Environment Setup - VS Code

VS Code Installation

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET).

To install VS Code for your specific platform, refer below.

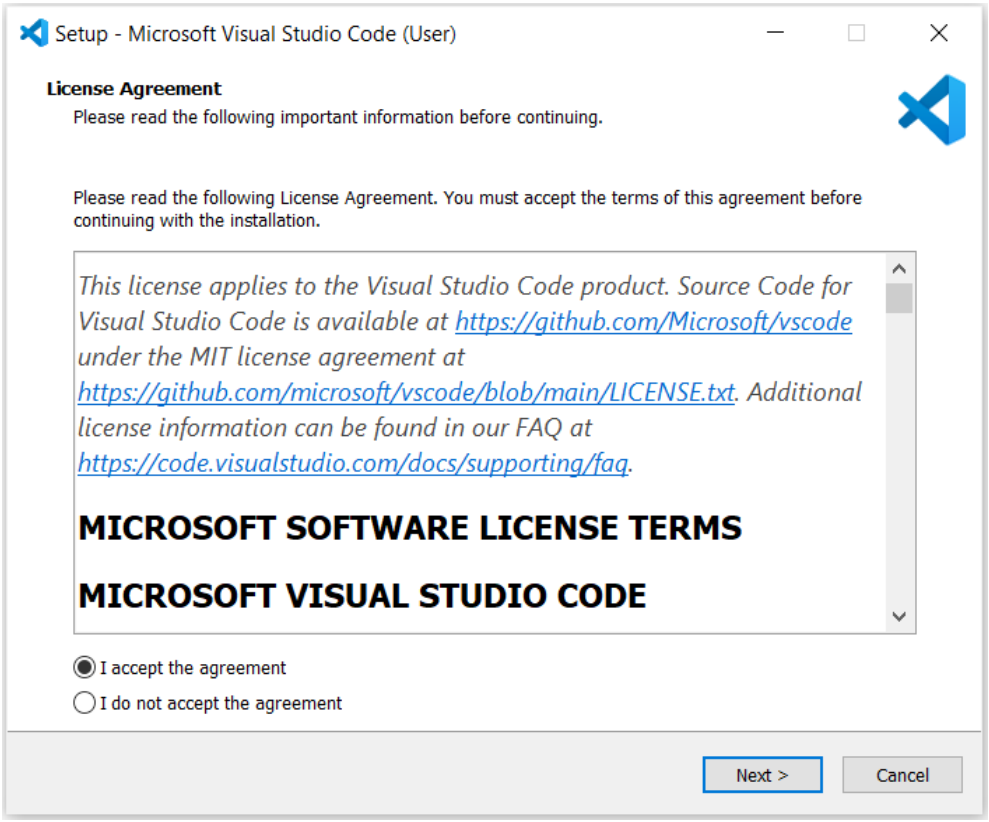
A. For Windows

1. Go to the link below. It will download the Visual Studio Code installer for Windows.

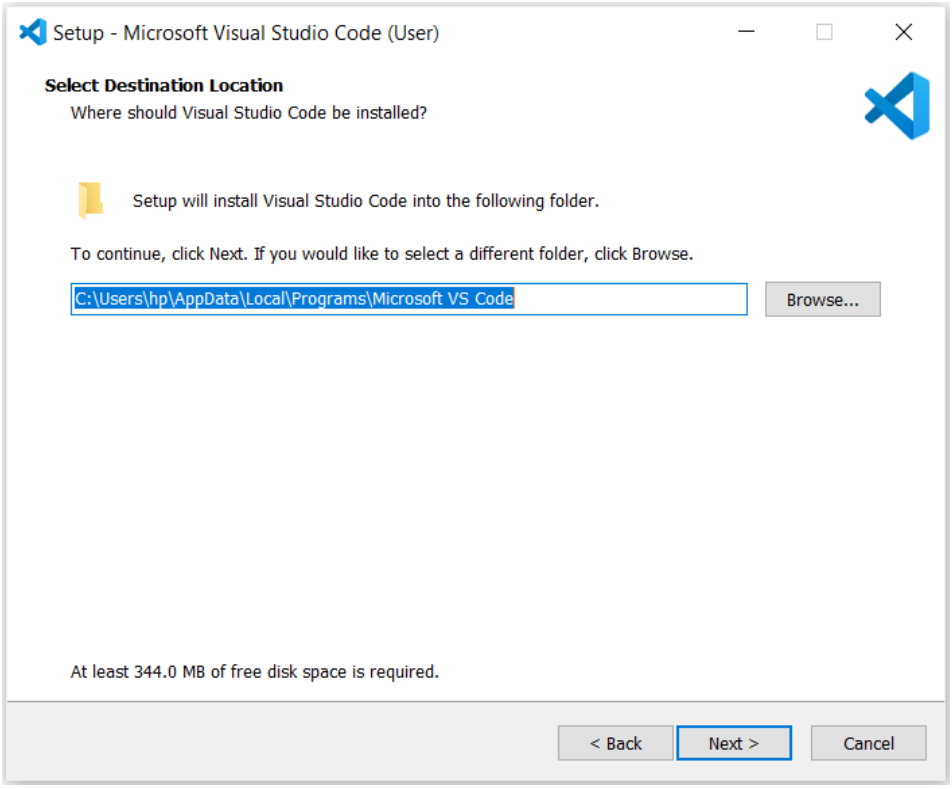
   https://go.microsoft.com/fwlink/?LinkID=534107

2. Once it is downloaded, run the installer (VSCodeUserSetup-{version}.exe).

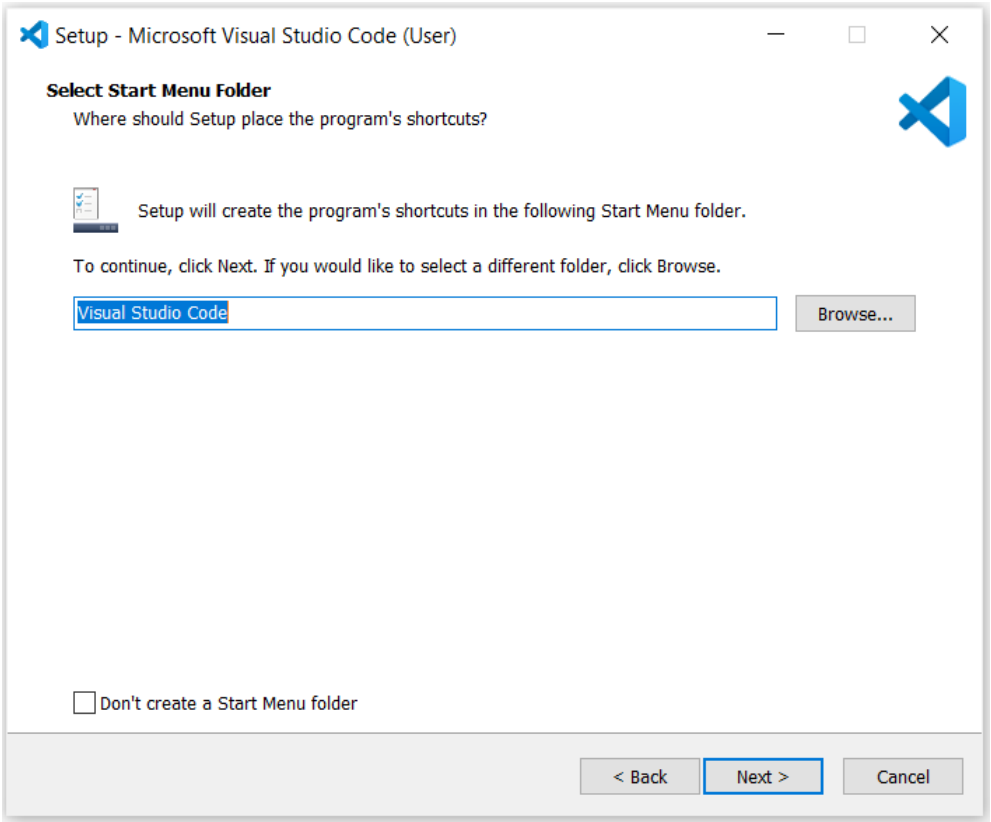
   Accept the agreement, and click Next.



