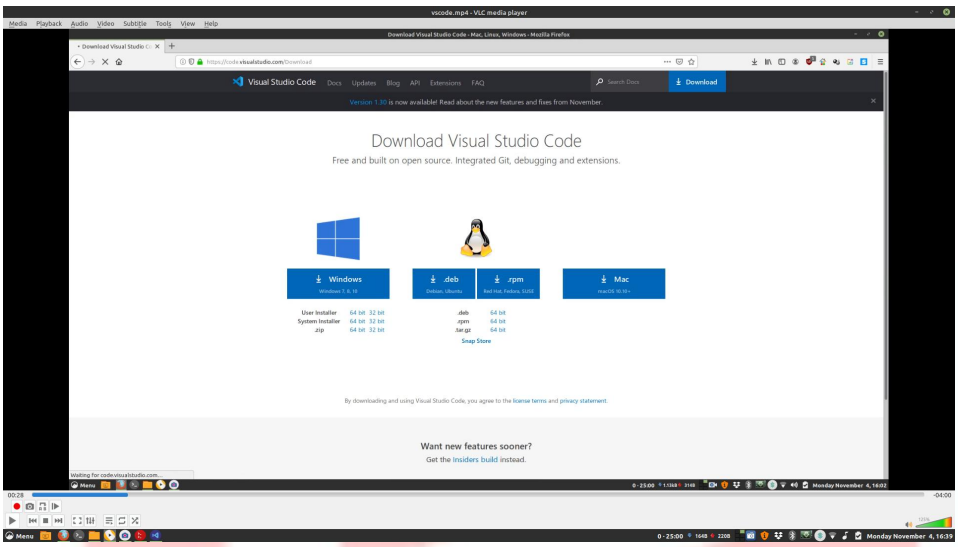
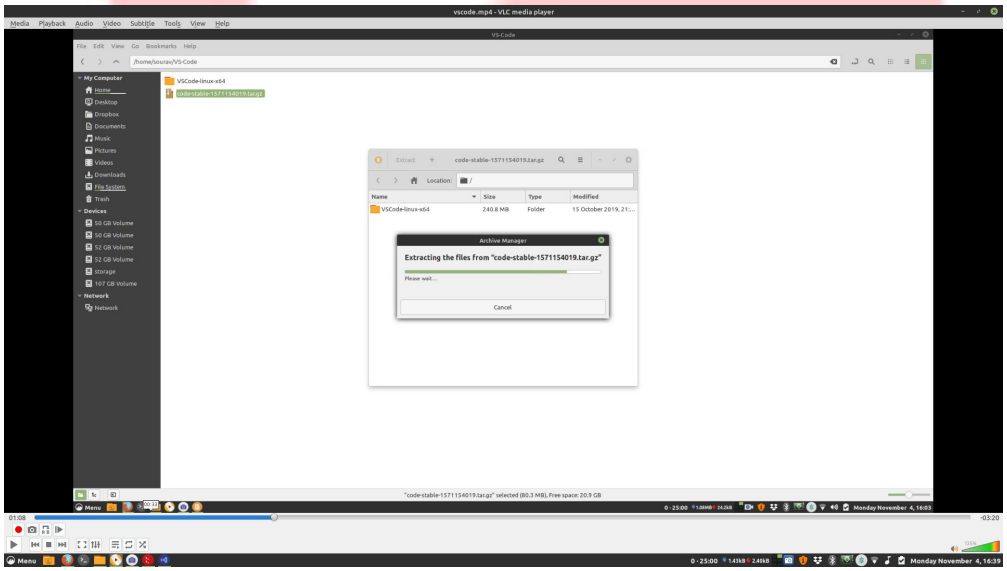
Task 1: Basics of VS Code

1. Download VS Code Portable

Download VS Code Portable (Linux 64 Bit) from here



Extract the .tar.gz file to install VS Code in your machine.

