HOW TO INSTALL AND USE PYCHARM IDE

PyCharm is developed by JetBrains and it is world no. 1 IDE for Python programmers. At present, this Python IDE is being used by large enterprises like Twitter, Pinterest, HP, Symantec and Groupon.

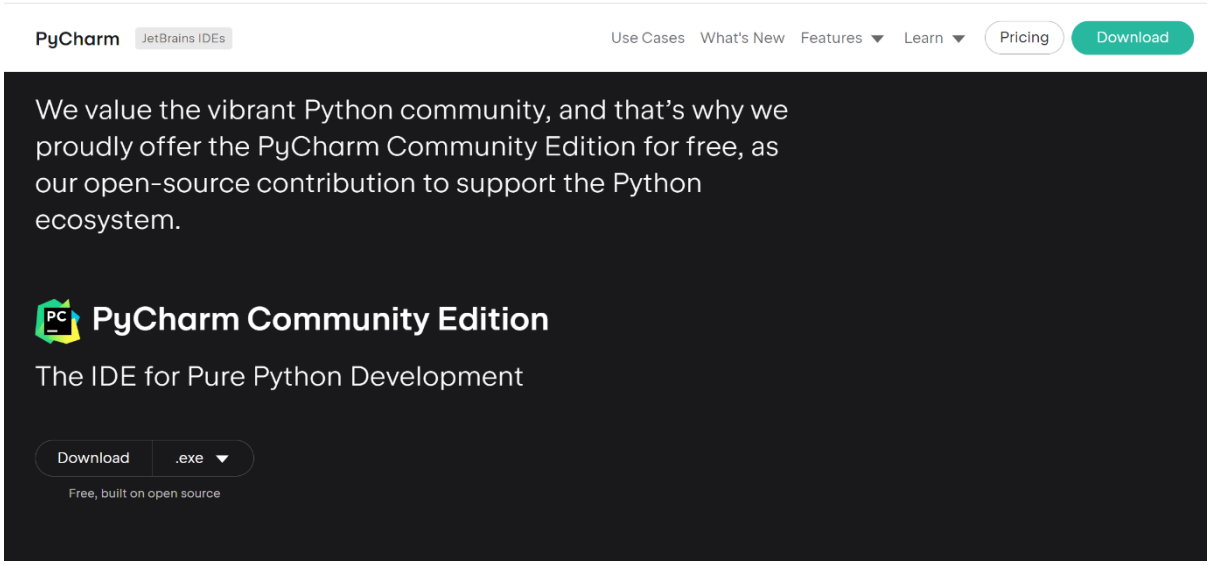
PyCharm is also compatible with Windows, Linux, and macOS. PyCharm provides all the tools we need for writing and executing Python programs. Also, PyCharm can be used to create Django projects.

PyCharm does not come with its own Python software. Hence, we should install Python in our system before we install PyCharm. Now, let us see how to install Community edition of PyCharm, since it is freely available.

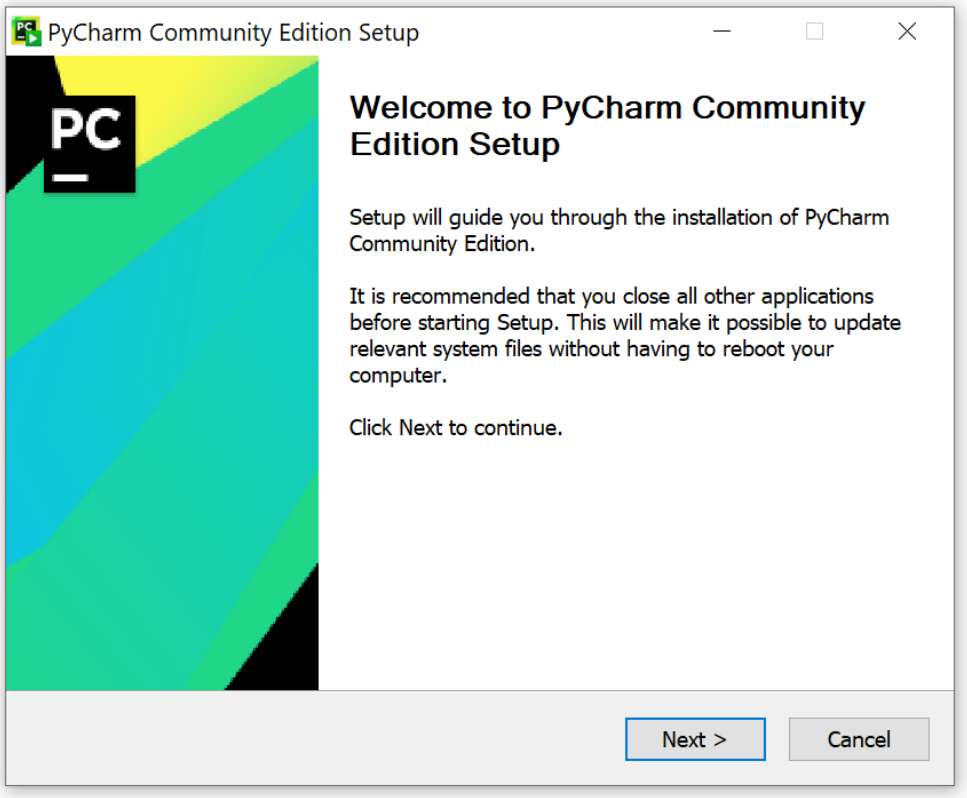
Step 1) To download PyCharm visit the website url:

https://www.jetbrains.com/pycharm/download/

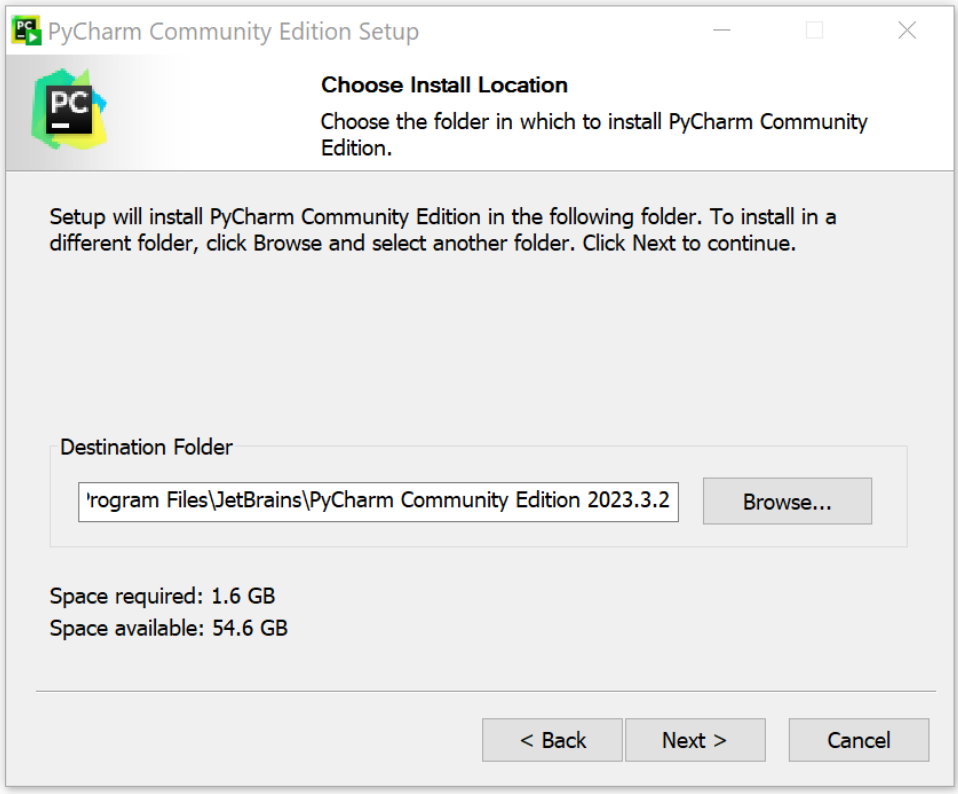
Go down in the above page to see “Download .exe” button below ‘PyCharm Community Edition’. Click on the “Download .exe” button and a file by the name “pycharm-community-2023.3.2.exe” will be downloaded into our system.



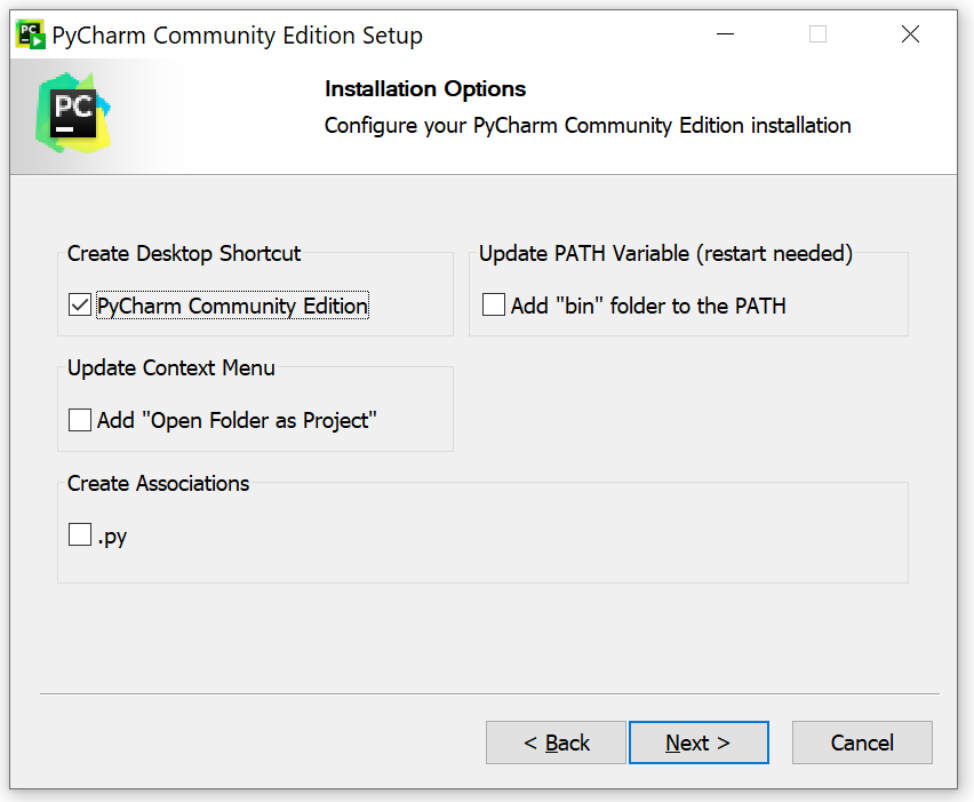
Step 2) Once the download is complete, run the .exe file to install PyCharm. The setup wizard should have started. Click “Next”.