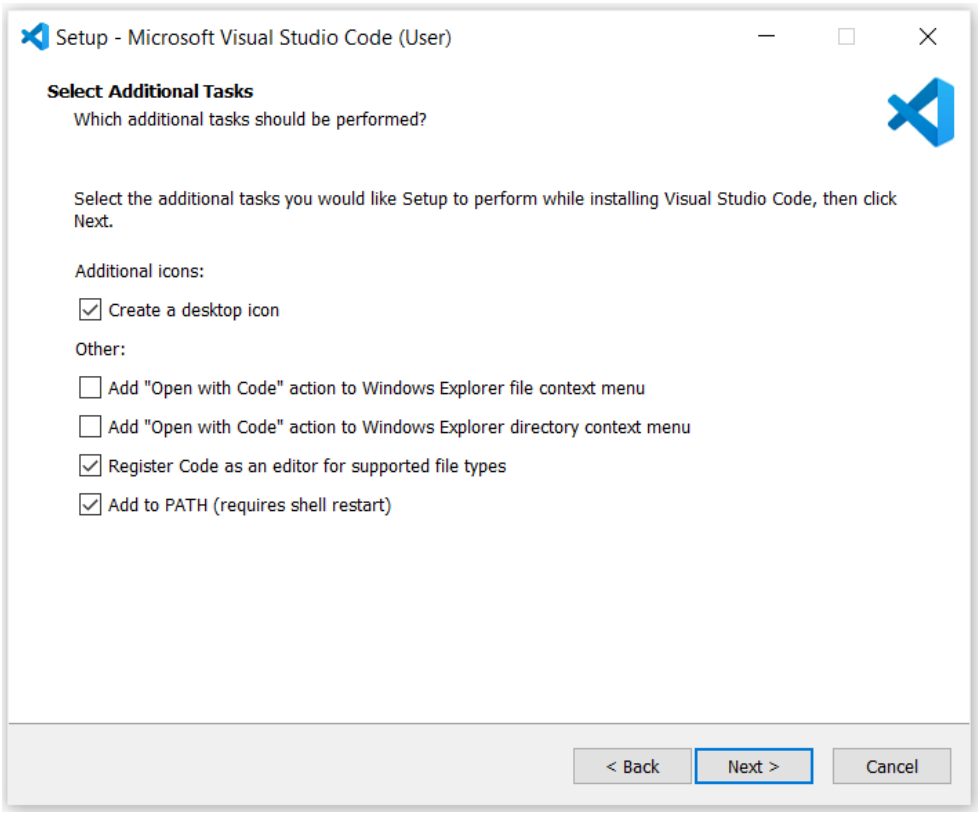
3. By default, VS Code is installed under C:\Users\{Username}\AppData\Local\Programs\Microsoft VS Code. Click Next.



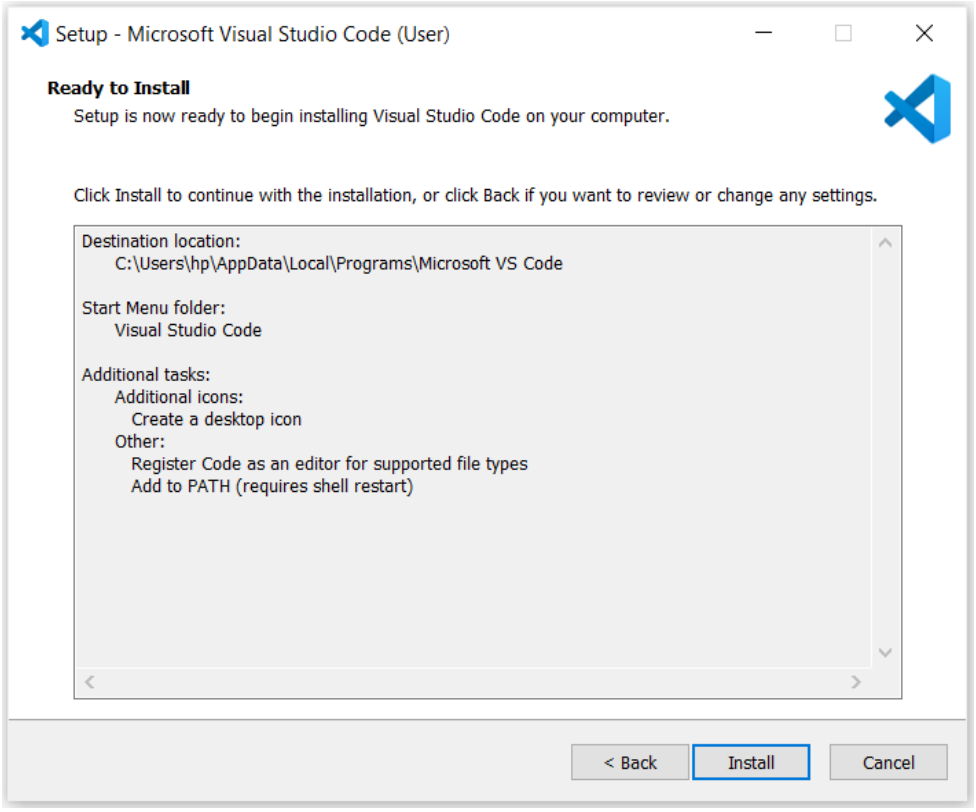
Click Next.



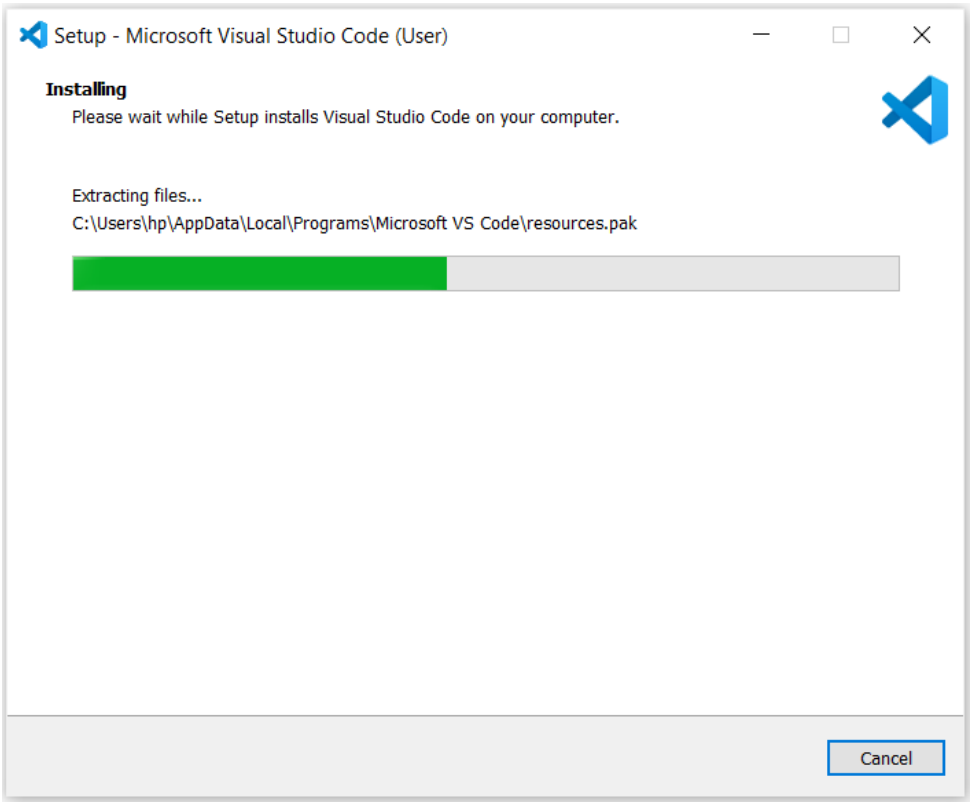
4. Select Additional Tasks, click Next.