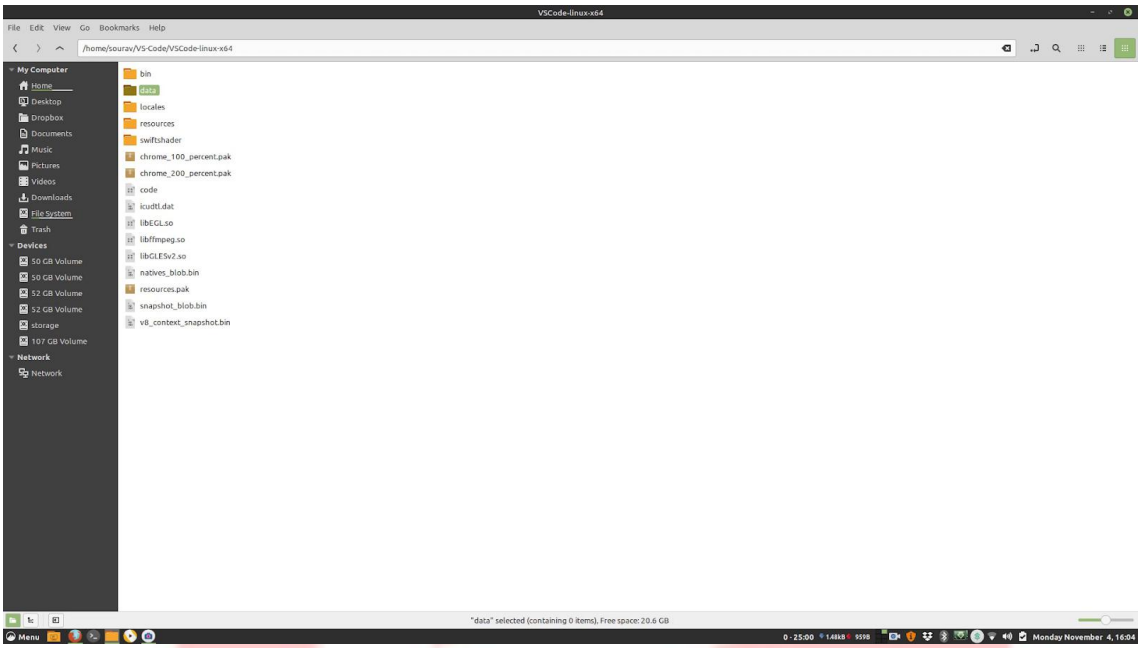


2. Enable Portable mode

Visual Studio Code needs to be run in Portable Mode. This mode enables all data created and maintained by VS Code to live near itself, so it can be moved around across environments.

After unzipping the VS Code download, simply create a “data” folder within VS Code's folder:

|- VSCode-win32-x64-1.25.0-insider
| |- Code.exe (or code executable)
| |- data
| |- …

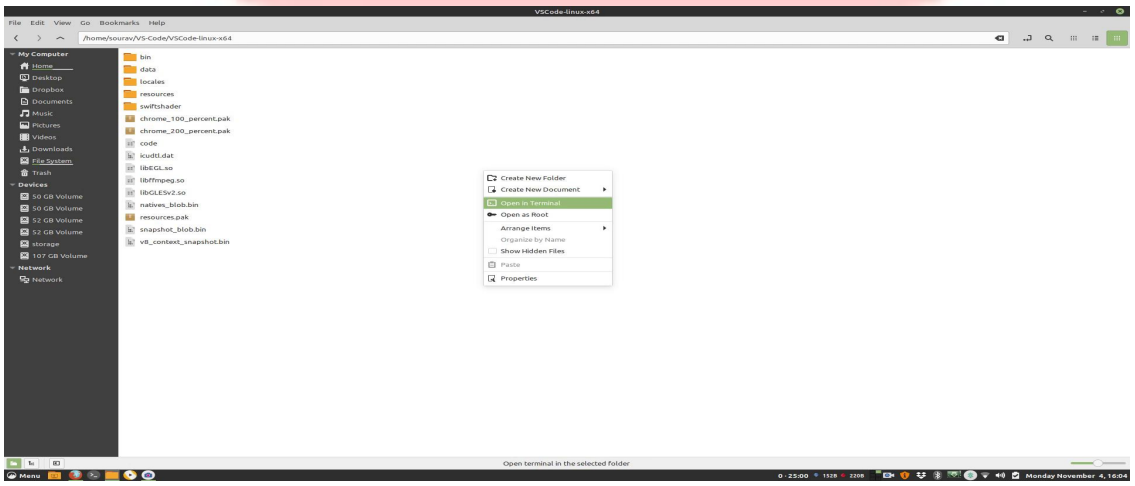


From then on, that folder will be used to contain all Code data, including session state, preferences, extensions, etc.

