**---Things to do after setup:** If you use a microsoft account and log in, settings will be imported.

-Delete these two registiries to get rid of “Open in visual studio” context menu option.  
  
 HKEY\_CLASSES\_ROOT\Directory\Background\shell\AnyCode  
  
 HKEY\_CLASSES\_ROOT\Directory\shell\AnyCode

-Show line numbers, tools -> options -> text editor -> all languages -> general  
  
-Set import export location as “C:\Computer Engineering\6- C\3- Notes\IDE” from tools -> options -> environment -> import and export settings.  
  
-You can go to tools -> options -> environment -> keyboard and type “File. Close”, select text editor, press Ctrl + W and click assign to set it as close this tab.

-Install the “format on save” extension. Either use the file on “C:\Computer Engineering\6- C\3- Notes\IDE” or use the link.

<https://marketplace.visualstudio.com/items?itemName=mynkow.FormatdocumentonSave>

-You can use middle click to close a tab. You can right click a tab to say close all but this.

**-Source Code:**Ctrl + LMB: Use this function of IDEs to learn what kinds of methods classes/objects have. Use it to learn more about those methods(arguments, return type).

**-Assembly Code**

Put a breaking point. Debug. When the breakpoint hits, right click and select go to disassembler.

Or, you can right click your project, go to properties, C/C++, output files, assembler output, assembly only listing. Clean solution, build solution. Go to project’s folder, Debug, .asm files will be there.

* **Debugging:**

-You can add breakpoints to your code by pressing F9 after selecting a line.  
-At the breakpoint tab at the bottom you can click on “Show Columns” to disable or enable what you want to see about your breakpoints.  
-We can right click the breakpoint on the circle or at the tab to click edit label and put a label to it that explains its purpose.  
-We can right click a breakpoint to add condition so it only breaks when condition is met. Also we can make it so it only breaks when it reaches a certain hit count(s).  
-We can right click to set a when hit message.  
-If you dont want to delete the breakpoint completely and just want to disable it, you can uncheck the point.  
-You can use F10(step in) to debug your code step by step.  
-You can use Ctrl+F10 to run to cursor. If you want to skip over to that part and want to test that part. You dont have to use this function just to skip over some parts of the code. You can use it to see when does execution go into a line. You can run to cursor there to see the next time when the execution reaches there.  
-You can go to breakpoint tab in the bottom after starting a debugging and click the “Go To Disassembly” button to see assembly translation of your code. Every single line of your code’s assembly counterpart will be shown.  
-The localswindow shows everything which are currently in scope. These variables are also automatically detect by the Visual Studio debugger during the debugging. Visual Studio determines which objects or variables are important for the current code statement and based on that, it lists down the autos variable.   
-Open locals window to see states of variables(what values do they have and if they are in scope(stack frame) or not.   
-Testing with different values: You can change values of variables during debugging by double clicking their value.  
-You can use immediate window to test a declaration of a class, struct or a union type or a function that is inside one of these.(class)  
-You can use call stack and parralel watch(open it from debugger-window) to see how recursive functions call and how their parameters change as stack depth goes forward.  
  
About error and debugging in C/C++: <https://www.youtube.com/watch?v=Ao2YgvoJEKI>  
  
Edit and continue: While you are debugging, you can change the code and step over again to continue debugging with the new code.

-You can use visualizer to see xml, html codes or strings etc. Just hover over declaration of the data type and a magnifier will appear. Click on the magnifier to reach visualizer.  
  
-You can chose multiple lines and tab, shift tab them at the same time.

* **Ln Col Ch(Line, Column, Character position)**

-"Ch" is the actual "character position" of your cursor, while "Col" is the visible column.  For instance, you could have 3 lines of code, all starting at the beginning of Area B.  One might report being at Col 12, Ch 12 because it's preceded by 11 spaces.  The next one might report being at Col 12, Ch 9 (preceded by 1 tab plus 7 spaces).  And the third might report at Col 12, Ch 6 (preceded by 2 tabs plus 3 spaces). (assuming you're using Visual Studio's default setting of Tabs equal 4 Spaces)

* **Save, Building and Run**

-Right click a source code from solution explorer and select compile. You can to save by right

clicking on them from its tab. Or you can use building.

Ctrl + S: Save

Ctrl + Shift + B : Build

F5 : Debug

Ctrl + F5 : Run

* **Opening more than one code/project**

-Right click VS on taskbar and open another instance.

-Or you can create multiple source files in a single project but you cant have multiple main functions in a single project so you have to comment one out.

-Or you can open a project in one instance and open a non project .cpp file in another instance.

* **Closing a project:**  In order to delete, , rename, move: file -> close solution
* **Crt Security:** Right click project at solution explorer, from C/C++ tab,

-Uncheck the “add SDL box when you are opening a new project.

-From Preprocessor tab, add "\_CRT\_SECURE\_NO\_WARNINGS" to preprocessor definitions.

* **Traverse**

Alt + Up/Down arrow: Current line is swapped with line above or below.

Ctrl + Enter: put the current line below and put a blank line to the current line.

Ctrl + X: Cut line.

Tab: Indent selected lines.

Shift + Tab: Deindent selected lines.

You can select a word by double clicking on it or you can select a row by triple clicking on it or a word on it(you can select an empty row like this too).