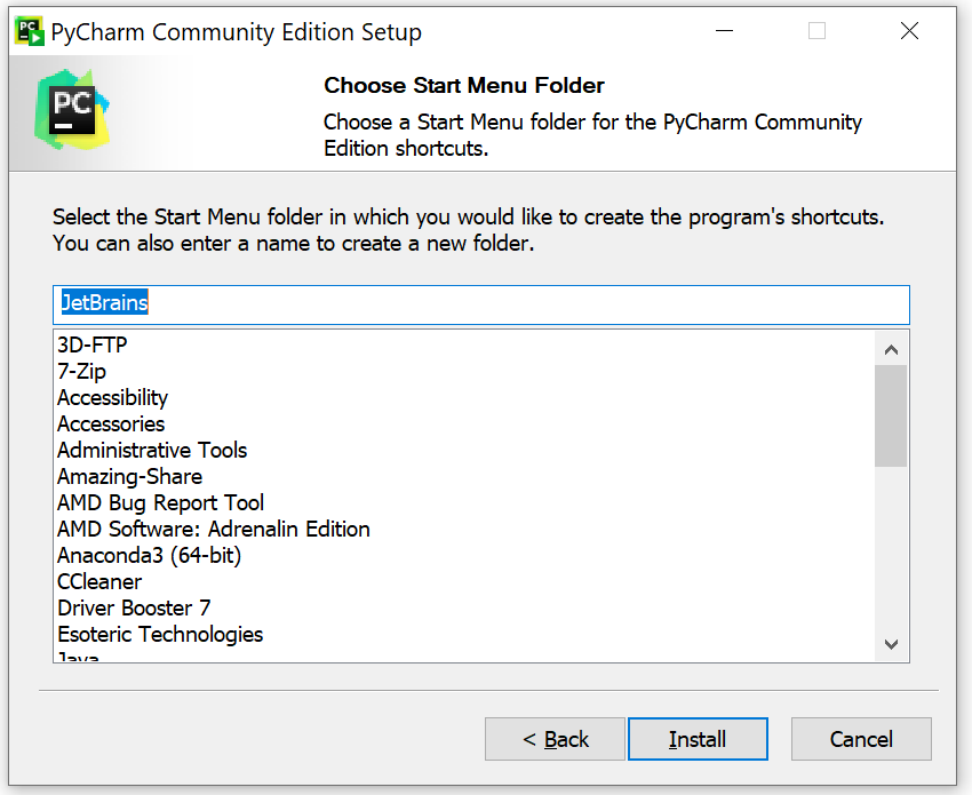
Step 3) On the next screen, change the installation path if required. Click "Next".



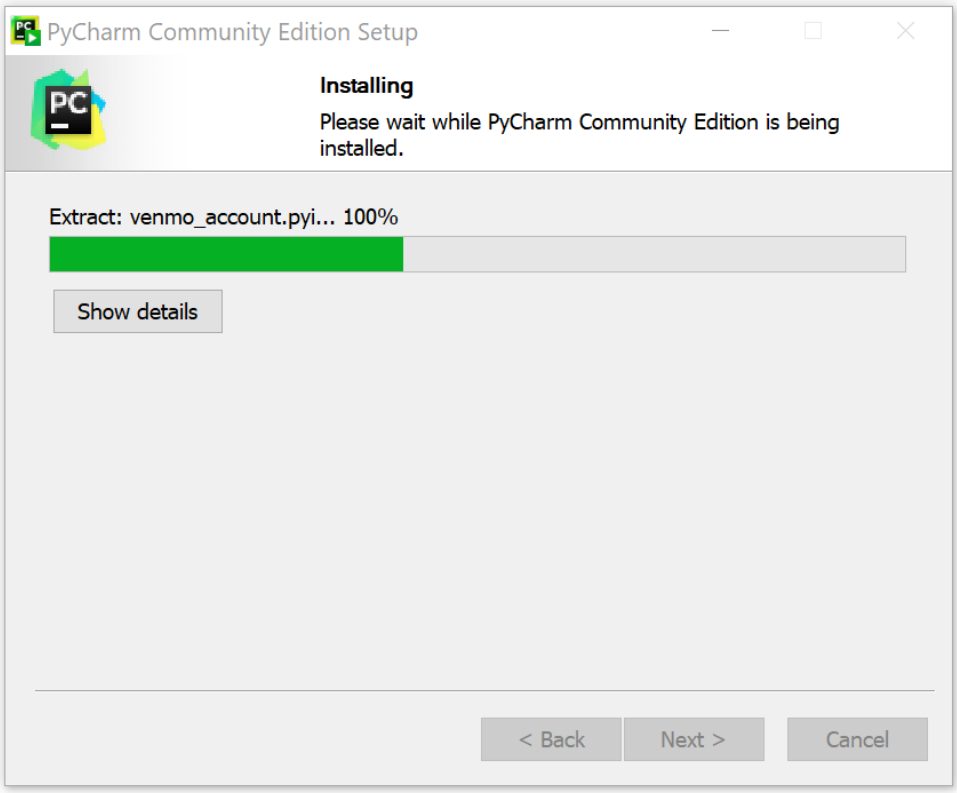
Step 4) On the next screen, we can create a desktop shortcut if we want. Select "PyCharm Community Edition". We can also select "Create Associations" checkbox to make Python programs open in PyCharm by default. After that, click on "Next".



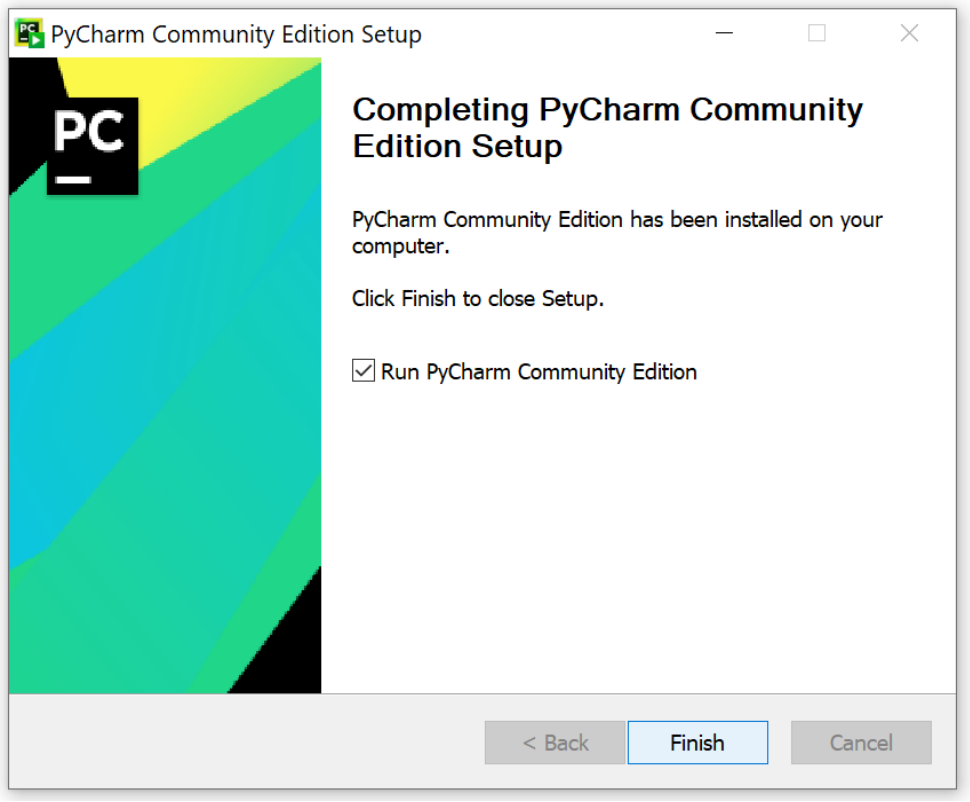
Step 5) Choose the start menu folder. Keep selected JetBrains and click on “Install”.



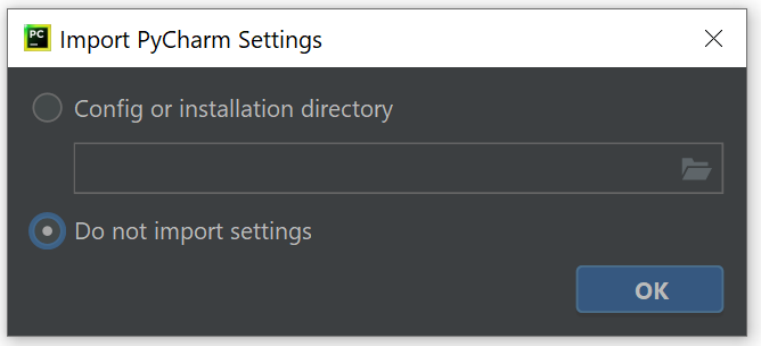
Step 6) The installation will start and we have to wait for its completion.



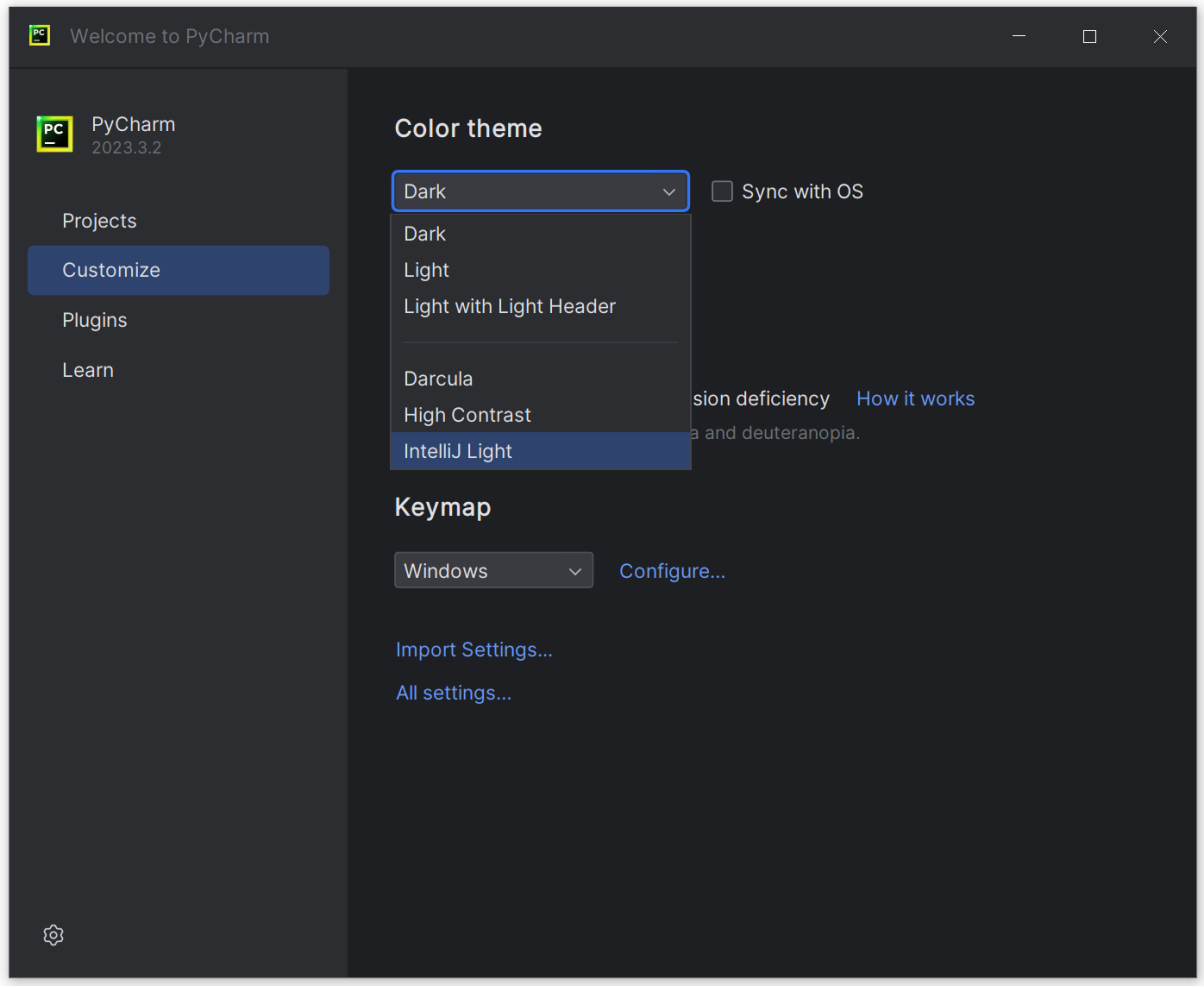
Step 7) Once installation is finished, we can click the “Run PyCharm Community Edition” box first and then click “Finish”.