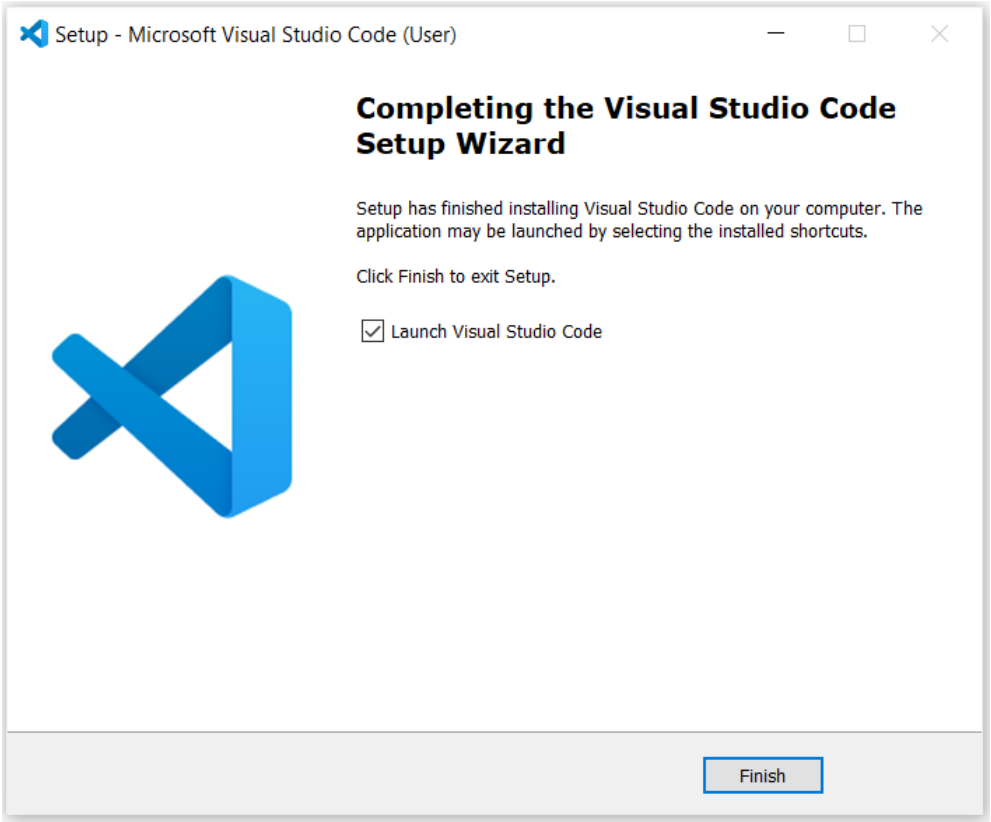
5. Install



Installation will begin.