The data folder can be moved to other VS Code installations. This is useful for updating your portable Code version: simply move the data folder to a newer extracted version of VS Code.

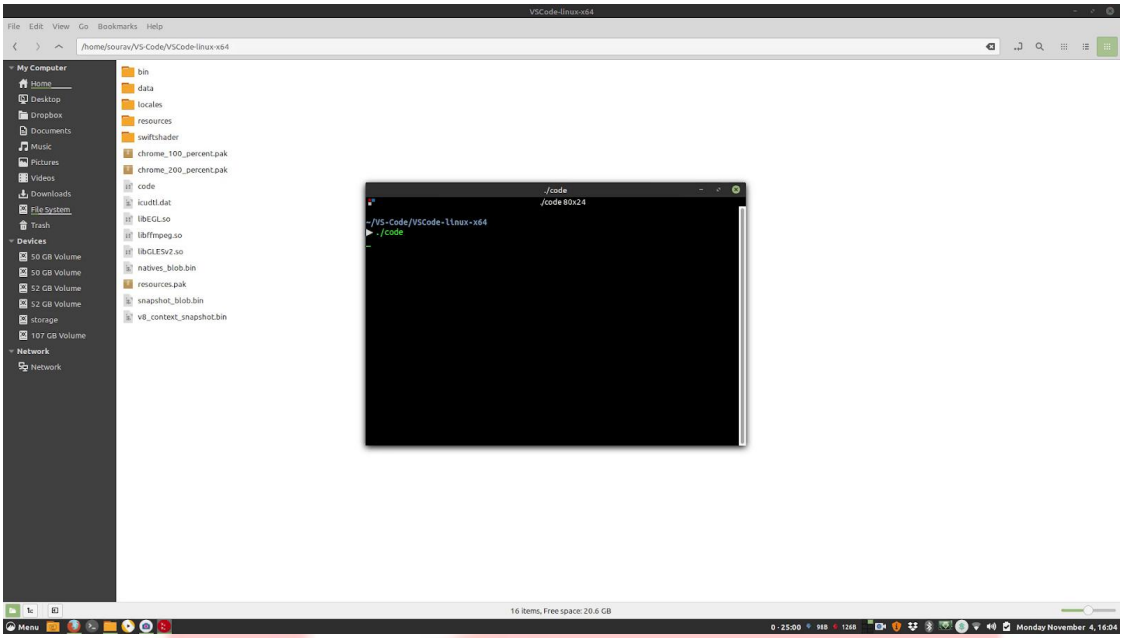
3. How to start VS Code Portable in Linux

Open terminal in the root folder of VS Code portable.

