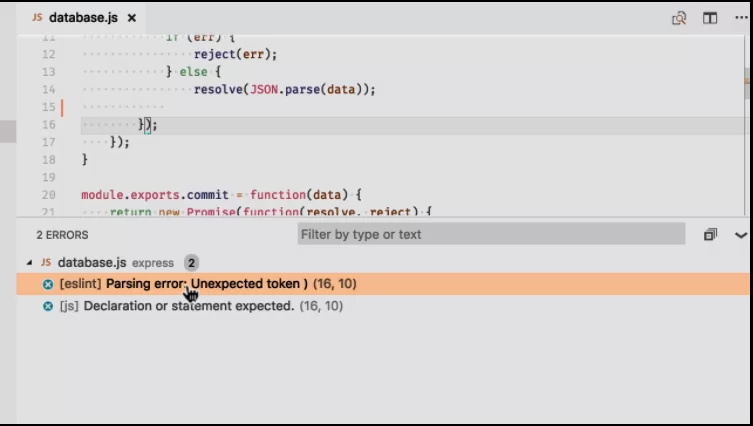
Visual Studio Code Features

The first feature visual studio code has is the status bar, the status bar tells you errors and whether or not it has errors. This helpful bar will pop up whenever you start a program, starting a program is done with NodeJS, NodeJS is a helpful tool for programmers and is used frequently. It is one of the best features of visual studio code, and is used very often in conjunction with the other features on this list. It will tell you the exact line of code, and space where your error is if you have one. The line visual studio code has you go to is messed up, and it expects you to fix it before moving on, once you do, it will tell you the next error in order.



The order it will be in is chronological and will abide by those rules always. The visual studio code status bar also tells you if the program is ok to run, and will run if done correctly. You can also filter your problems by errors or warnings, warnings haven’t happened yet and errors have. The status bar is a very helpful tool for when you want to prevent errors from happening very early on, this is very helpful when you are coding a big project and don’t want to stop.



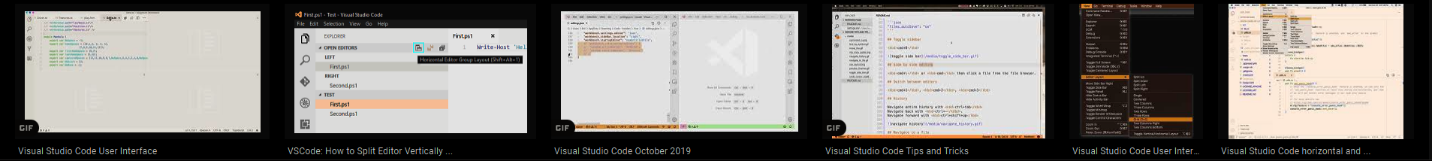
A shortcut allows for a feature to be accessed quicker without manually clicking, and this applies to the status bar also. The status bar applies as a failsafe for errors and applies as so.



