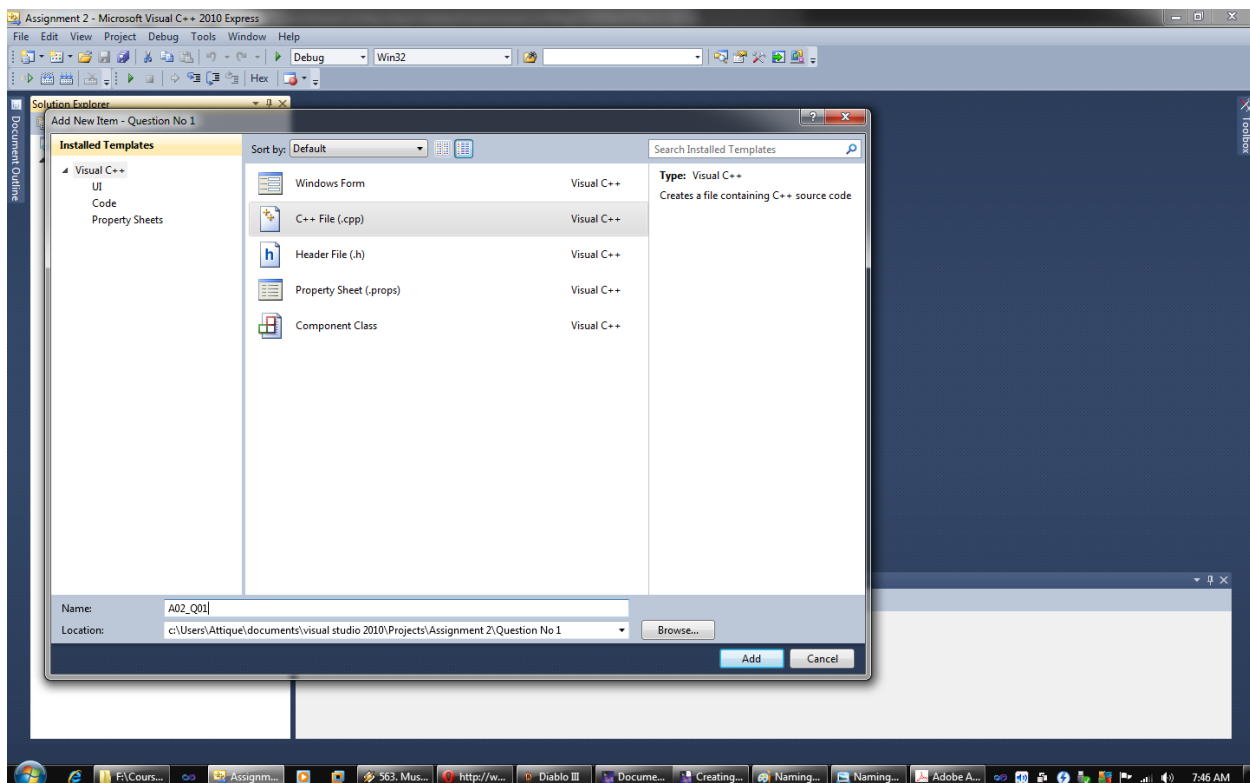


Figure 8: Adding a C++ File (.cpp)

3.3 Navigating Through Errors

The solution can be built (compiled) using the F7. If build is successful, you will get the appropriate message in the output window. Otherwise, a description of all errors will be displayed in **Output** window. You can navigate through error and warning messages with F4 key. A convenient way to navigate errors is to use the **Error List**. You can bring it up by **View > Other Windows > Error List**.