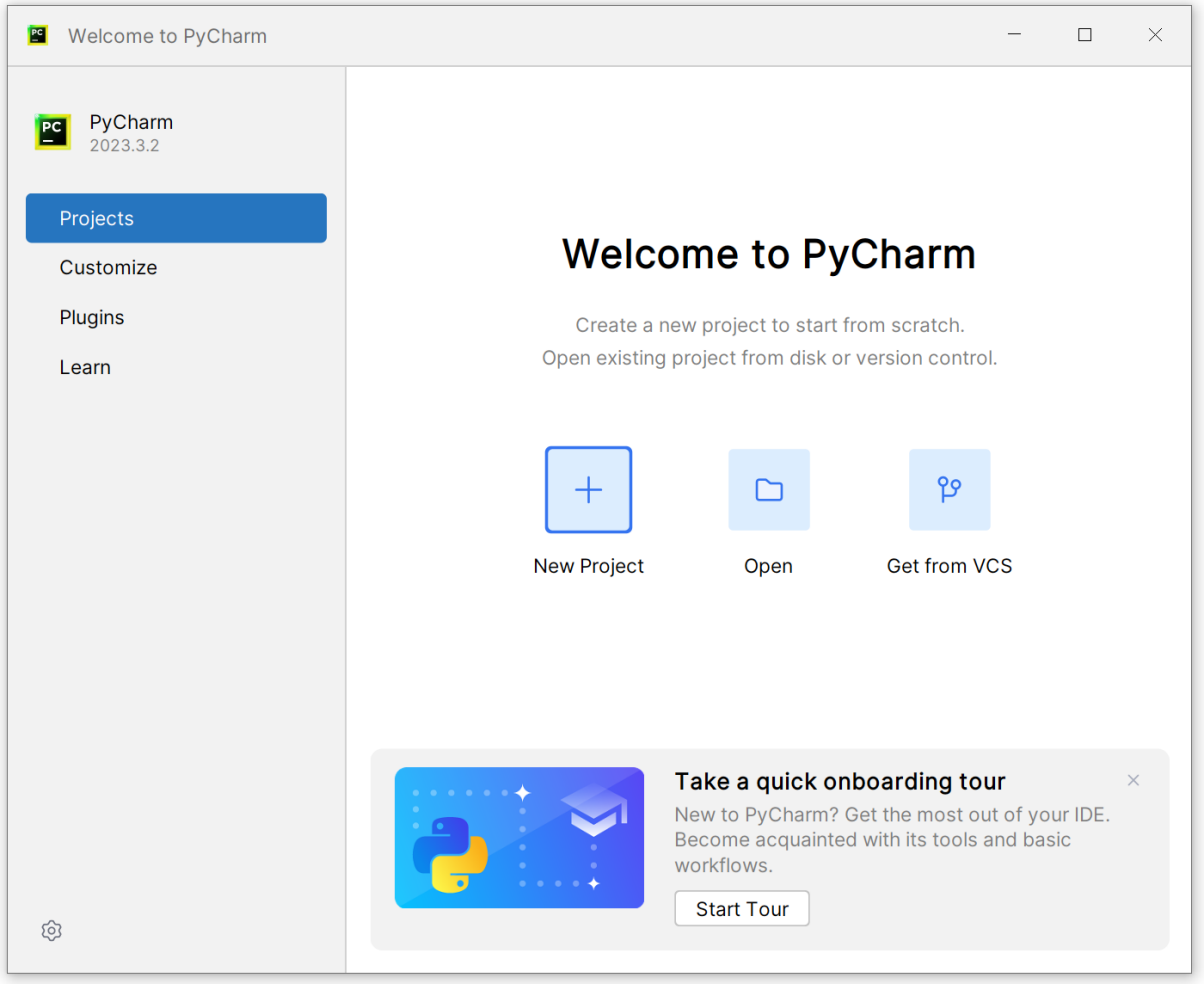
Step 8) After we click on "Finish", the Following screen will appear. Select “Do not import settings” and click on “OK”.



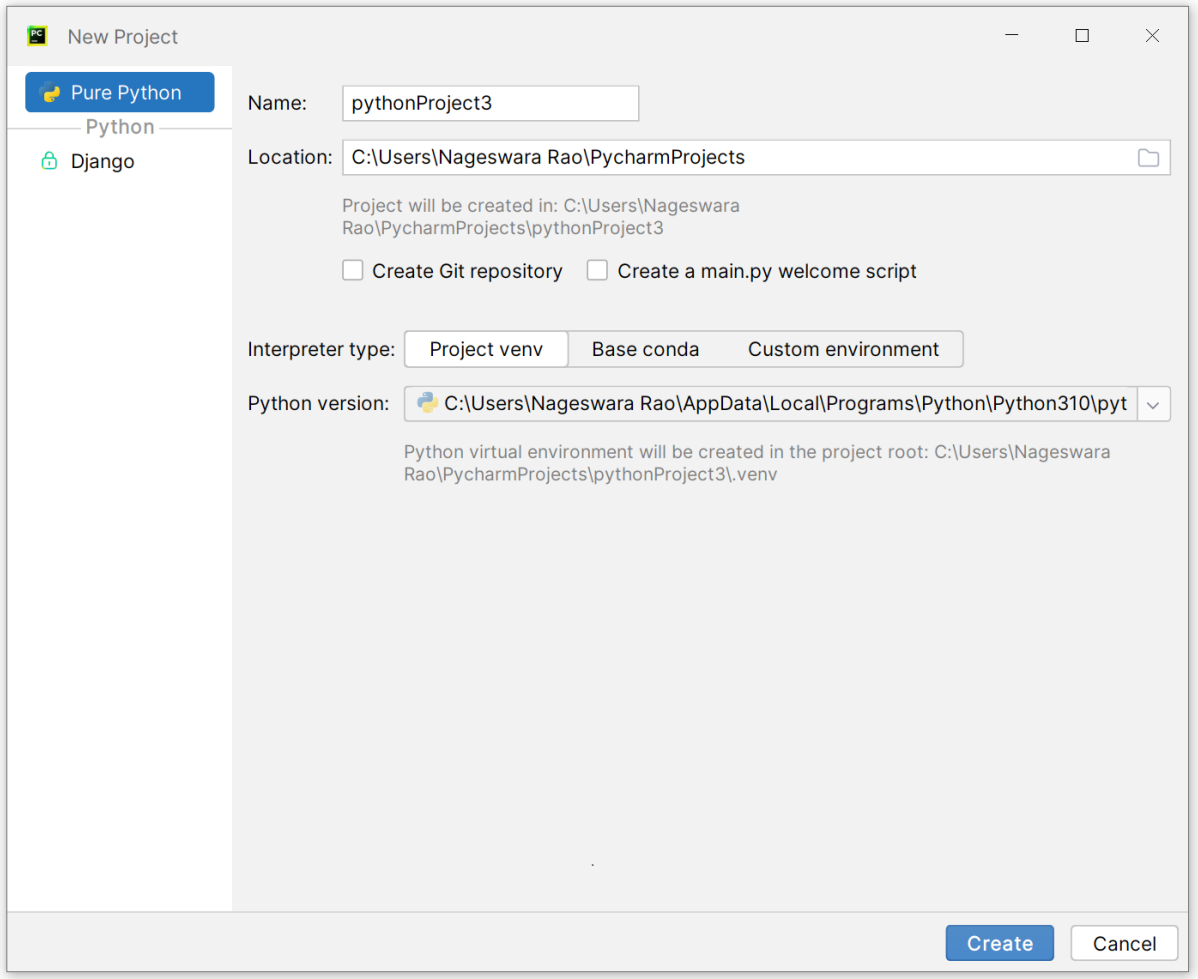
Step 9) In the next screen, we can select “Customize” and then a Color theme either “Darcula” or “IntelliJ Light”. Darcula theme sets background color dark and IntelliJ Light theme sets it to light color. Then click on “Skip Remaining and Set Defaults” button.



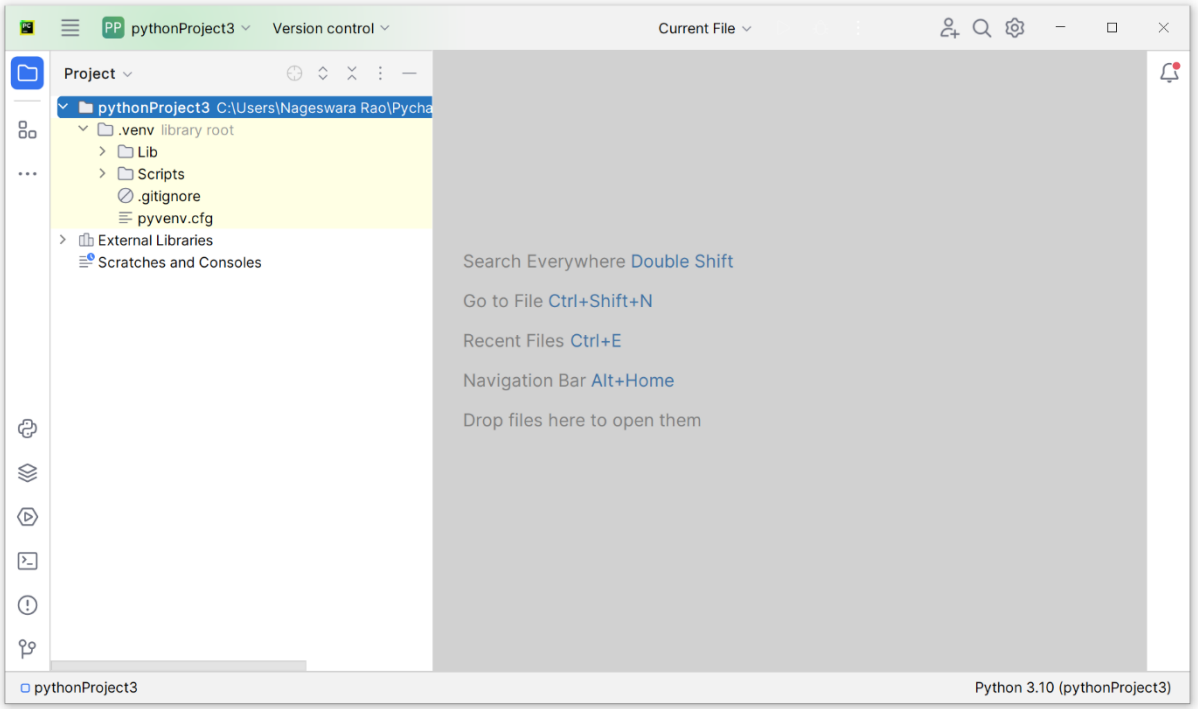
Step 10) In the next screen, click on ‘Projects’ and then click on ‘New Project’.



Step 11) In the next screen, the project title, project directory location along with Python version being used for the project – all these details are displayed. To give a new title for the project, we can retype it. Then click “Create” button at the bottom.