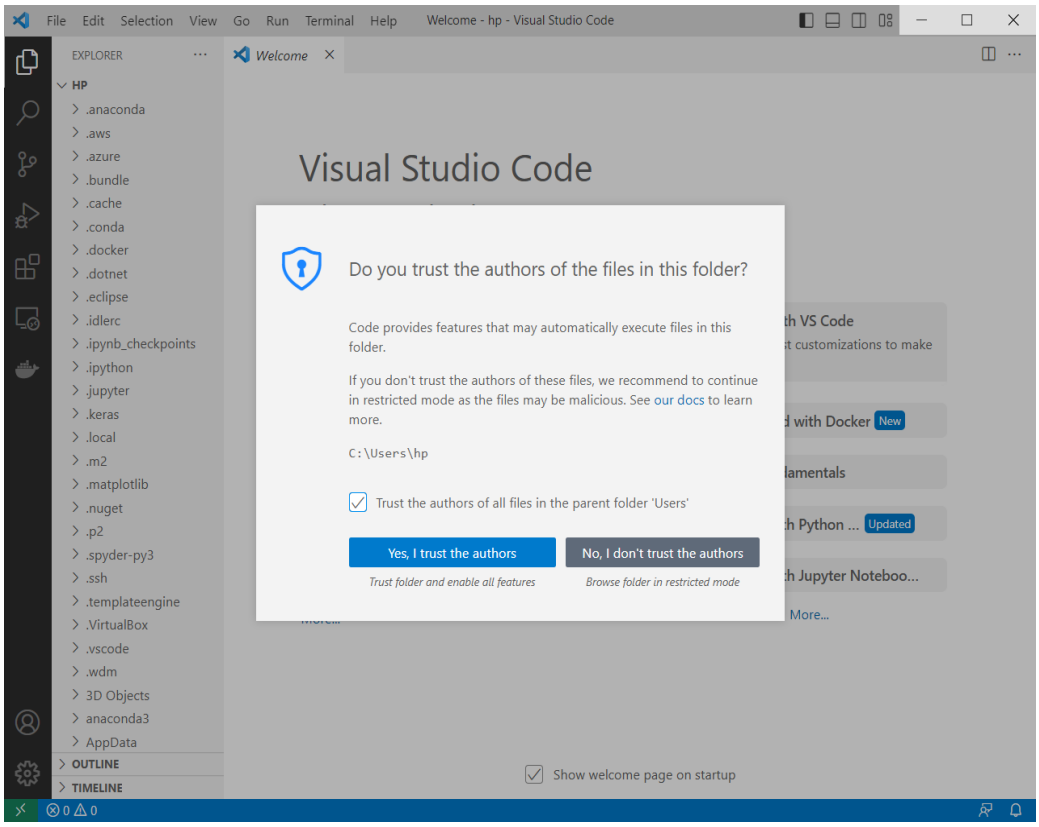


6. Finish and Launch VSCode.

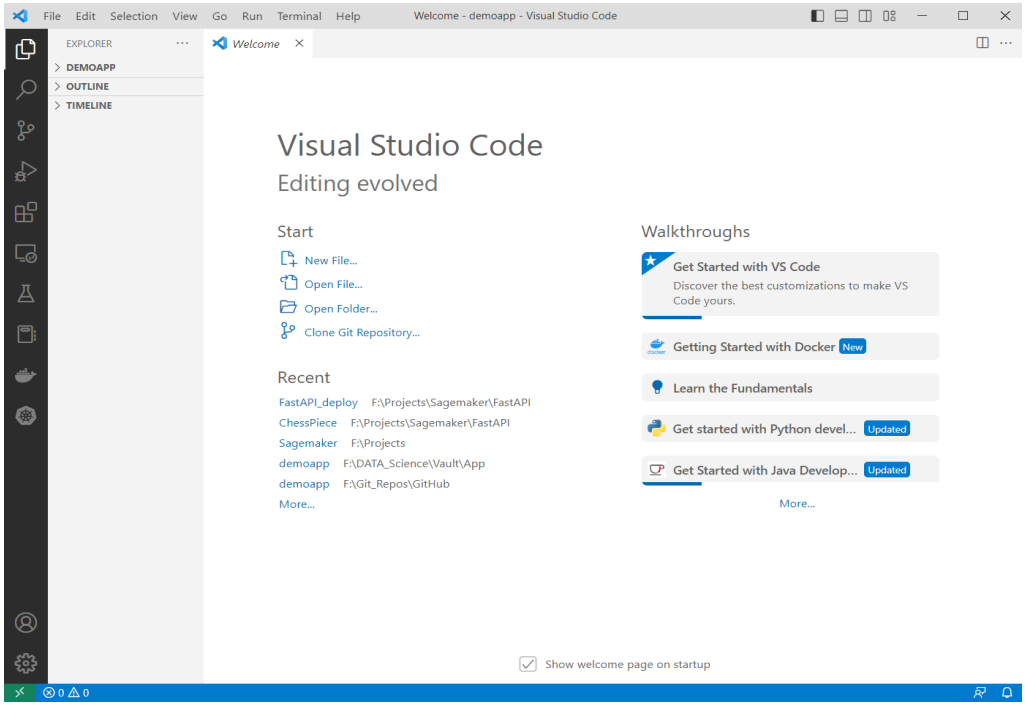


7. Once finished, it will automatically start the VSCode.

Note that: Setup will add Visual Studio Code to your %PATH%, so from the console as well you can type '`code .`' to open VS Code on that folder. Trust the authors when prompted.



VS Code will start.



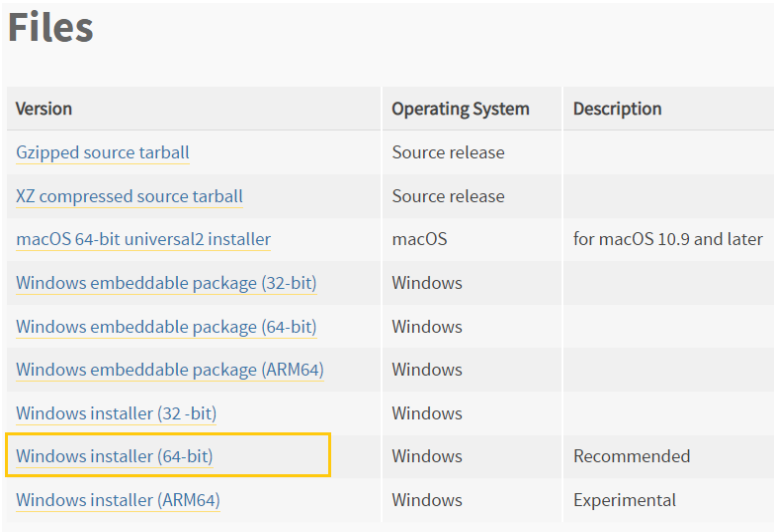
Install Python on Windows

1. Go to the link below to explore the installer for Windows.

   https://www.python.org/downloads/windows/

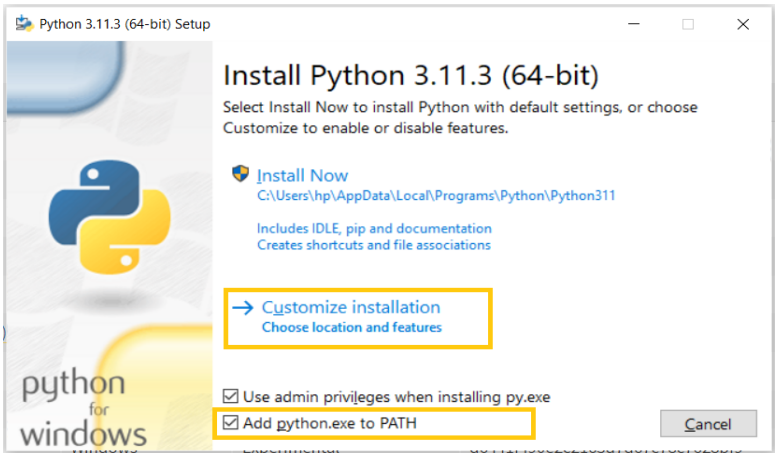
2. Search and Select the Python 3.11.3
   Note that there might be a new Python version available, but we recommend you to install Python 3.11.3 as the assignments are tested with this version of Python. In case you use the other versions, there might be issues later when you will be required to install different libraries in future assignments.

3. Download the installer according to your system.

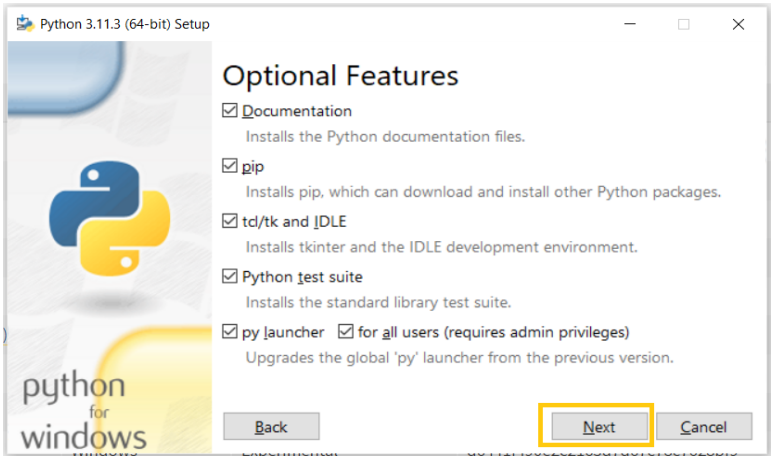


4. Once downloaded, start the installer.

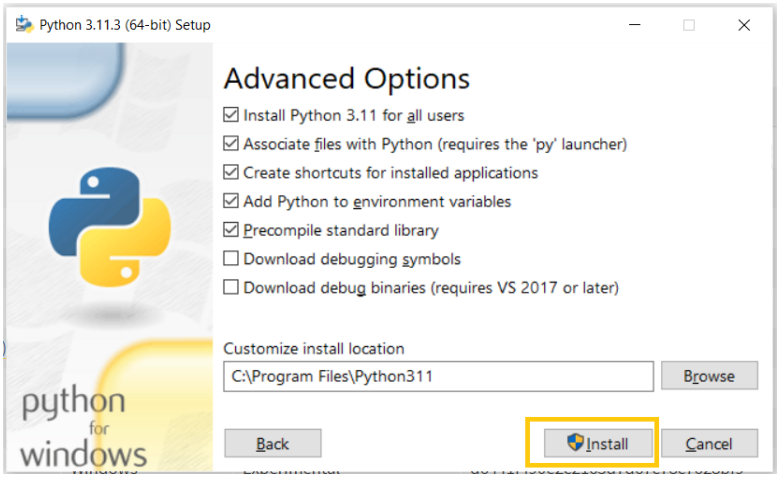
   Enable '*Add python.exe to PATH*', and go to *Customize installation*.



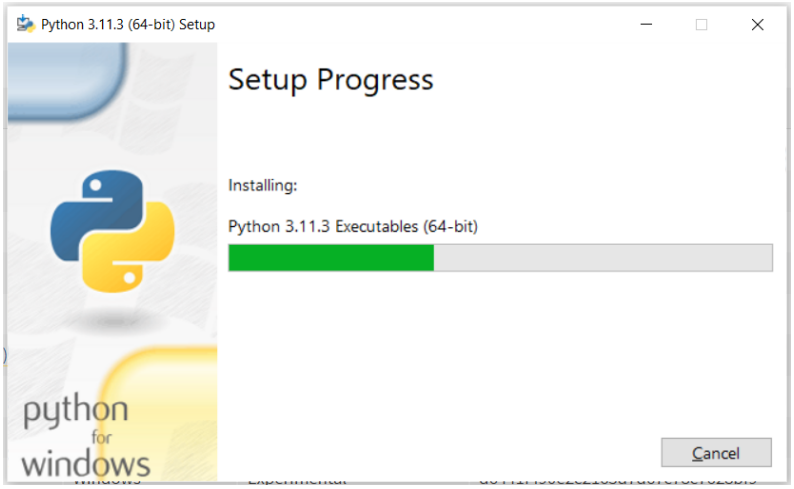
5. Click Next.