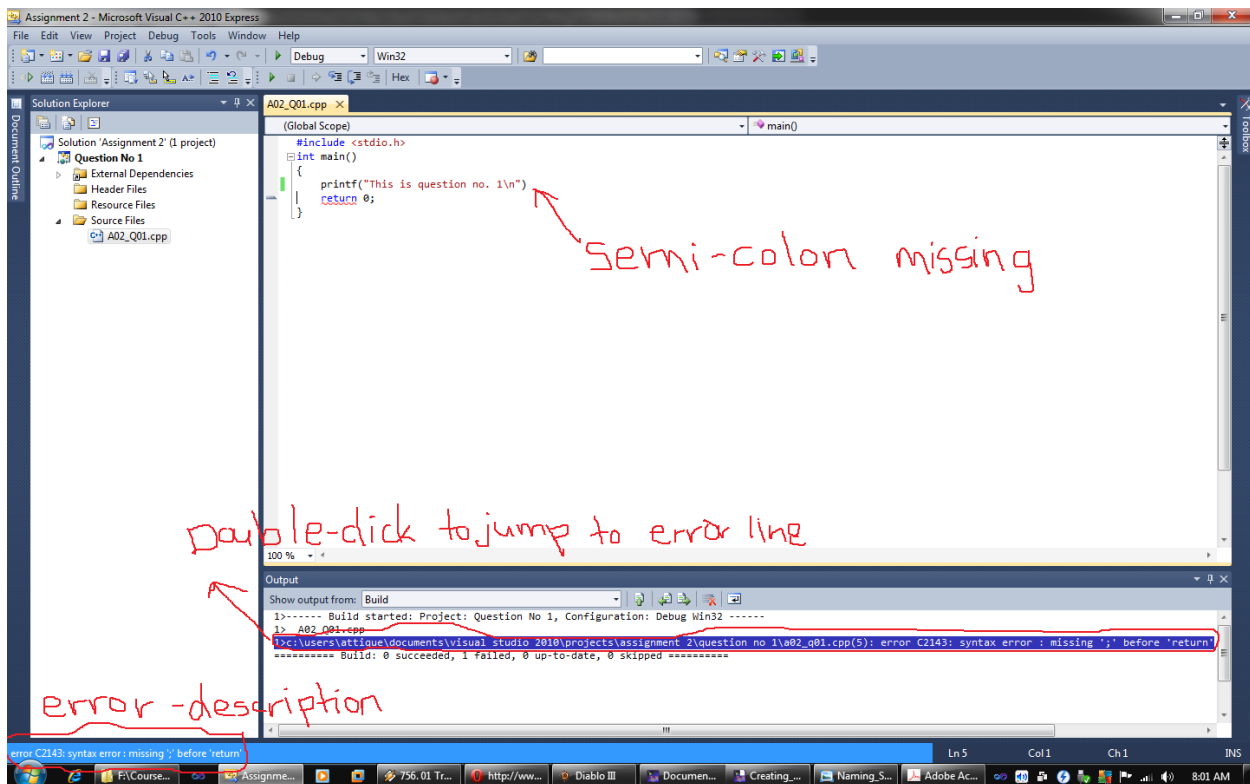


Figure 9: Error(s) during build

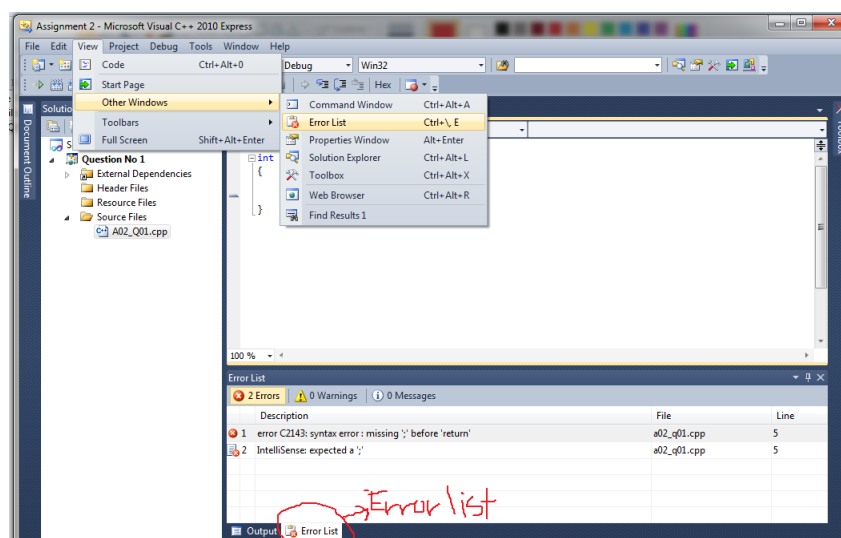


Figure 10: Displaying the Error List

3.4 Program Execution and Debugging

After a successful Build you can test your program by either running the executable (Start without Debugging, Ctrl+F5) or start debugger (F5). Debugger will stop only at any Breakpoints in the program and will exit if there aren't any breakpoints or no input directives. To single-step through your program in debug mode use the Step Over key (F10). Breakpoints can be placed by moving the cursor to desired line in your code and pressing the F9 key. Pressing F9 again will remove an already existing breakpoint.