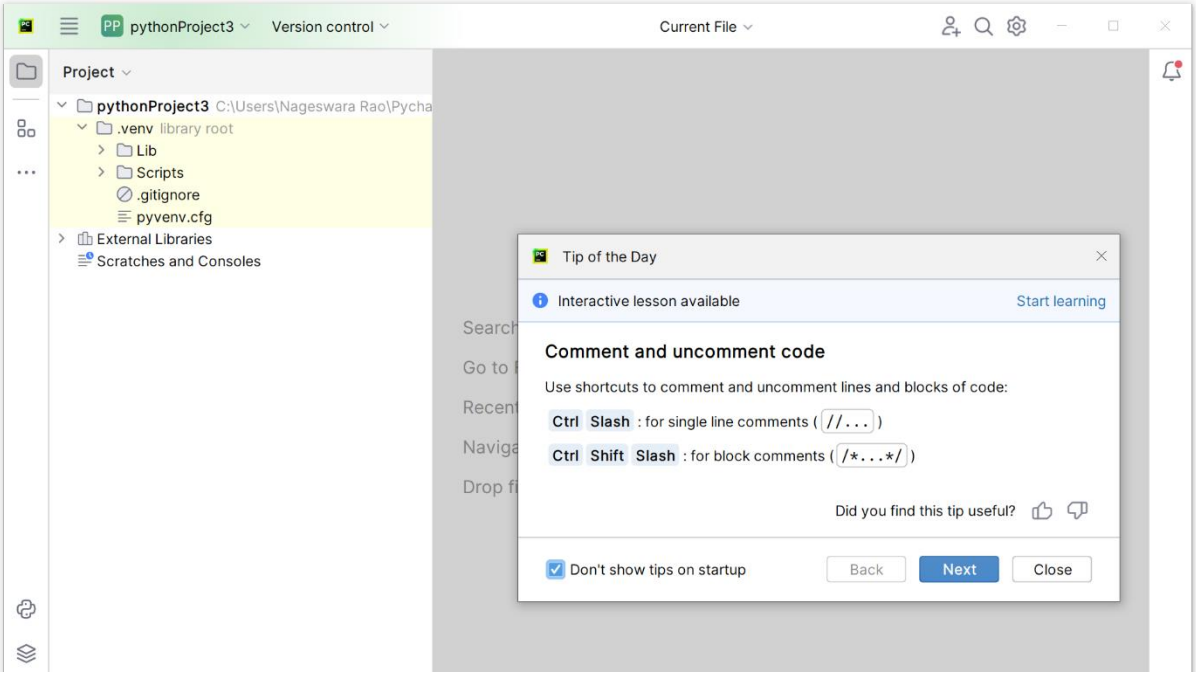


Step 12) A new Project screen will open. A tree structure with the project name in the root will be displayed. Here, we can close by clicking 'X' button at the top right corner.

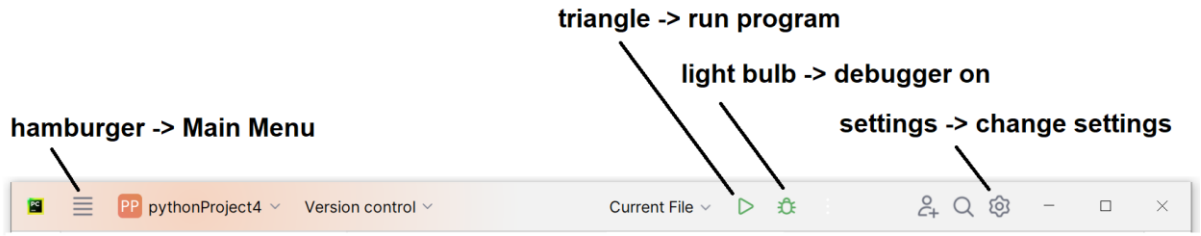


CREATING FIRST PROGRAM

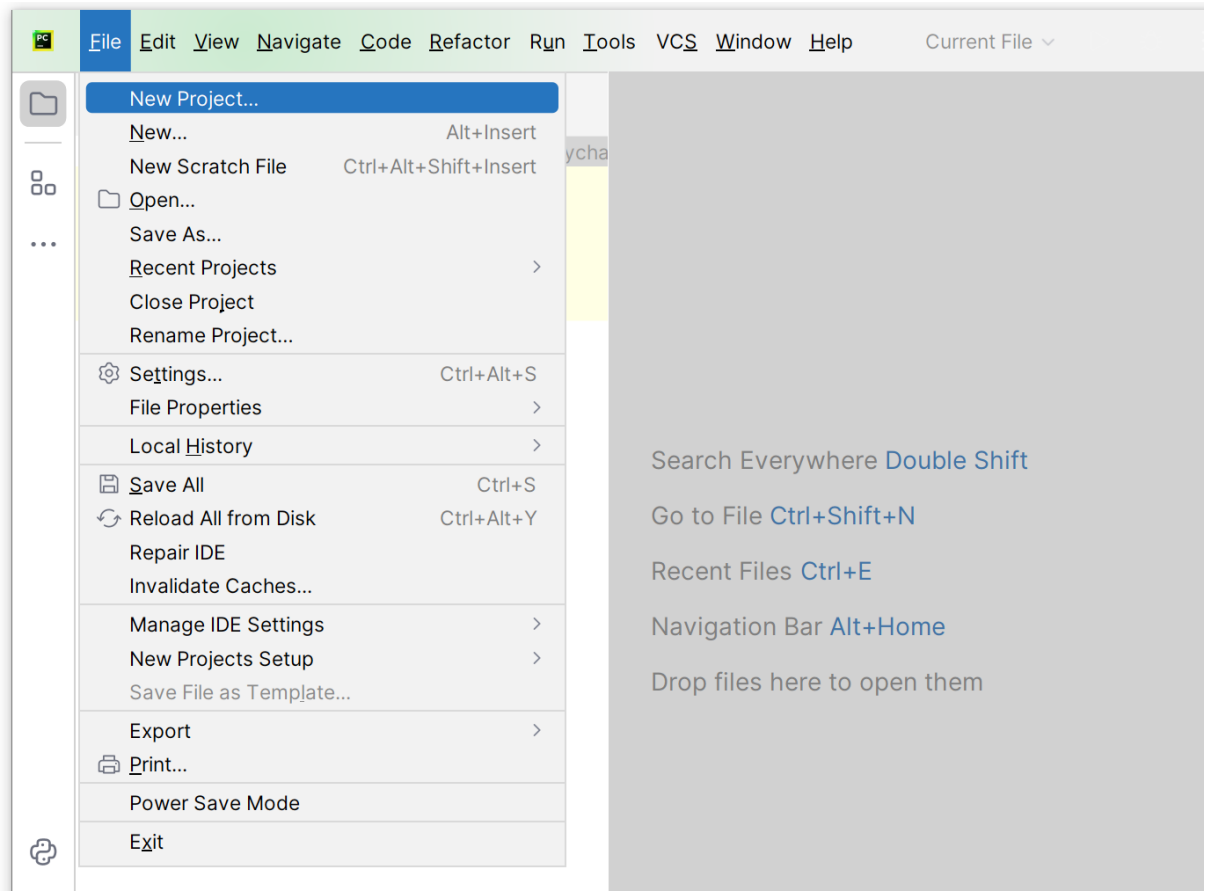
Step 1) Open PyCharm Editor by double clicking the PyCharm icon created on your desktop by the name “PyCharm Community Edition 2023.3.2.” The project screen will be displayed along with ‘Tip of the Day’. Select “don’t show tips on startup” and then click “Close” button.



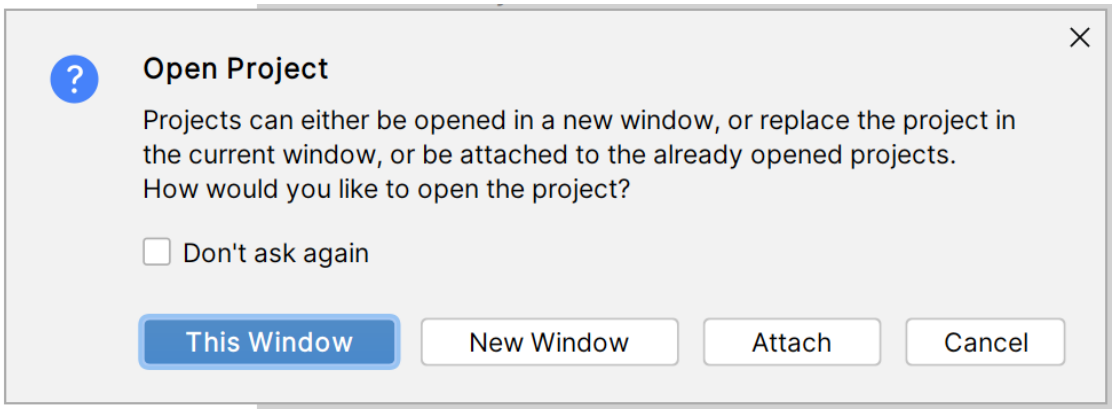
Note: Let us observe the following icons in the top menu bar.



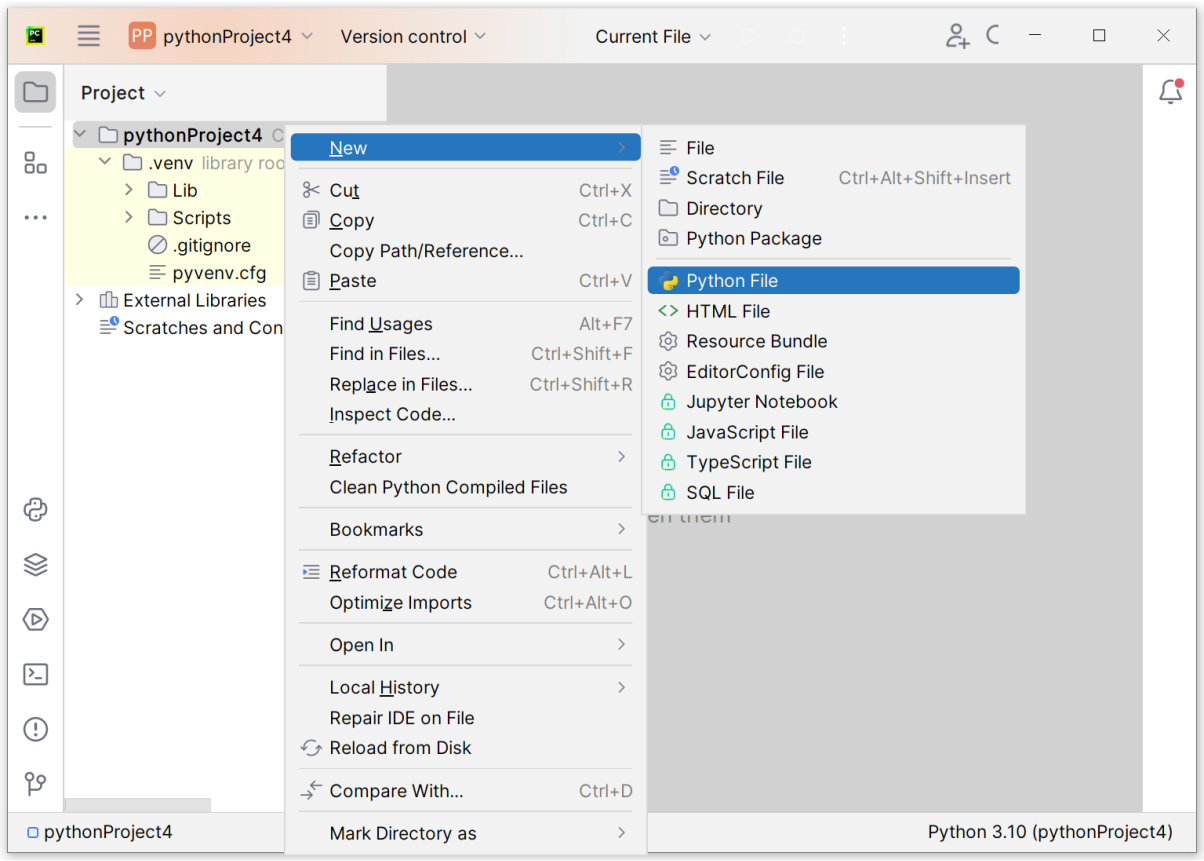
Step 2) In its opening screen, PyCharm always opens a recently closed project. We can continue with this project by adding python programs to it. In case we want to create a new project, then click on hamburger button available at the top left corner in the menu bar. Then click on “File” → “New Project”.