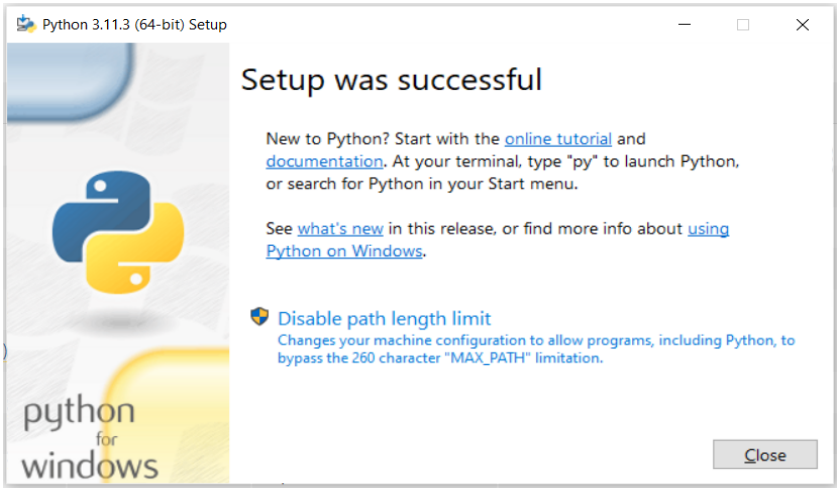
6. Enable items as shown below, and click *Install*.



7. Installation will begin.

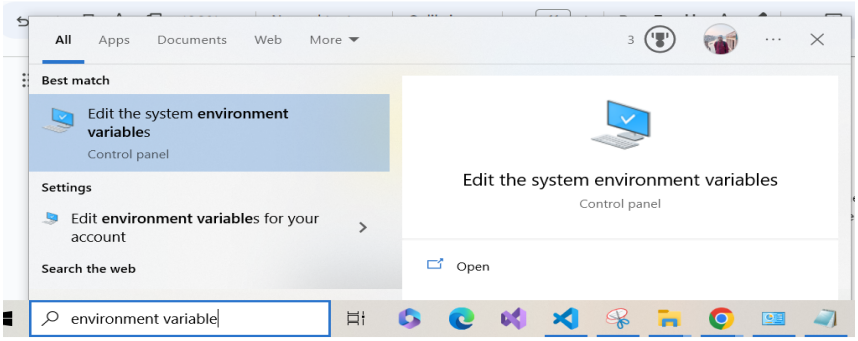


8. Once the Installation finishes successfully, close the wizard.

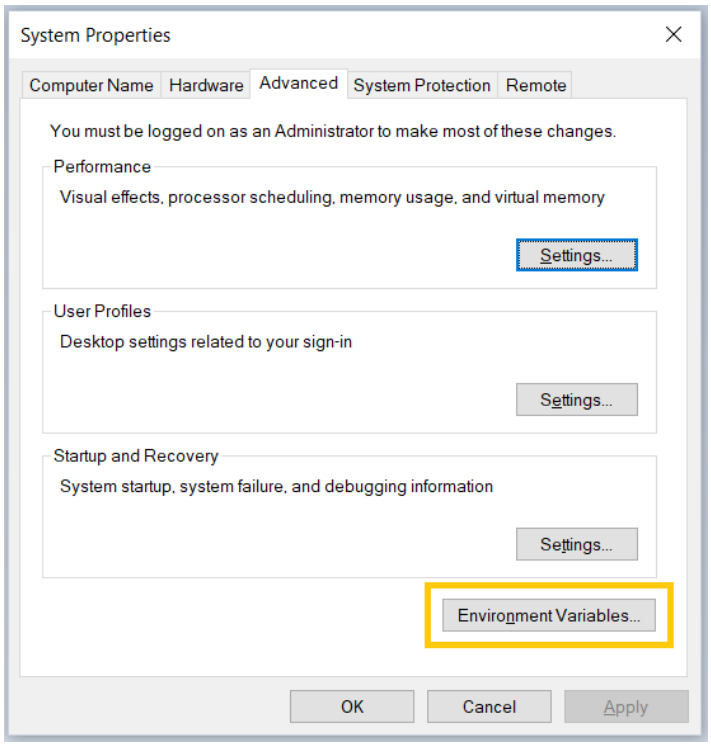


9. Now, make sure the location of your Python installation is included in your PATH environment variable.

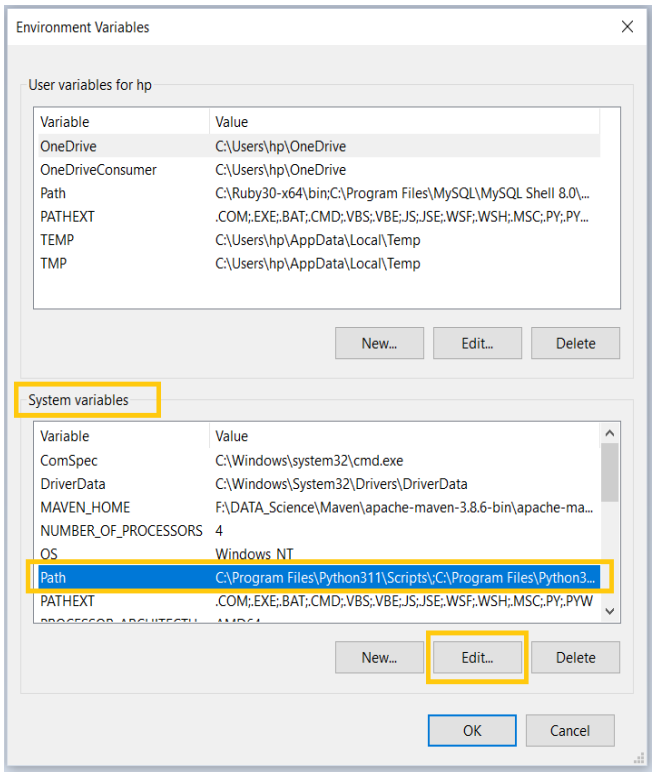
   Go to the Search tab, and search for “Environment Variables”, and select “*Edit the system environment variables*”.



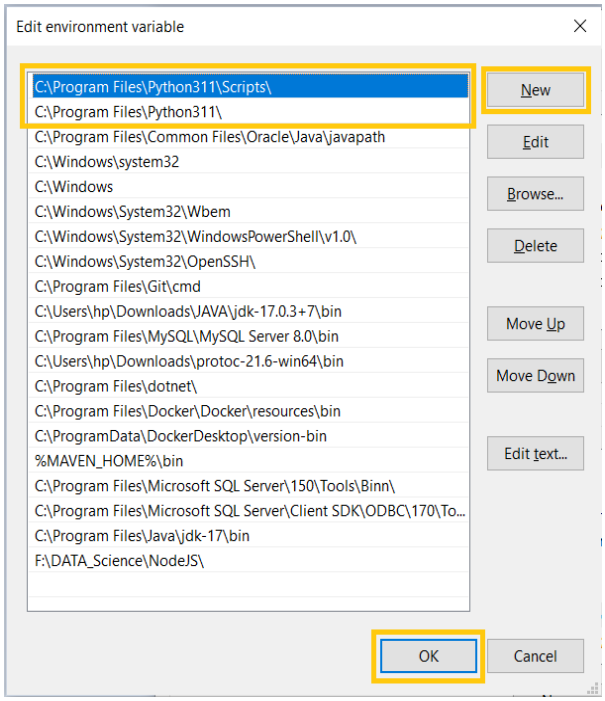
10. Go to Environment variables.