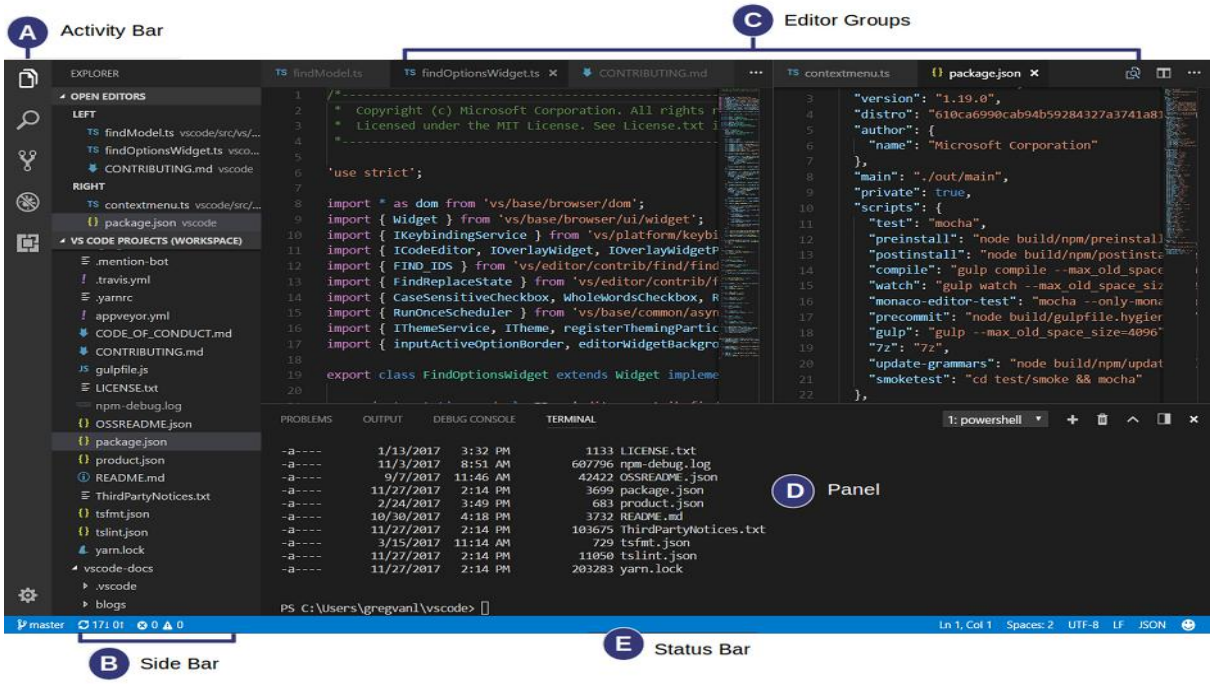


Make “code” file executable.
To do this use this command:
sudo chmod +x code

Now to start VS code use this command:
./code



4. VS Code User Interface



At its heart, Visual Studio Code is a code editor. Like many other code editors, VS Code adopts a common user interface and layout of an explorer on the left, showing all of the files and folders you have access to, and an editor on the right, showing the content of the files you have opened.

Basic Layout

VS Code comes with a simple and intuitive layout that maximizes the space provided for the editor while leaving ample room to browse and access the full context of your folder or project. The UI is divided into five areas:

Editor - The main area to edit your files. You can open as many editors as you like side by side vertically and horizontally.

Side Bar - Contains different views like the Explorer to assist you while working on your project.

Status Bar - Information about the opened project and the files you edit.

Activity Bar - Located on the far left-hand side, this lets you switch between views and gives you additional context-specific indicators, like the number of outgoing changes when Git is enabled.

Panels - You can display different panels below the editor region for output or debug information, errors and warnings, or an integrated terminal. Panel can also be moved to the right for more vertical space.

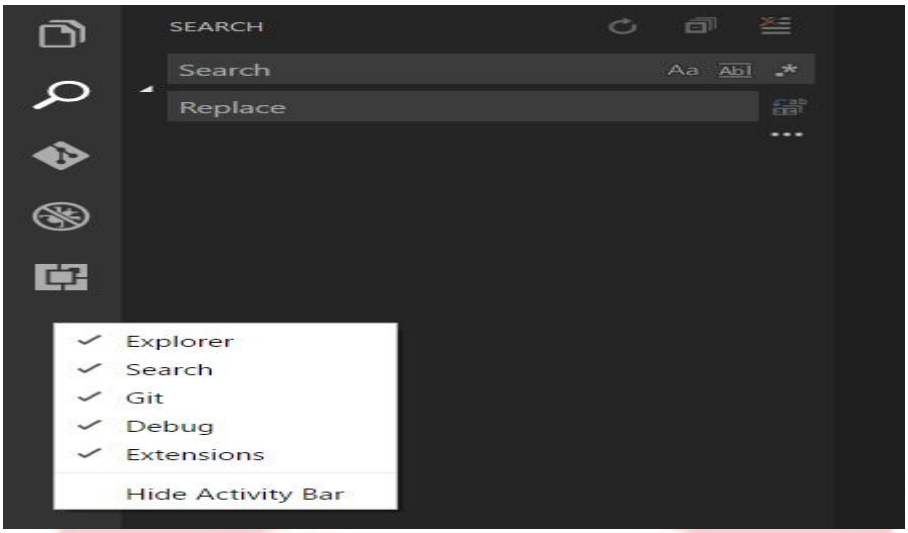
Each time you start VS Code, it opens up in the same state it was in when you last closed it. The folder, layout, and opened files are preserved.

Open files in each editor are displayed with tabbed headers (Tabs) at the top of the editor region. To learn more about tabbed headers, see the Tabs section below.