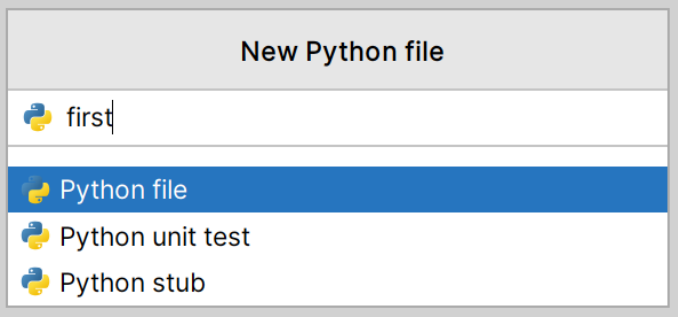
Step 3) Then PyCharm displays a new project title that can be changed as per our wish. Click on "Create" button at the bottom to create the new project with this title. Then it will prompt us whether to open the new project in the same window or another window. Select "This Window" button to open the project in the same window.



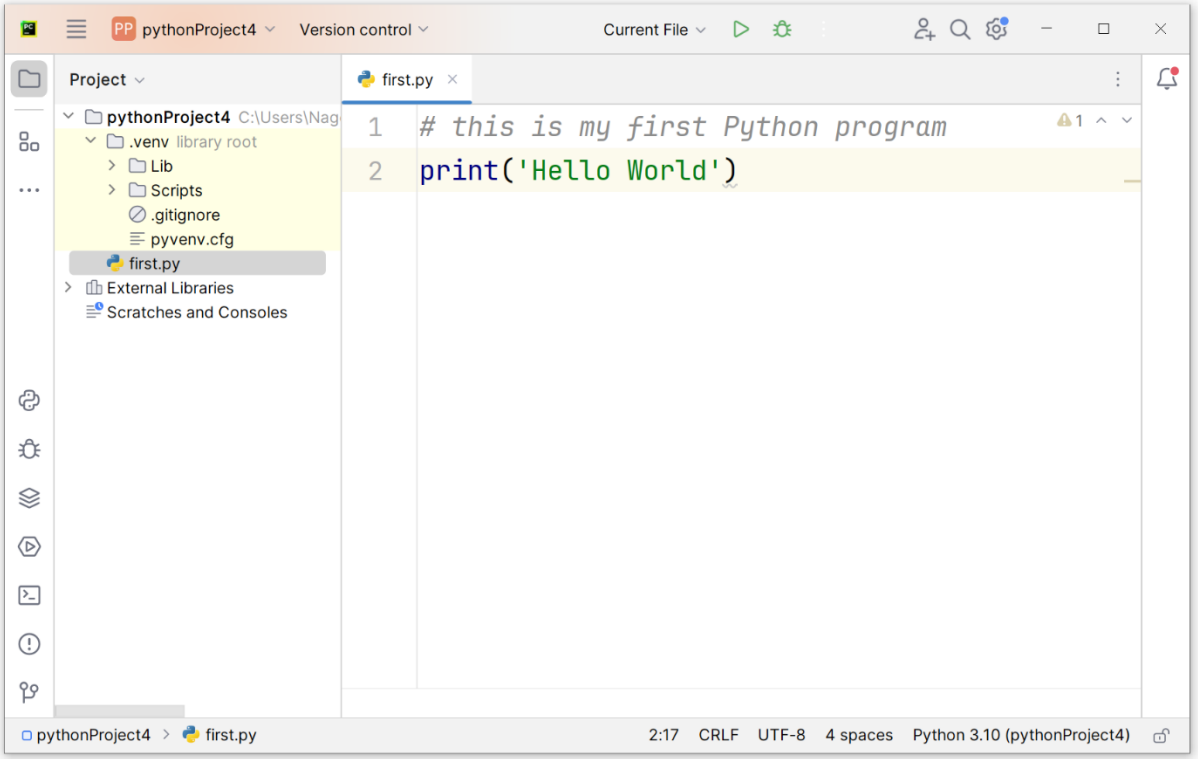
Step 4) Right click on the project name and then select "New". Next, select "Python File".



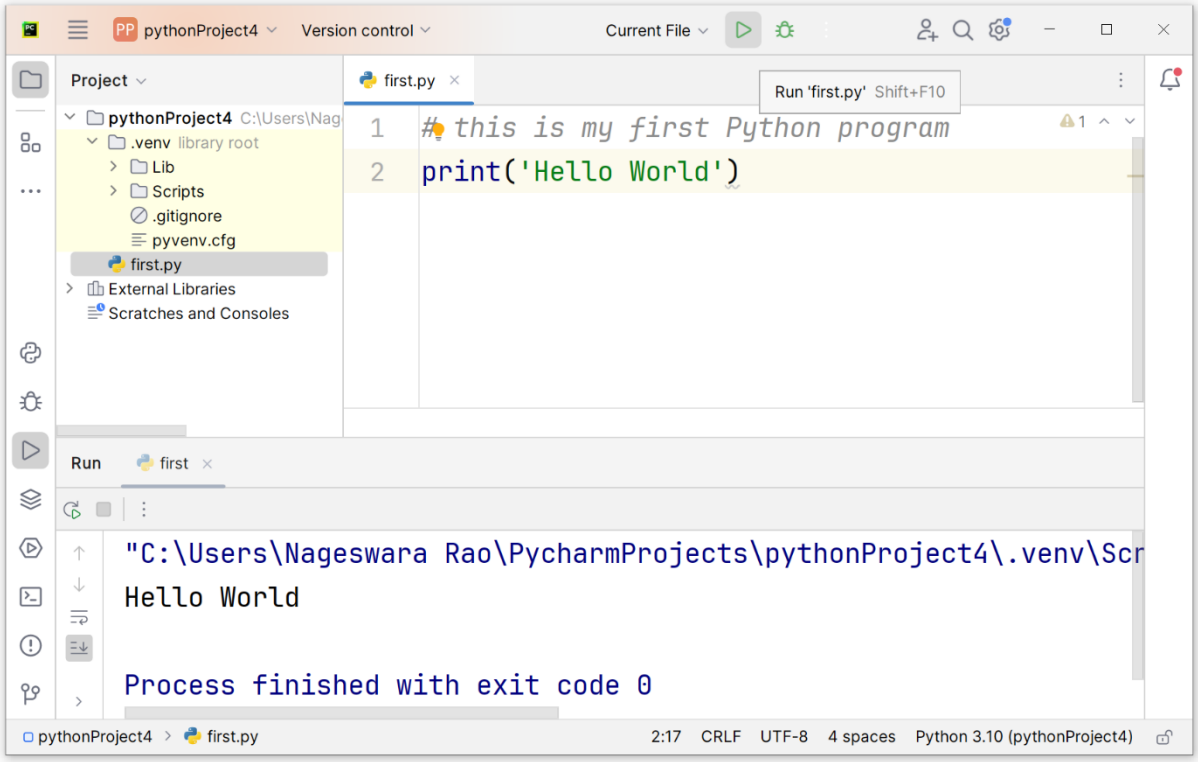
Step 5) A new pop up will appear. Now type the name of the file we want (Here we give “first”) and press <Enter> button.



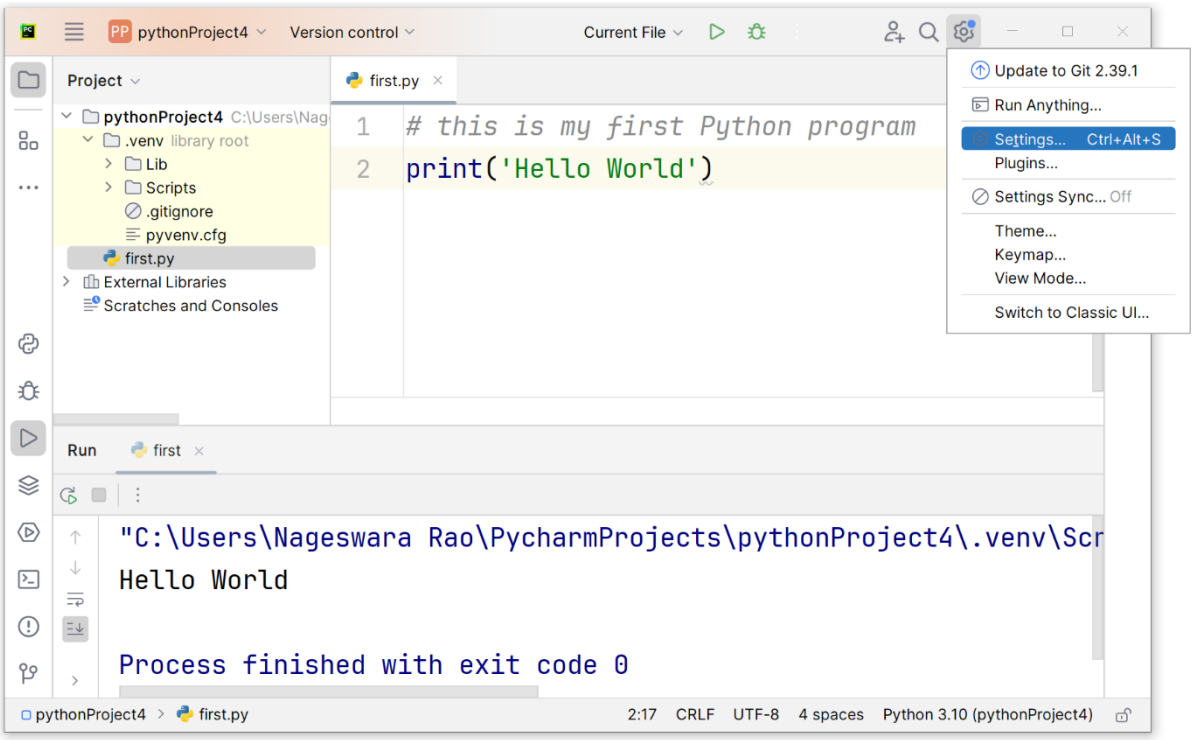
Step 6) It opens a blank page where we can type our program. The program name “first.py” can be seen added to the tree structure. Now let us type a simple program as shown in the screenshot.