11. A new window will pop up. For *System variables*, select *Path*, and click *Edit*.



12. In the Edit environment variable window, make sure the paths for your Python installation are present. They should be present automatically right after installation. If not, you can add them using the *New* button, and press *OK* for the subsequent open windows.



13. Once the Environment path is set, check if python is installed successfully. Open the command prompt by typing 'cmd' in the search bar, and check the version installed using following command:

```
python --version
```

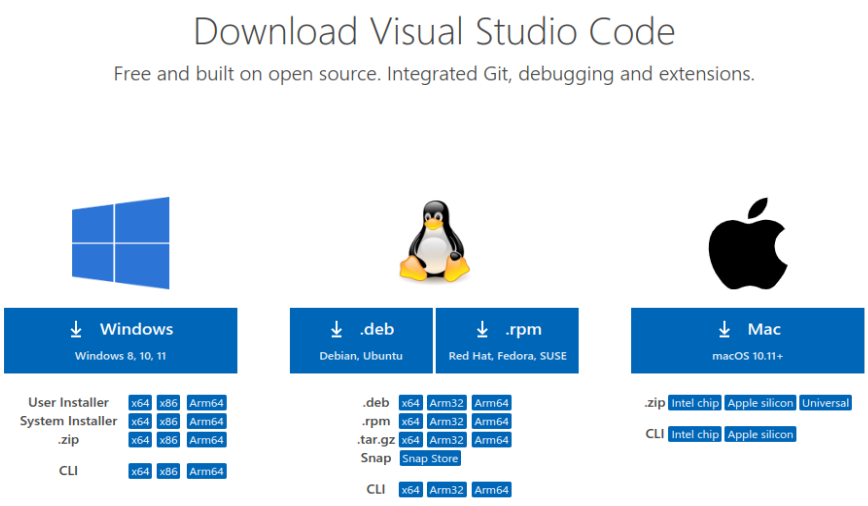
```
C:\Users\hp>python --version
Python 3.11.3
```

Hence, python installed successfully.

B. For Linux

1. Go to the link below. It will download the Visual Studio Code installer for ubuntu. Download the **.deb** package for Ubuntu.

https://code.visualstudio.com/docs/setup/linux



2. After downloading the installer, open the terminal and go to Downloads. Check the list of files in Downloads using `ls` command.

```
shlok@shlok-Latitude-3480:~$ cd Downloads/
shlok@shlok-Latitude-3480:~/Downloads$ ls
 00-ScalarDataTypes.mp4
 01_Lists_and_Tuples_1.mp4
 02-Conditions-Loops-Blocks.mp4
 03-Lists-Strings-for.mp4
 04_List_and_Tuples_2.mp4
'1491772946_Tax Saving Components - April 2017.pdf'
'5th october dhai assignment.odt'
 Abhijeet_Shinde_Assignment_7.pdf
'ACP DevOps  Week 2 Assignment.mp4'
'ACP DevOps  Week 2 Assignment.rar'
 All-files
```

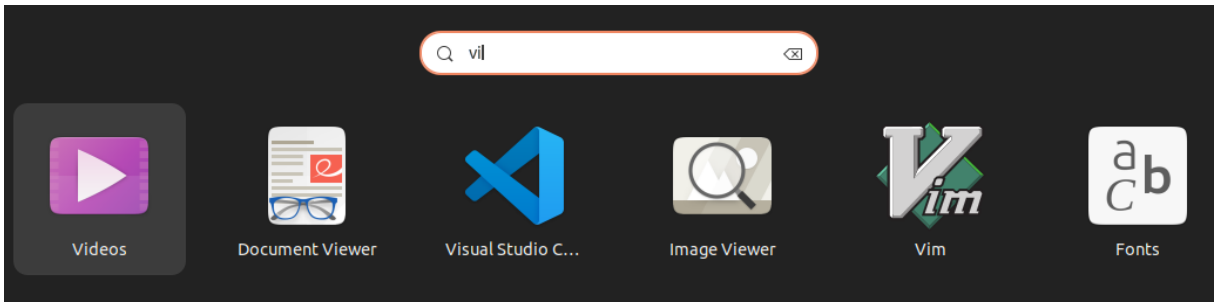
```
 code_1.76.0-1677667493_amd64.deb
```

3. Use the below command to install the Visual Studio Code and give the filename **code_1.76.0-1677667493_amd64.deb**

```
sudo dpkg -i code_1.76.0-1677667493_amd64.deb
```

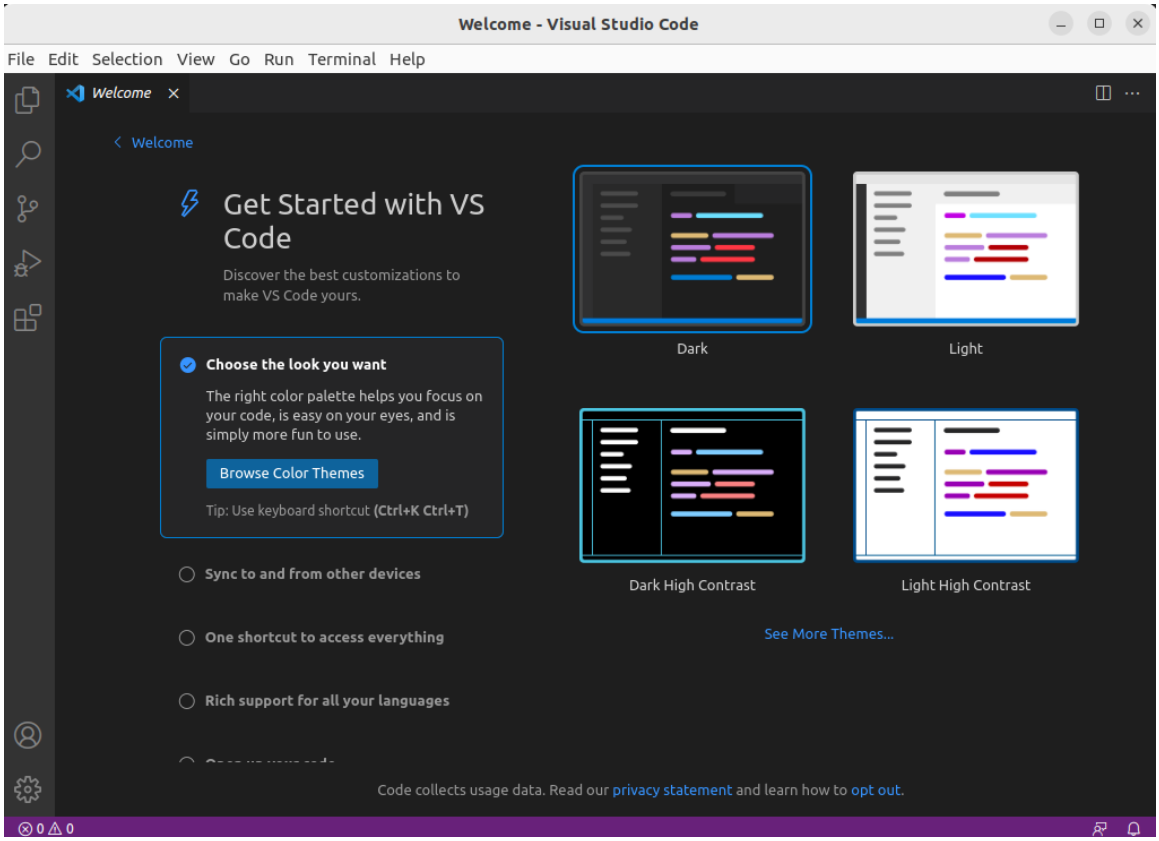
```
shlok@shlok-Latitude-3480:~/Downloads$ sudo dpkg -i code_1.76.0-1677667493_amd64.deb
[sudo] password for shlok:
Selecting previously unselected package code.
(Reading database ... 479631 files and directories currently installed.)
Preparing to unpack code_1.76.0-1677667493_amd64.deb ...
Unpacking code (1.76.0-1677667493) ...
Setting up code (1.76.0-1677667493) ...
Processing triggers for mailcap (3.70+nmu1ubuntu1) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu3) ...
Processing triggers for desktop-file-utils (0.26-1ubuntu3) ...
Processing triggers for shared-mime-info (2.1-2) ...
```