Tip: You can move the Side Bar to the right hand side (**View** > **Move Side Bar Right**) or toggle its visibility (Ctrl+B).

Activity Bar

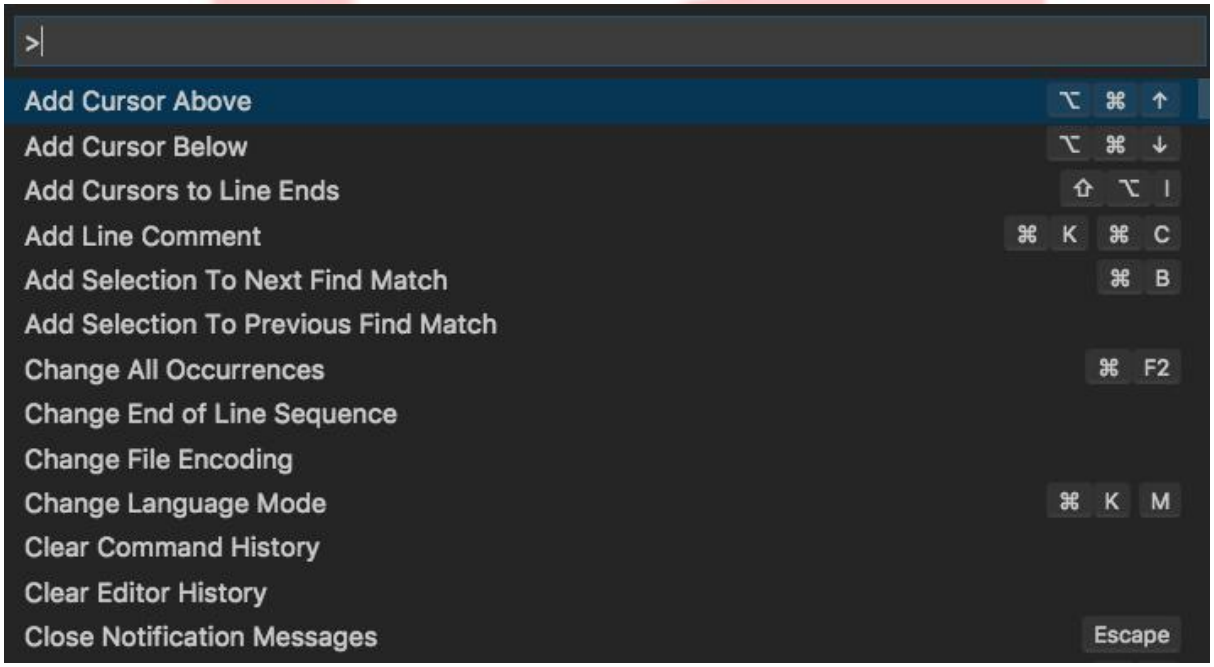
The **Activity Bar** on the left lets you quickly switch between Views. You can also reorder Views by dragging and dropping them on the **Activity Bar** or remove a View entirely (right click **Hide from Activity Bar**).



Command Palette

VS Code is equally accessible from the keyboard. The most important key combination to know is Ctrl+Shift+P, which brings up the **Command Palette**.

From here, you have access to all of the functionality of VS Code, including keyboard shortcuts for the most common operations.