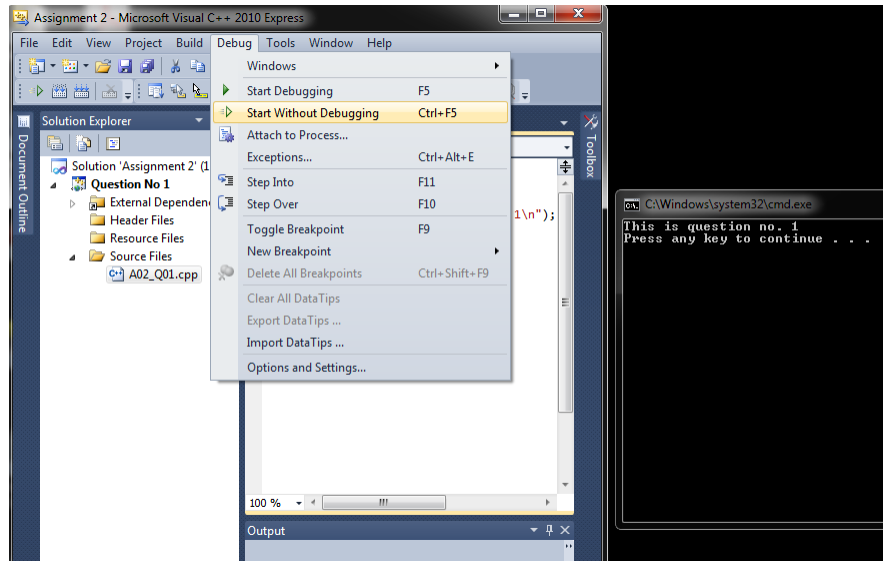


Figure 11: Program execution

4 Adding New Projects to an Existing Solution

We're now going to add a new project to our solution and name it **Question No 2**. In the Solution Explorer window right-click on solution name and add a new project. Follow the steps from Section 2 for creating a new empty project of type Win32 Console Application.

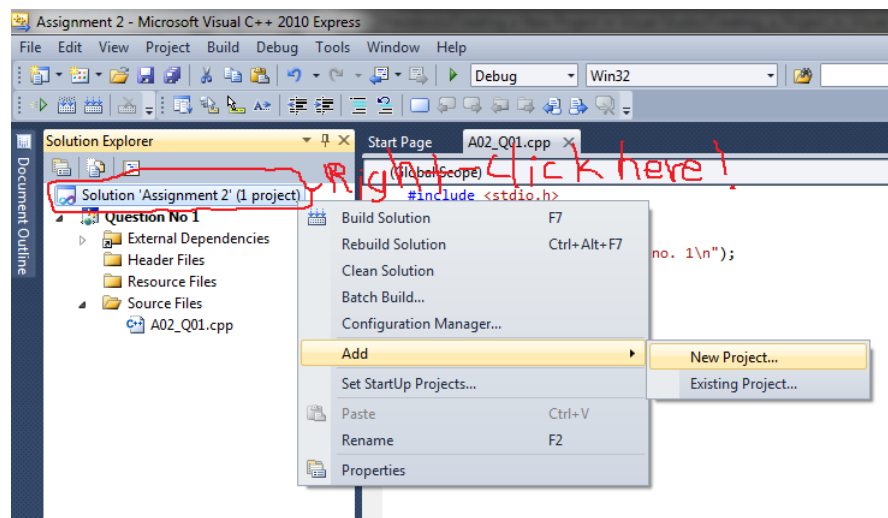


Figure 12: Adding new projects to solution

Add a C++ source file (.cpp) to your project following the steps in Section 3.2. We'll name our file A02_Q02.cpp here. Before you can compile and run your new project you must set it as **Active Project**. This can be done by right-clicking the project name (Question No 2 in this case) and selecting **Set as StartUp Project** from the drop down menu. Active project name will be in **bold**. Figure 14 shows the complete solution space with output window and **Question No 2** selected as active project.