4. After the installation is complete, search Visual Studio in the system. You can open it by clicking on Visual Studio Code or it can be opened from the terminal by just giving the `code` command.




```
shlok@shlok-Latitude-3480:~/Downloads$ code
shlok@shlok-Latitude-3480:~/Downloads$
```

5. Further, you can see the welcome page of the VS Code.



Install Python on Linux

1. Go to the link below and refer to the steps under Linux.

   https://wiki.python.org/moin/BeginnersGuide/Download

C. For macOS

1. Check the installation steps given at https://code.visualstudio.com/docs/setup/mac
2. To install Python, go to the link below and refer to the steps under Mac.

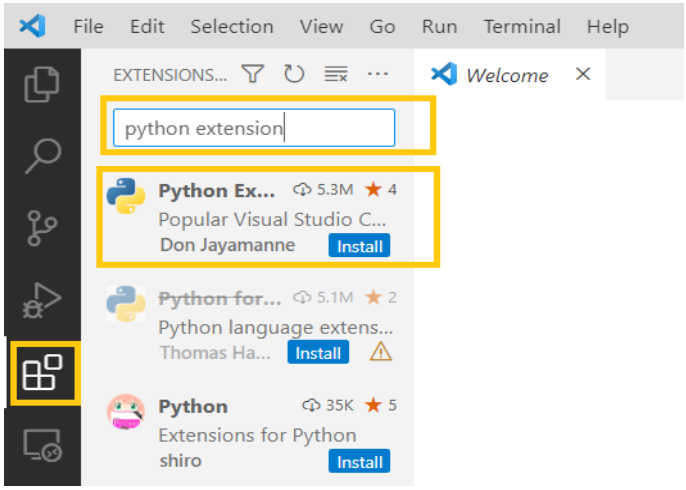
   https://wiki.python.org/moin/BeginnersGuide/Download

Setup Inside VS Code

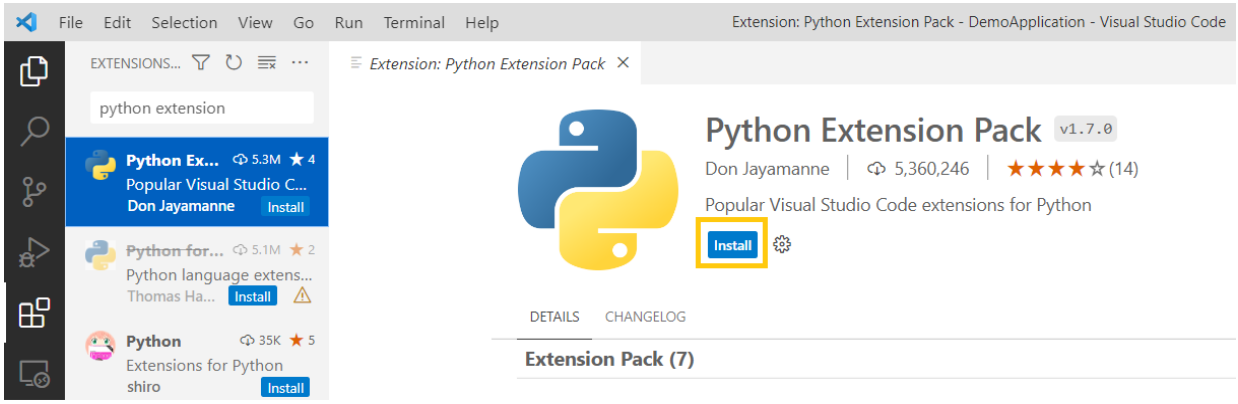
Setup Python in VS Code

Once you have installed VS Code, and want to add Python support to it, install the Python packages as shown below.

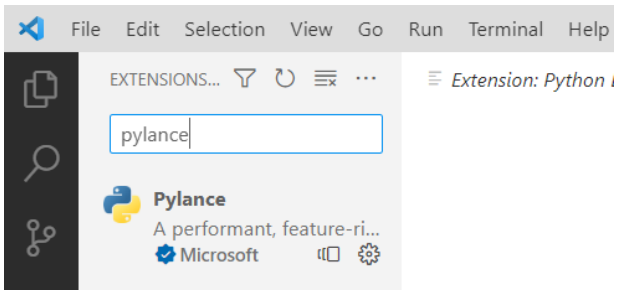
1. Go to the Extension icon on the left panel, search for “Python extension” , and select the one published by Don Jayamanne.



2. Click Install.



3. Similarly, install the Pylance.



4. Once installed, close the VS code.

5. Create a directory, named *Project*, for your application files. Within the *Project*, open VS code.

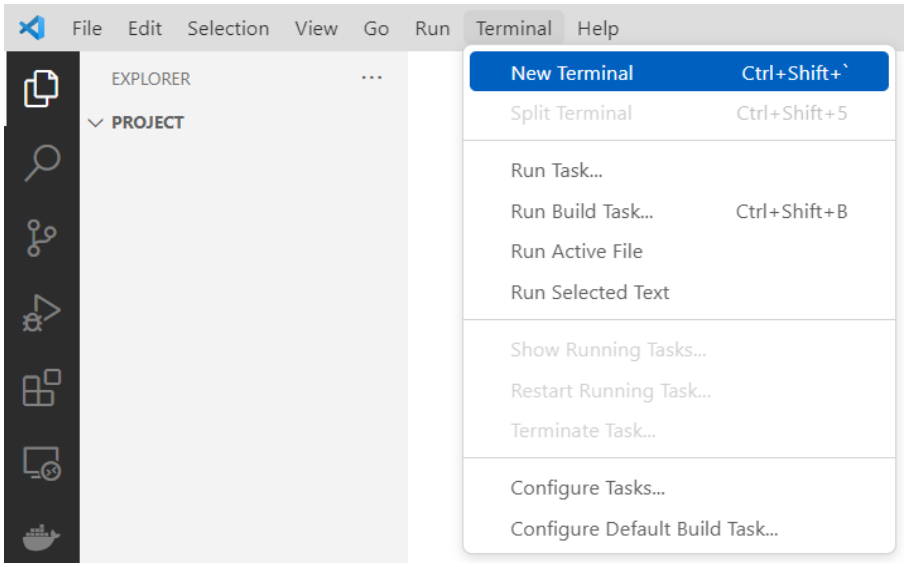
```
mkdir Project
cd Project
code .
```

```
F:\>mkdir Project

F:\>cd Project

F:\Project>code .
```

6. Now, to create a Virtual Environment, first go to *Terminal -> New Terminal*.



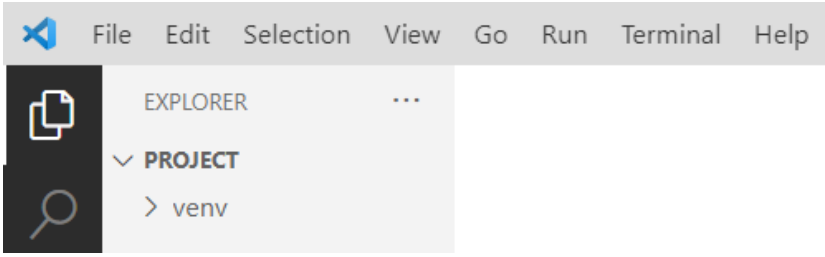
This will open a new terminal window at bottom.