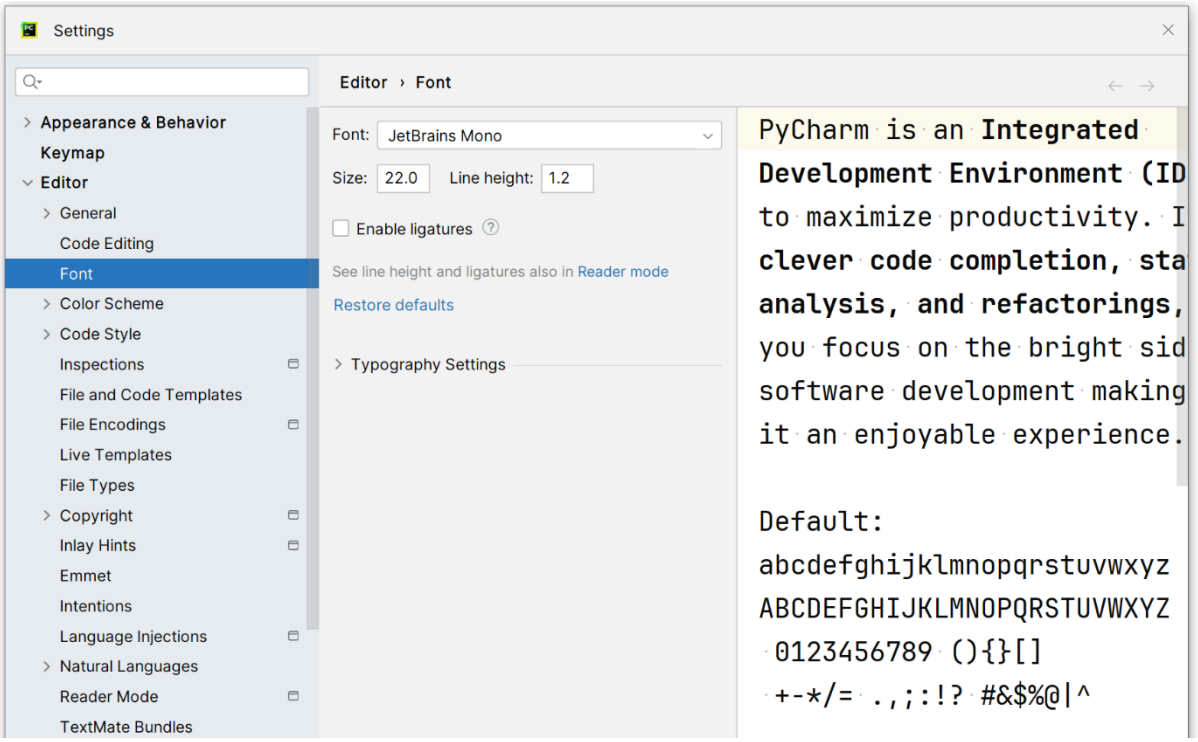
Step 7) To run this program, we have to click on triangle icon in the menu bar and the result will be displayed at the bottom of the screen inside a separate window called “Run window”.



Step 8) We can make changes in the font, size, color theme by clicking the settings icon in the menu bar. Click on settings icon and then select “Settings” in the pull down menu.



Step 9) Select “Font” under “Editor” option, and select the font name, size etc. then click on “OK” button. Similarly, to change the color theme, select “Color Scheme” under “Editor” option.

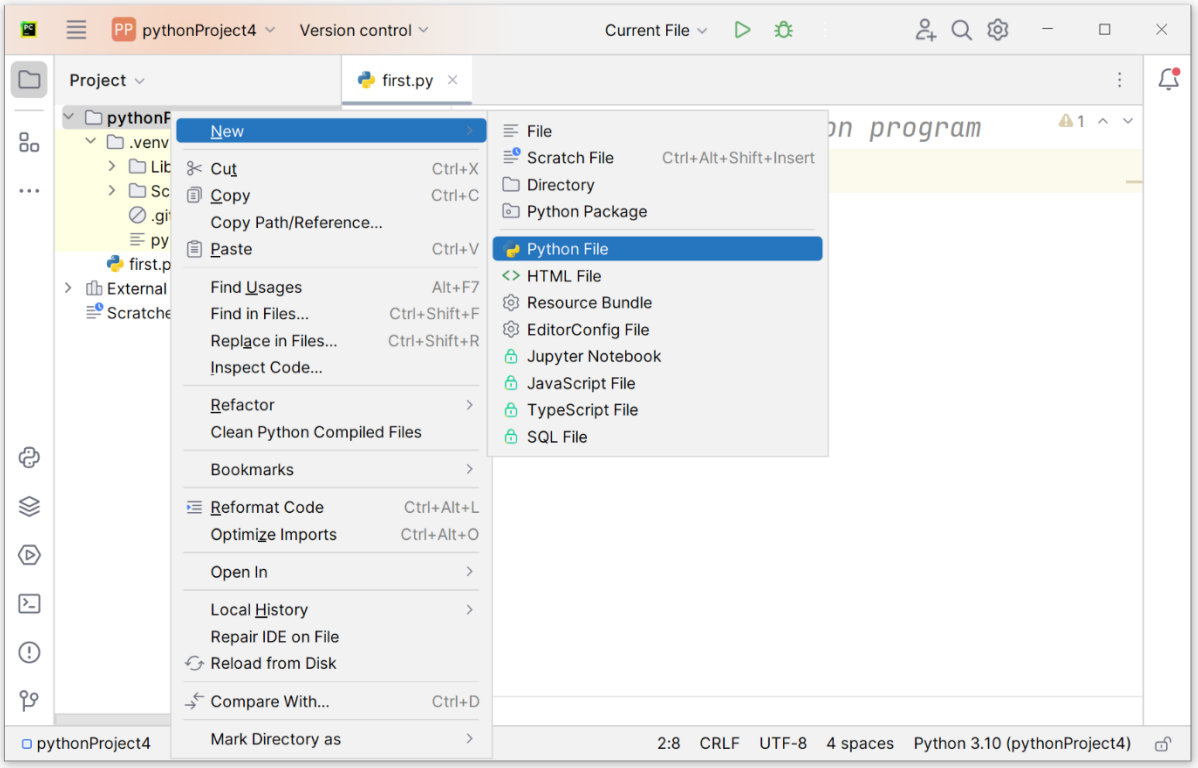


Step 10) To save our project, click on “Hamburger” in the menu bar. Then select “File” and then “Save All”. To terminate PyCharm, select “File” and then “Exit”.

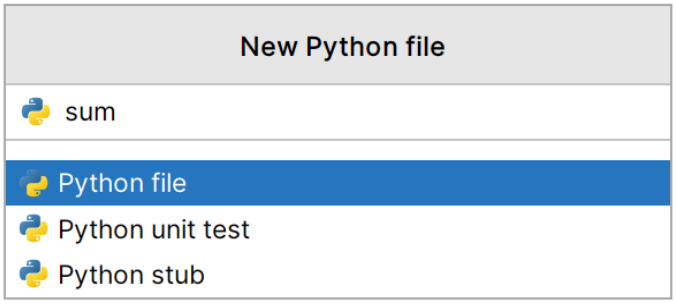
TRACING AND DEBUGGING

It is possible to execute every line of the program and see the output generated at that line. This is called program 'tracing'. When we do tracing, we can understand if the output is correct in that line or not. When the output is not as expected, then we have to remove any errors in that line which is called 'debugging'.

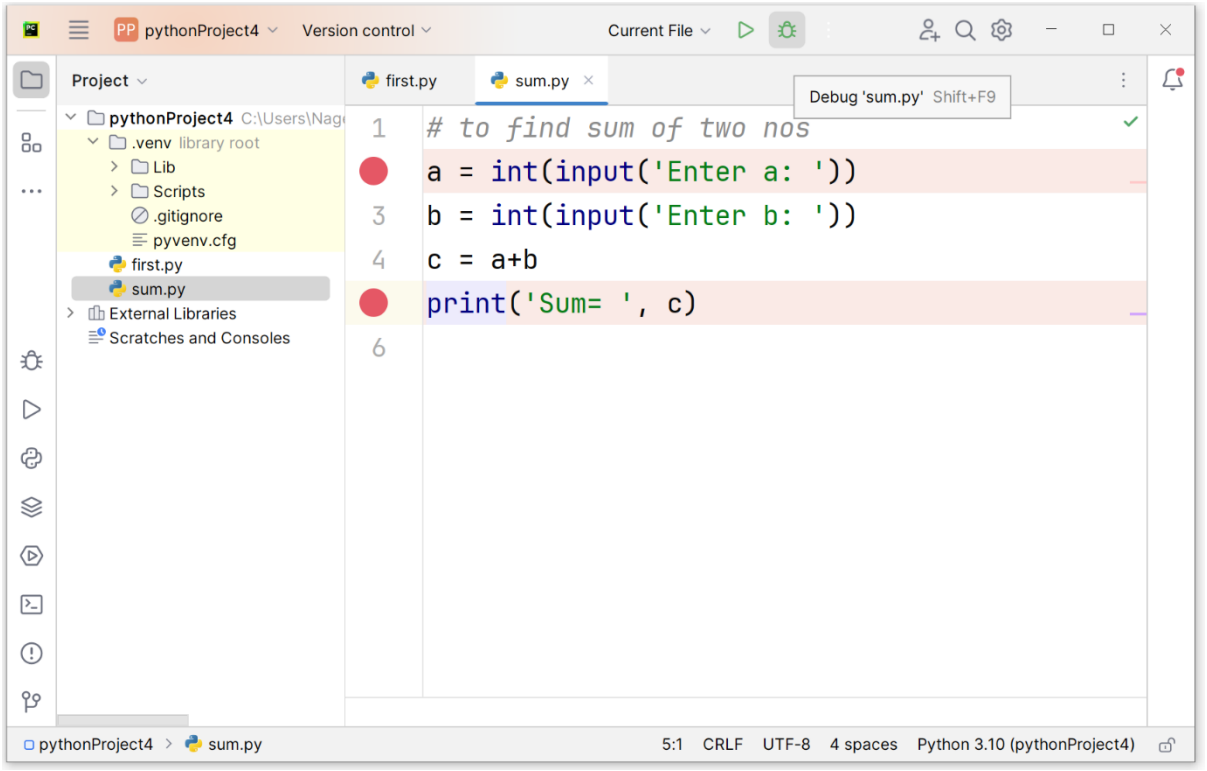
Step 1) Double click to open "PyCharm Community Edition 2023.3.2" shortcut. It will open the recently closed project by default. Let us right click on the project name, then select "New" and then "Python File".



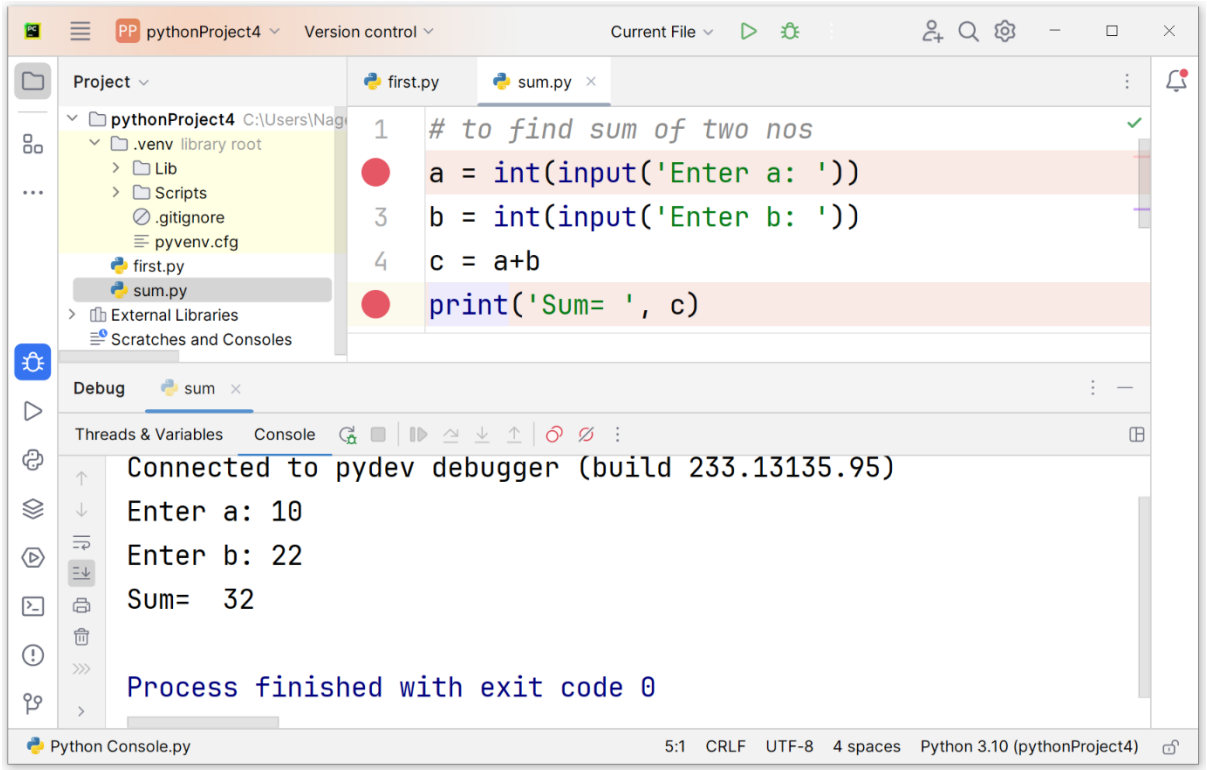
Step 2) Enter the file name as "sum" and then press <Enter> key. This will open a new blank page where we can type sum.py program.



