**-Auto completion:** 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). You can click on something and press F12 to see its source code or look it up on reference to learn even more about it.

* **Debugging:**

- Breakpoints are used to pause execution of program when you run it.(they are not important when you are debugging with step into etc.)

-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 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 need to uncheck the box in breakpoint tab.

-You can use F10(step in) to debug your code step by step.

-Step into means keep running the code statement by statement. If we are at a function call and we step into that will take us inside function and we will keep running statement by statement. If we were to say step over when we were at the line that calls the function, computer would execute the function and get us to the next statement after it. Step out runs the code until you get out of the function you are in right now.

-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 to code when you first run it to get to the place you want. 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 it exection reachers there.

-You can go to breakpoint tab in the bottom after starting a debugging to see assembly translation of your code. Every single line of your code’s assembly counterpart will be shown.

-To open locals window to see states of variables(what values do they have and if they are in scope(stack frame) or not. An interesting thing you will see is variables are declares instantly when you get into a function. You dont need to declare or initialzie them. Your initialization just gives it the first non random value it will have.

-Testing with different values: you can just hover

-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>

-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 : Run

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

-birden fazla kod dosyasına bakmak istiyorsan ya birden fazla instance açman gerekiyor(taskbar'da sağ tıklayıp recent'dan bi tane açarak)

-yada aynı projede bi c source file daha yaratman gerekiyo. Ama ayni projede iki main fonksiyonu olamayacagi icin digerlerini commentler ile(/\* \*/ ) disable etmen gerekiyor.

-yada bir proje, birde proje olmayan .c dosyası açarsın.

* **Closing a project:**

o projeyi dosyada silebilme, isim değiştirebilme, başka yere taşıma yapabilme için) file -> close solution

* **Main Fonksiyonu:** Bir projede birden çok main fonksiyonu olamaz.
* **Crt Security:** Projeye solution explorerda sağ tıkla, c/c++ tabından

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

- preprocessor alt tabındaki preprocessor definitions'a şunu ekle "\_CRT\_SECURE\_NO\_WARNINGS"

* **Visual Studio Options:**

-satır sayılarını gösterme, tools-options-text editor- all languages-general

-Set import export location as c-notes-ide notes from tools options environment import and export setings.

-You can use middle click to close a tab. You can right click a tab to say close all but this. You can go to tools-options-envoronment-keyboard and type File.Close , select text editor and press ctrl+w to set it as close this tab.

-dark mode:

* You can press f12 to see the **source code** of the selected thing.

-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).

* **Auto and Local:** The **locals**window 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.
* **Traverse**

-Netbeans de shift + alt + alt ust ok current satiri ust, altla swap yapiyordu. Buradaalt + ust al ok.

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