7. Run the below command to create a Virtual Environment.

   `python -m venv <YOUR-PATH>\venv`

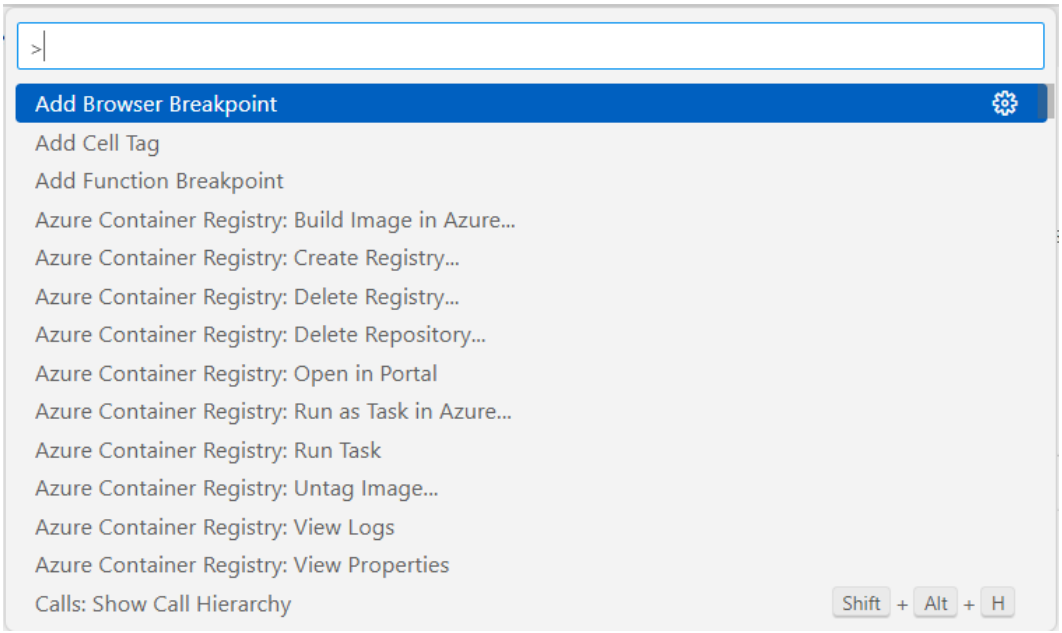
   For example: `python -m venv F:\Project\venv`

   The environment folder will appear in your project workspace.



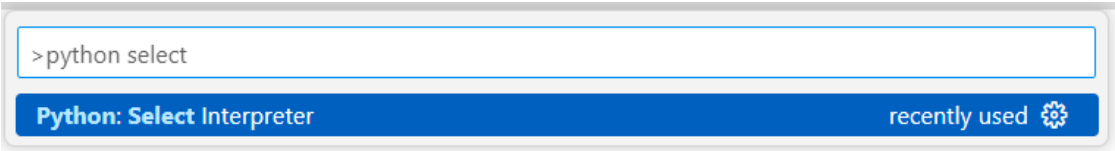
   Note that once the virtual environment is created, it needs to be started before using it.

8. Now, within VS Code, go to View > Command Palette OR press Ctrl+Shift+P. This will open Command Palette through which you have access to all of the functionality of VS Code.

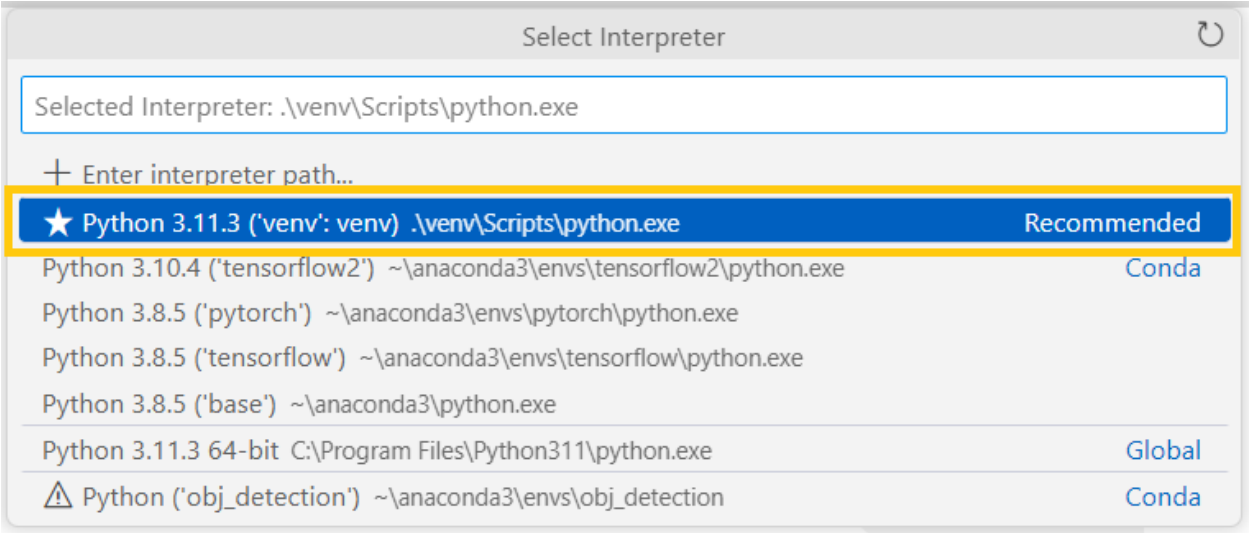


   The Command Palette provides access to many commands. You can execute editor commands, open files, search for symbols, and see a quick outline of a file, all using the same interactive window.

9. In the Command Palette, search for “Python Select Interpreter”. Then select “*Python: Select Interpreter*”.



10. Select your Python Interpreter. Select the one within the virtual environment i.e. *\venv\Scripts\python.exe*.



11. Now, to start the virtual environment, open a new terminal. It should start the virtual environment automatically.

If the virtual environment doesn't start automatically, run the below command in VS code terminal.

```
venv\Scripts\activate
```

You will notice that now you are inside the virtual environment.

12. Sometimes you will get an **error** message regarding **UnauthorizedAccess** while starting the virtual environment.

For example:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS F:\Project> venv\Scripts\activate
venv\Scripts\activate : File F:\Project\venv\Scripts\Activate.ps1 cannot be loaded because running scripts is disabled on this system.
about_Execution_Policies at https:/go.microsoft.com/fwlink/?LinkID=135170.
At line:1 char:1
+ venv\Scripts\activate
+ ~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : SecurityError: (:) [], PSSecurityException
    + FullyQualifiedErrorId : UnauthorizedAccess
PS F:\Project>
```

In that case, check the existing execution policy using the below command.

```
Get-ExecutionPolicy
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS F:\Project> Get-ExecutionPolicy
Restricted
PS F:\Project>
```

If it shows '*Restricted*', change it to '*Unrestricted*' using the below command.

```
Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy Unrestricted -Force
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS F:\Project> Get-ExecutionPolicy
Restricted
PS F:\Project> Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy Unrestricted -Force
PS F:\Project> Get-ExecutionPolicy
Unrestricted
PS F:\Project>
```

Now, try starting the Virtual Environment. It should start fine.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS F:\Project> Get-ExecutionPolicy
Unrestricted
PS F:\Project> venv\Scripts\activate
(venv) PS F:\Project>
```