Step 3) Set breakpoints at a line (beginning) and another line (ending) in the program. For this purpose, click on the area at the left side of the program where the line numbers are displayed. When clicked, it displays a red dot.



Step 4) Click on light bulb icon present in the menu bar. This will switch on the debugger mode. When we press “F8” button, it runs the line at the first red dot and displays the result of that line. When we press “F8” again, it goes to the next line, runs it and displays the result. In this manner, we should keep on pressing “F8” to run line by line till we reach the last red dot. Results are displayed in the “Debug window” that opens at the bottom.

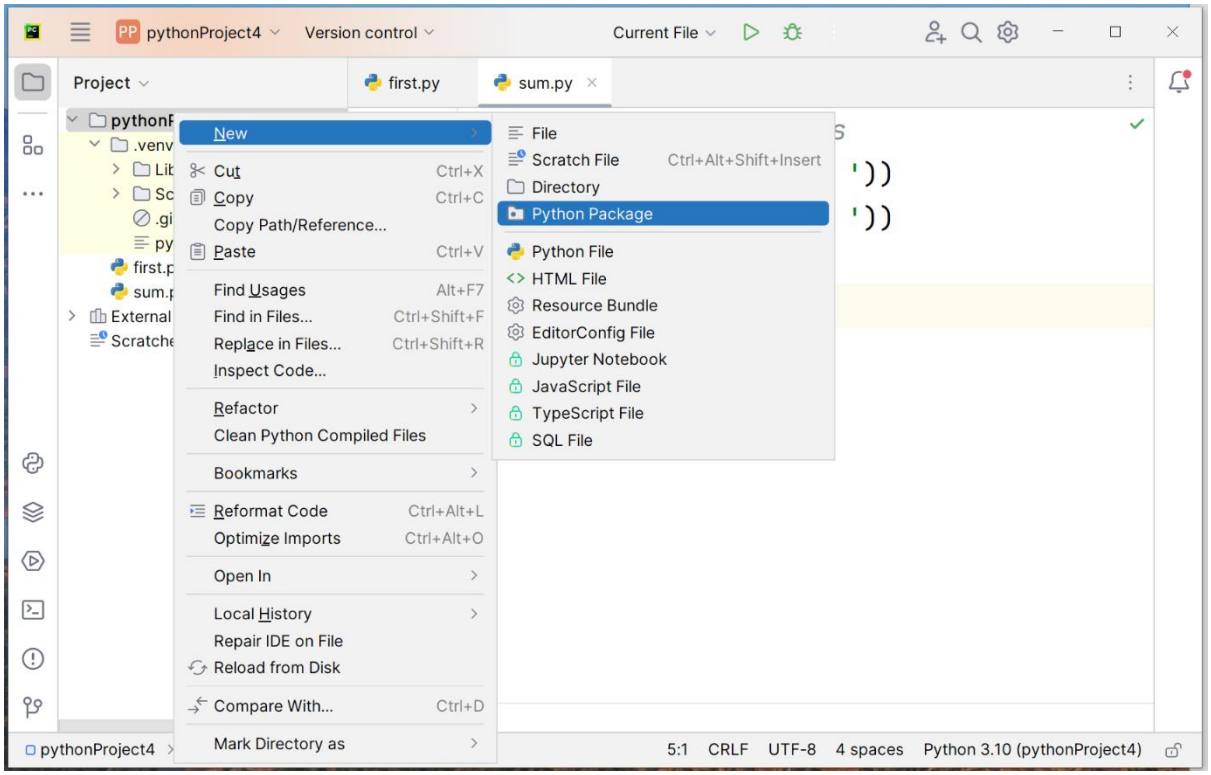


Step 5) To remove the breakpoints (red dots), click on them once again. The project can be closed by clicking on hamburger button and then selecting “File” → “Exit”.

WORKING WITH PACKAGES

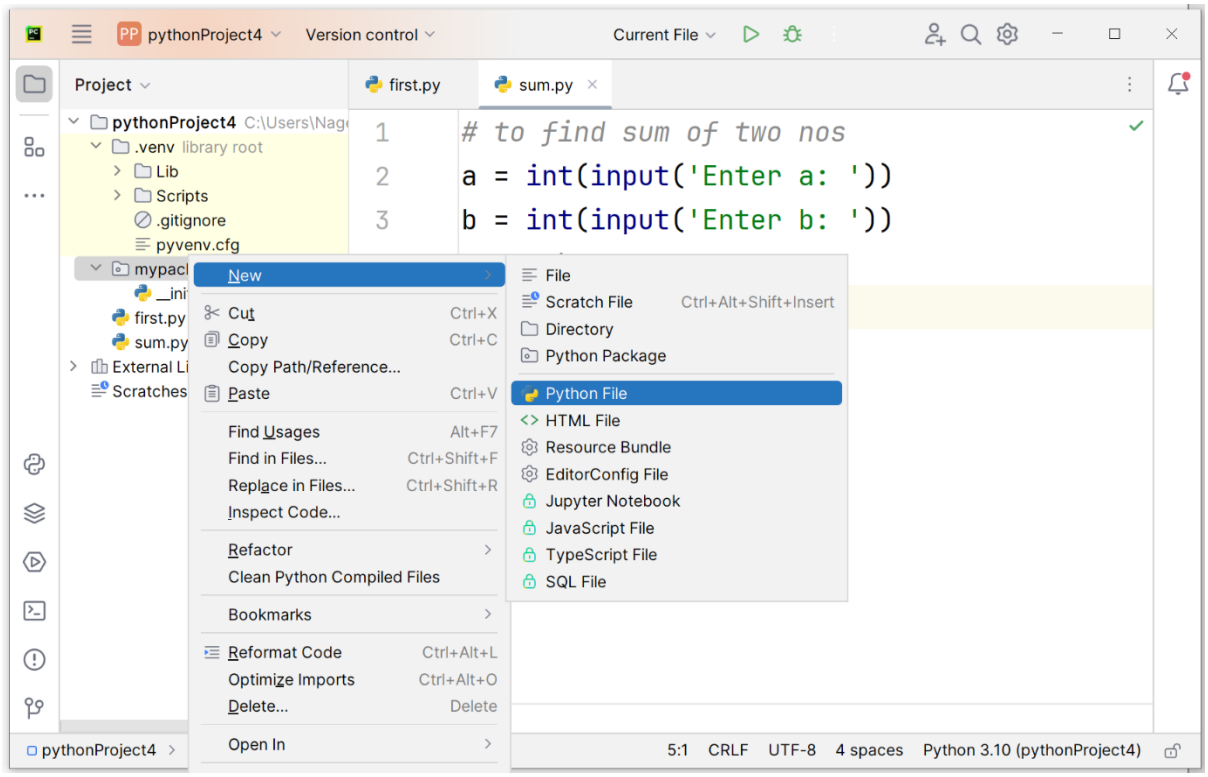
A package is a folder that contains several modules. A module is a python program that contains classes, functions, or any Python objects. Once we create a package, we can import and use it in another program using “import” statement.

Step 1) First, select the project by right clicking on it. Select “New” and then “Python Package”.

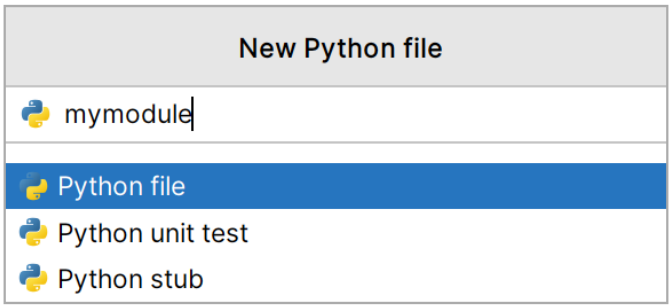


Step 2) Type the package name as “mypackage”. Then press <Enter> key. We can see that “mypackage” has been added in the tree structure of the project.

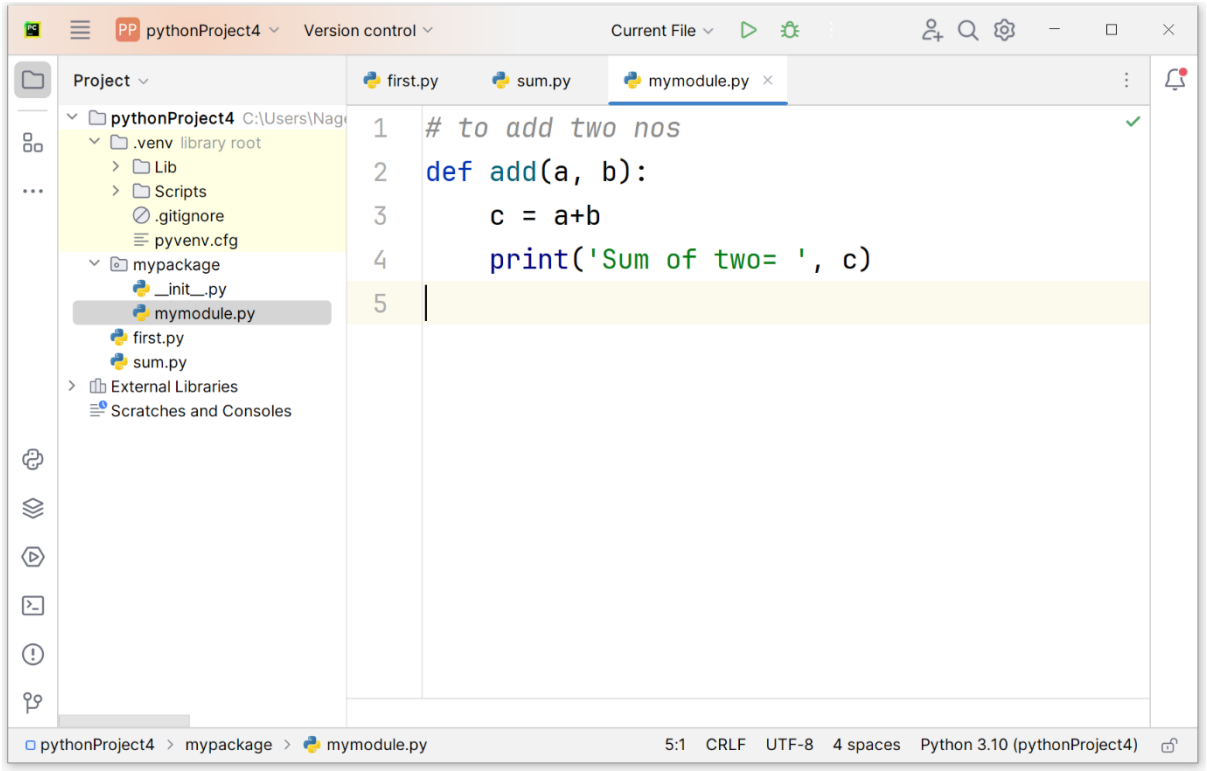
Step 3) Up to now, we created the package (or a new folder). Now, we have to add python programs to it. These python programs are called modules. For this purpose, right click on “mypackage” and select “New” and then “Python File”.