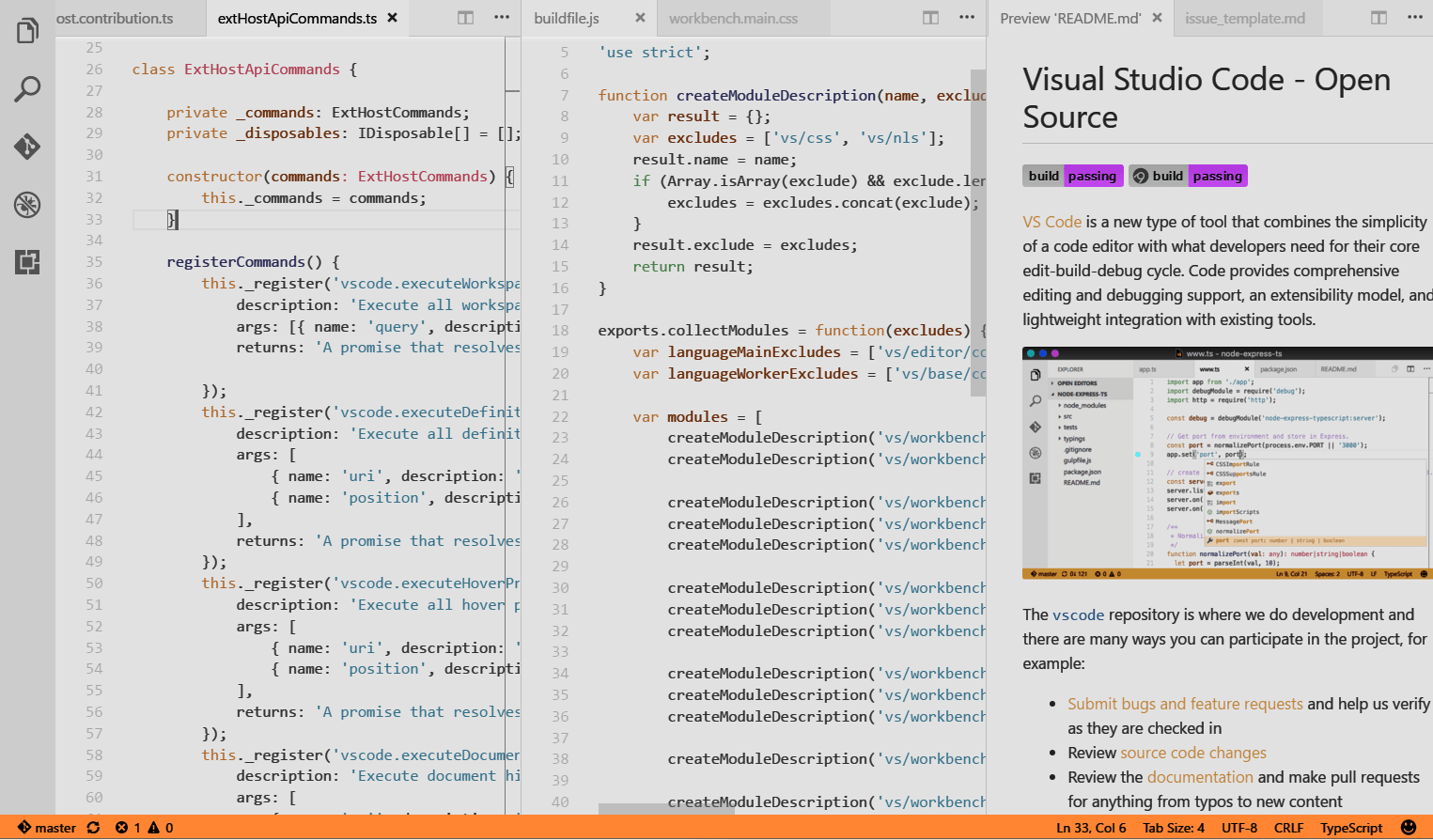
the work environment this helps create allows for errors without painful scrutinizing to find each and every single error that you did throughout your program, and unlike other program writers, the status bar in visual studio code will let you mess up. It also has a keyboard shortcut to allow for quicker access. This shortcut is ctrl + shift + m, and allows you to see the status bar without manually clicking where the status bar is. This is because it is user friendly accessible and wants to appear as so in it’s work. Visual studio code will help you wherever it can. This is because it aims to be a user-friendly tool in usage. Visual studio code also uses the status bar to do its best work.

Split view is the second feature, this is a very special feature put into visual studio code to help out with multiple parts of a program at once. Being on multiple parts of a program can be confusing, and split view helps to minimize mistakes when working on multiple things at once. This is because programmers have to multitask and your job will expect you to do so without hesitation. Split view emphasizes on more work being done at the same time. 

When you do split view, you are allowing yourself to get more work done easier than you normally would, this is because work just gets done easier in programming when you do it on multiple websites.



This feature is essential to dual-working because of its properties. This is because split view works much like a second monitor on the same screen. The keybind to do this by keyboard is ctrl+/ for easier access.



You can also open to the side using ctrl + enter, and you can open it manually using the split editor using the upper right corner of the editor. There is another feature where you can drag files from your file explorer using your file explorer. Dragging files from the file explorer is easy and requires little to no effort when done. This is because the windows file system is easy to use. Windows designed this system with the average person in mind. The only other option is to set the split view to the bottom and work from there.

The final feature is the mini-map. This feature is the most useful out of the three, other than being a slice of life feature. During programming, you may be lost as to where you are, this is concerning because you can lose time and therefore in a job environment lose money.



The mini-map is used to also find lines of code that you might not remember the location of. It uses your entire program to see the flesh of whatever you are working on. This is also used to find issues, much like the status bar.



You can also move the mini map by setting it to the left side of your screen. Or you can disable the mini-map altogether by setting it in the user or workspace. Setting these in the user or workspace can be done with editor.minimap.side": "left", and for disabling the mini map editor.minimap.enabled": false.