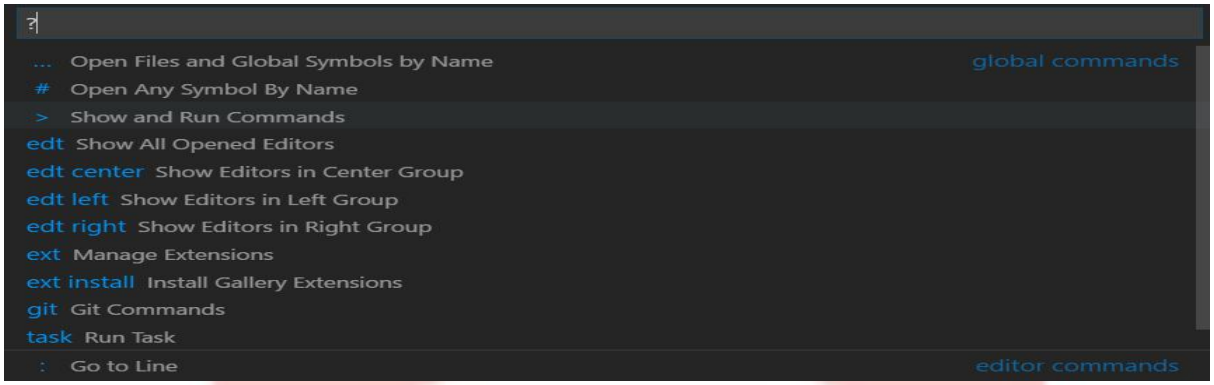


The **Command Palette** provides access to many commands. You can execute editor commands, open files, search for symbols, and see a quick outline of a file, all using the same interactive window. Here are a few tips:

- Ctrl+P will let you navigate to any file or symbol by typing its name
- Ctrl+Shift+Tab will cycle you through the last set of files opened
- Ctrl+Shift+P will bring you directly to the editor commands

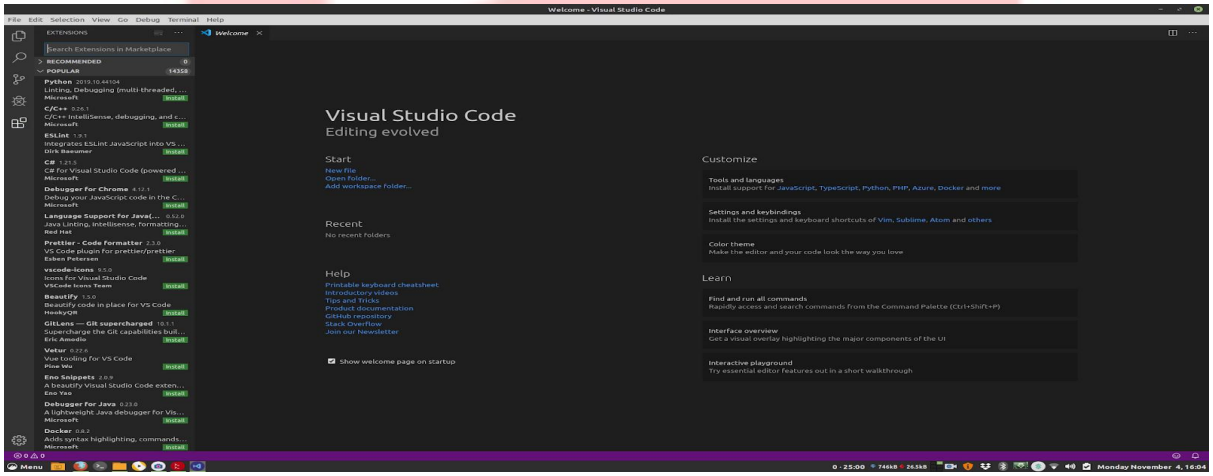
- Ctrl+Shift+O will let you navigate to a specific symbol in a file
- Ctrl+G will let you navigate to a specific line in a file

Type ? into the input field to get a list of available commands you can execute from here:



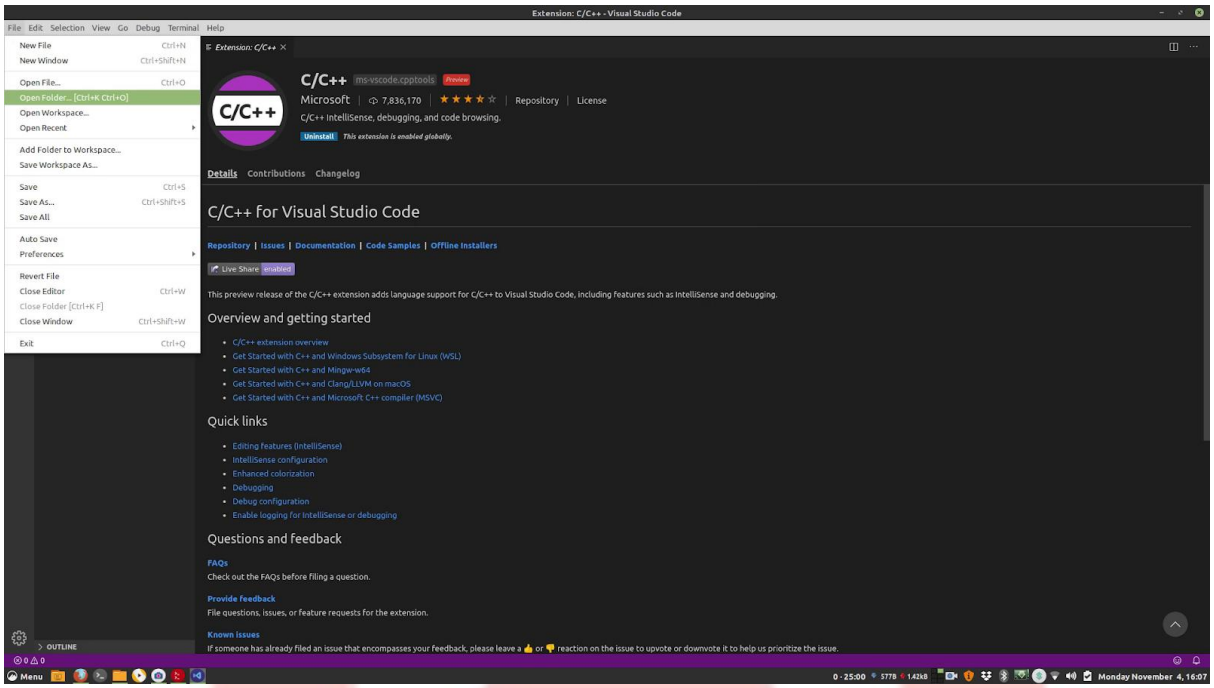
5. How to install Python and C/C++ IntelliSense extension from Marketplace

First click on Marketplace icon in the Activity bar. It is the bottom most icon usually.
In the marketplace search for “Python”. Install the “Python” extension developed by Microsoft.
Now to install C/C++ IntelliSense extension search for “C++” in marketplace. Install the “C/C++” extension developed by Microsoft.

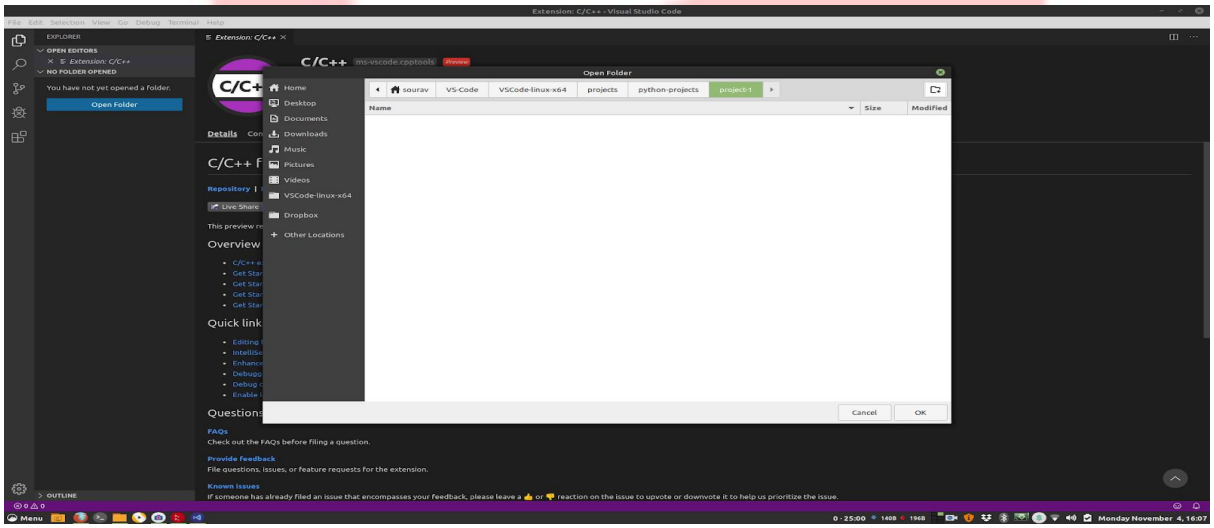


6. How to Create Project Folder in VS Code

In the top bar menu select File > Open Folder



Now navigate to your desired location, create your project folder and give it a suitable name. After creating your project folder go inside the folder and then click OK.



Now your project folder is added to VS Code's workspace.