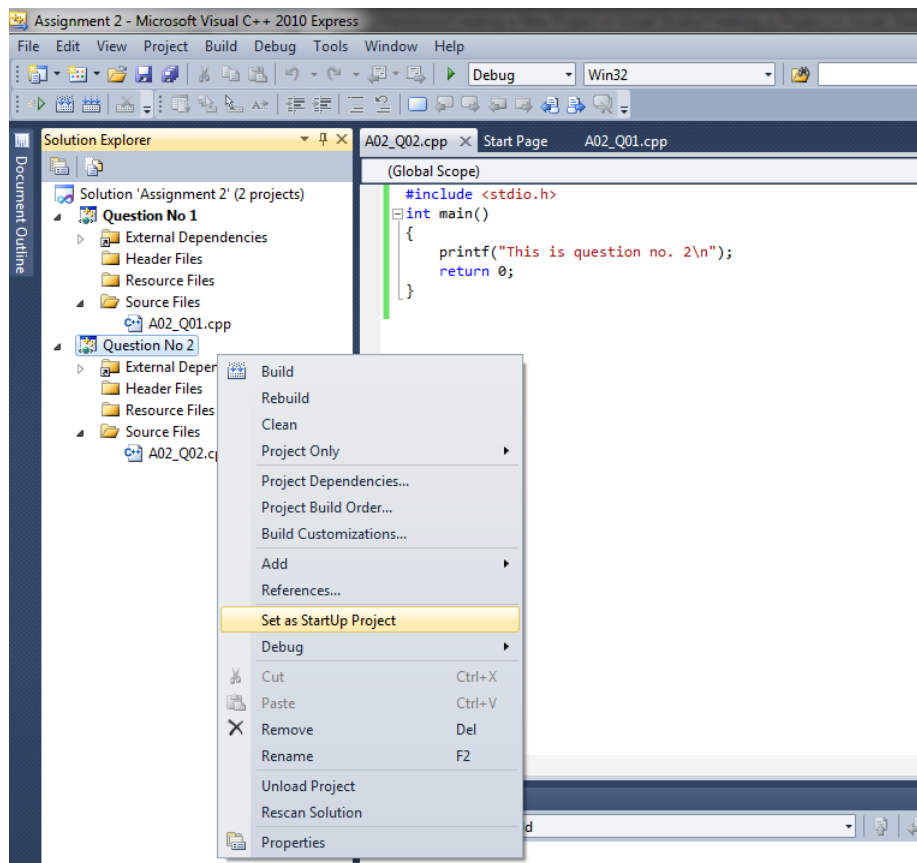


Figure 13: Setting active project

5 Transferring Solutions

In order to transfer your work you need to copy the whole solution folder located in the **My Documents** folder. The solution folder contains a [Solution Name].sln file. You can open your solution by double-clicking on this file, provided that Visual Studio is installed on that particular computer.

The solution folder also contains separate folders for each project in the solution. The C++ code (.cpp) files are located in their respective project folders. In order to conserve space you can delete any folders named **Debug** in the solution or project folders. You can also delete **ipch** folders or any **.sdf** files. The **Debug** folder in solution folder contains executables from each project.