7. How to select Python Interpreter and Conda Environment for your project from within VS Code

When you want your python to run in VS code you need to configure a python interpreter for it. You may have virtual environments with different versions of python or different packages

installed, each environment has its own interpreter. Now, in the case you run your virtual environments through conda — configuring this setting is not straight forward.

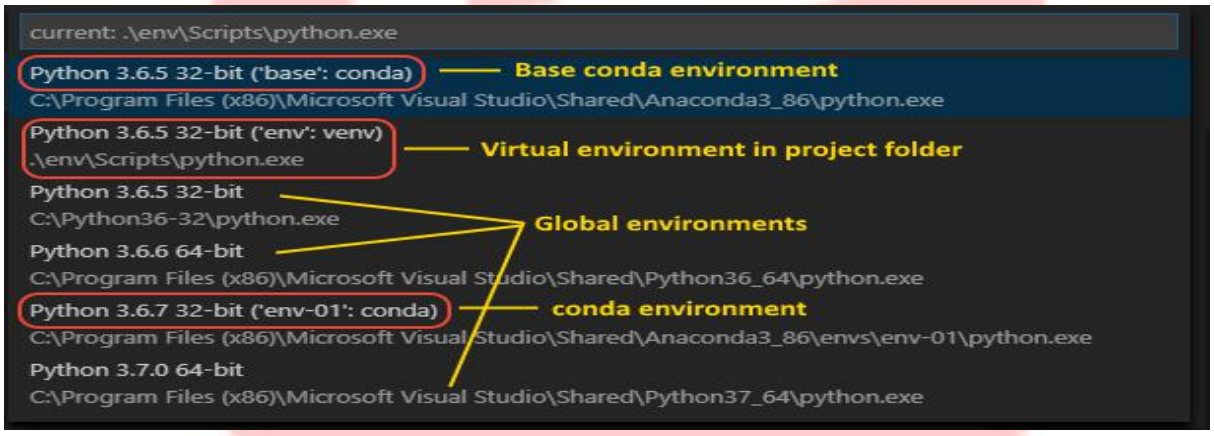
You need to 1) set your interpreter and 2) MAKE SURE YOUR CONDA ENVIRONMENT IS ACTIVATED.

Setting your python interpreter

Setting your interpreter is pretty easy, you got a few ways to do it (taken from VS code documentation):

a) Open the Command Pallete (Ctrl + Shift + P) and type "Python: Select Interpreter", select the command and it should present you a list of available interpreters (the ones Python extension has detected)

The **Python: Select Interpreter** command displays a list of available global environments, conda environments, and virtual environments. (See the Where the extension looks for environments section for details, including the distinctions between these types of environments.) The following image, for example, shows several Anaconda and CPython installations along with a conda environment and a virtual environment (env) that's located within the workspace folder:



b) On the status bar, in the bottom left corner of the screen, use the **Select Python Interpreter** option, if available (it may already show a selected interpreter). Then select from the list presented to you.

Click on Select Python Interpreter

c) Change your `python.pythonpath` manually in the settings —to get to settings hit Ctrl +, (i.e. Ctrl + comma) then select w*orkspace settings* tab and set:

"python.pythonPath": "<path-to-your-interpreter>/bin/python"

Note: If you select an interpreter without a workspace folder open, VS Code sets python.pythonPath in your *user settings* instead, which sets the default interpreter for VS Code in general. The user setting makes sure you always have a default interpreter for Python projects. The workspace settings lets you override the user setting.

8. How to activate conda environment

Making sure your conda environment is activated can be done using shell arguments. There should be an activate.bat script that's starting the anaconda prompt in your Anaconda/Miniconda installation, in the Scripts folder. This activate.bat script location looks something like this:

If your shell is Bash or a Bourne variant, enable conda for the current user with

This command:
echo ". /home/<user>/miniconda3/etc/profile.d/conda.sh" >> ~/.bashrc

or, for all users, enable conda with

This command:
sudo ln -s /home/<user>/miniconda3/etc/profile.d/conda.sh /etc/profile.d/conda.sh

The options above will permanently enable the 'conda' command, but they do NOT put

conda's base (root) environment on PATH. To do so, run

This command:
conda activate

in your terminal, or to put the base environment on PATH permanently, run

This command:
echo "conda activate" >> ~/.bashrc