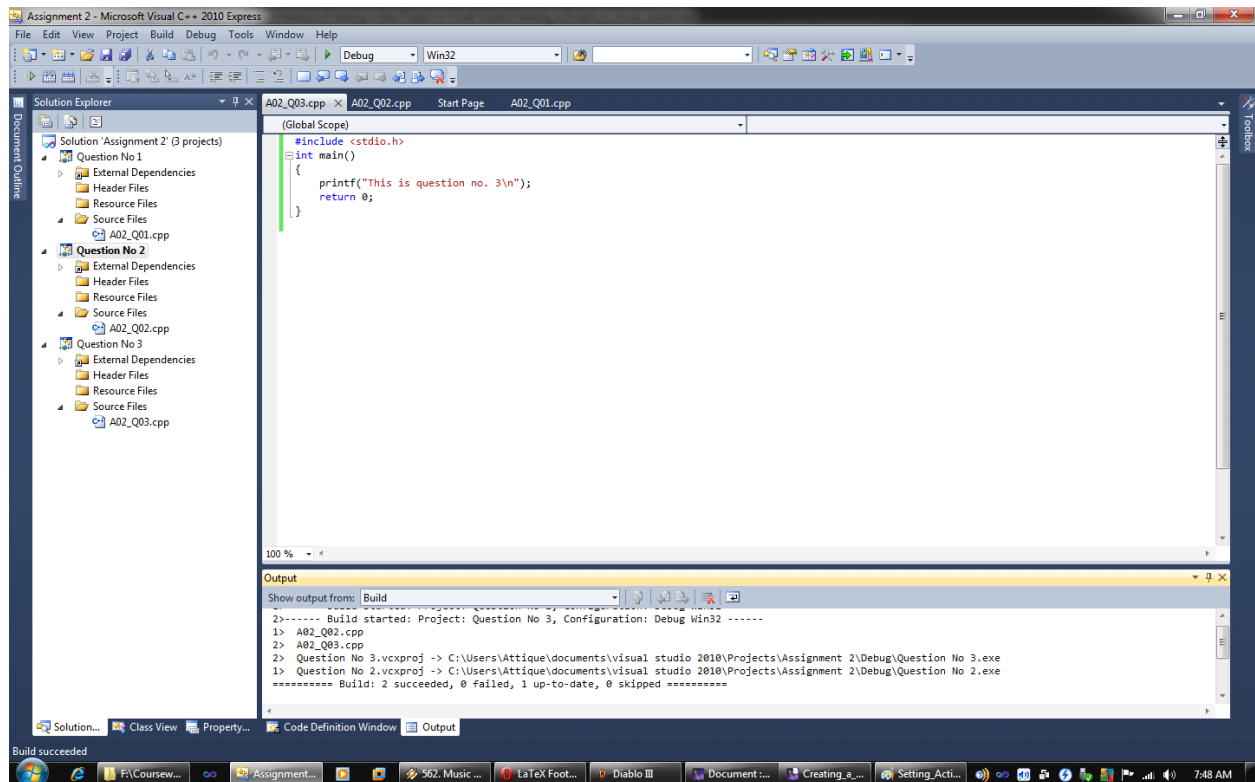


Figure 14: Final solution space

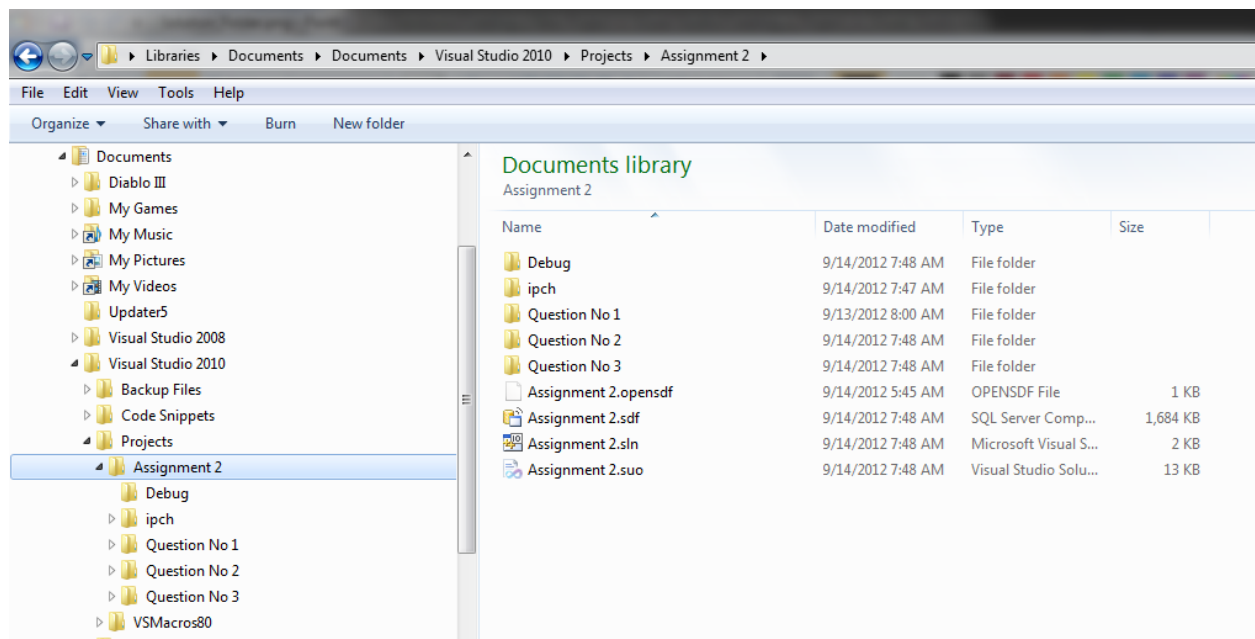


Figure 15: Solution folder