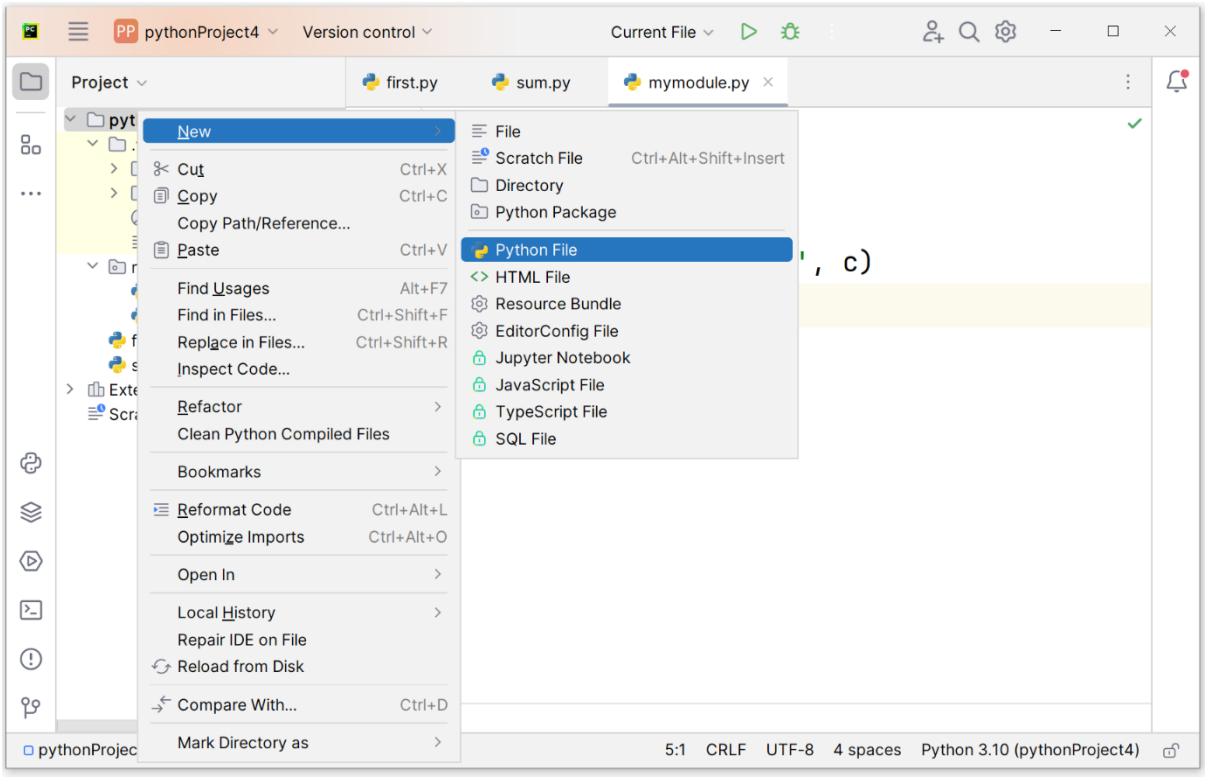
Step 4) Type the file name as “mymodule” and then press <Enter> button.



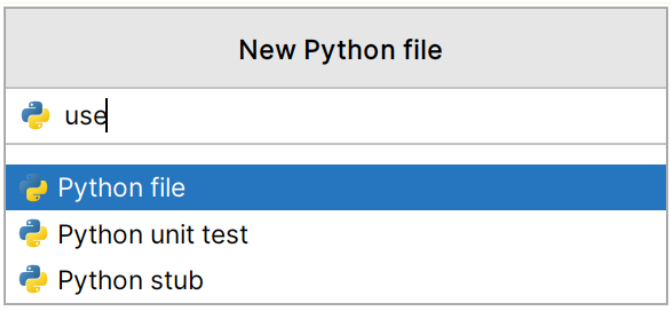
Step 5) A new tab sheet opens by the name “mymodule.py”. Type a function or a class etc. there. For example, we can type a simple add() function code. So far, we created a package by the name “mypackage” and in that a module “mymodule.py” that contains add() function.



Step 6) We can call and use add() function of "mymodule" that is existing inside "mypackage" folder. This can be done from any other program existing in the same project. Let us create another program by the name "use.py" in the project. For this purpose, right click on the project name, select "New" and then "Python File".

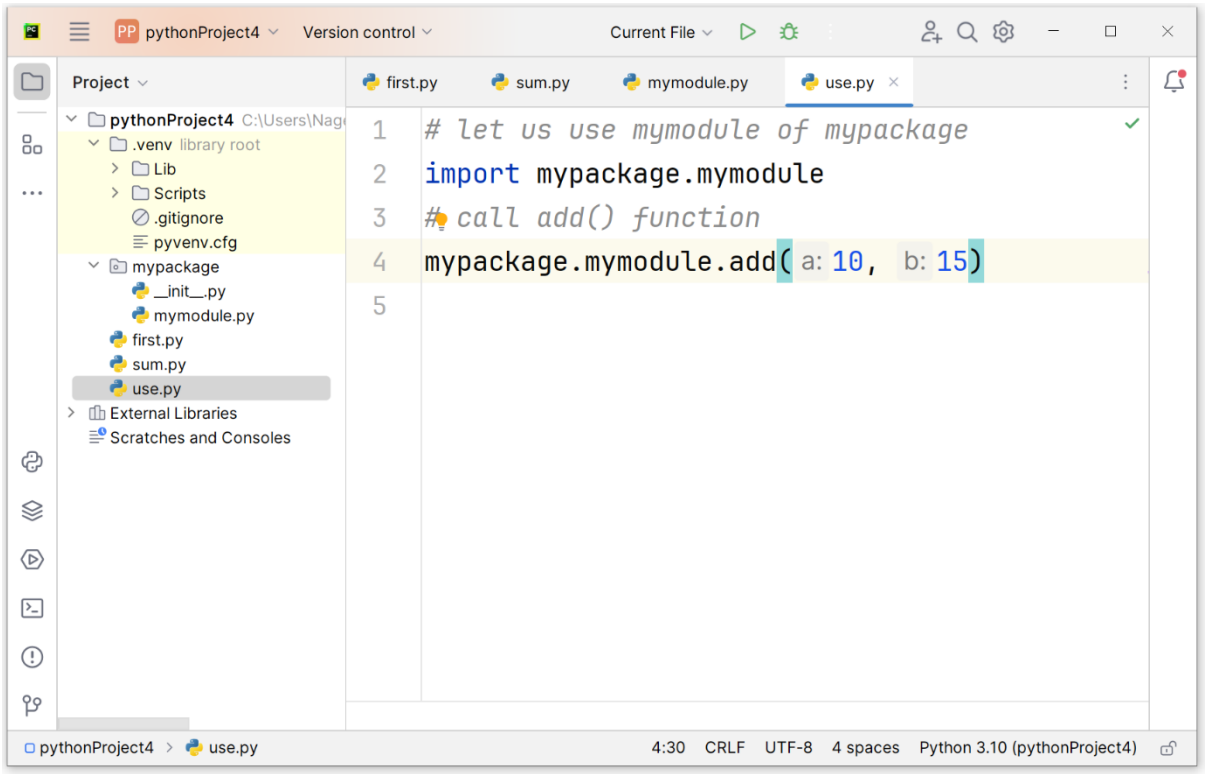


Step 7) Enter the file name as "use" and then press <Enter> key.

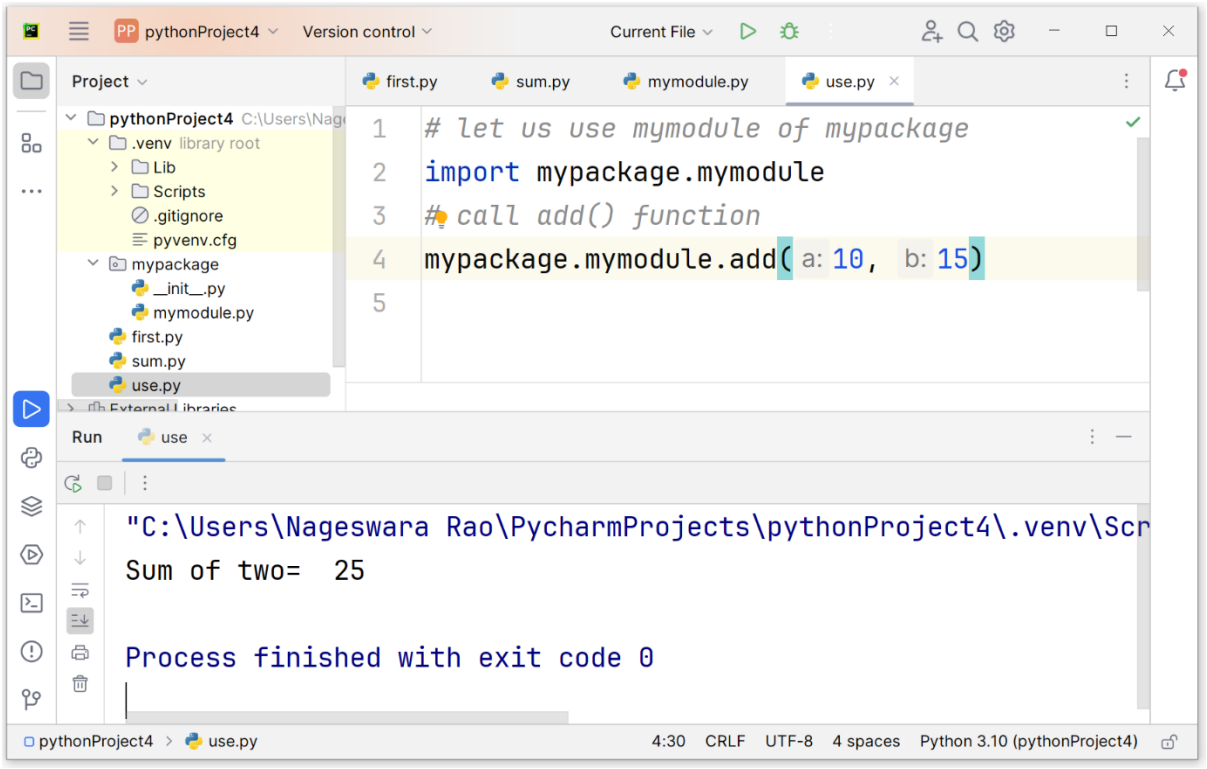


Step 8) We can see "use.py" appearing in the new tab sheet. Let us type a program that uses the "mypackage". For this purpose, we have to import the package using any of the following 4 ways:

- import mypackage.mymodule
  Then call the function as: mypackage.mymodule.add(10, 15)
- import mypackage.mymodule as mm
  Then call the function as: mm.add(10, 15)
- from mypackage.mymodule import *
  Then call the function as: add(10, 15)
- from mypackage.mymodule import add
  Then call the function as: add(10, 15)



Step 9) In the menu bar, click on the triangle icon to run the "use.py" program.



Step 10) To delete any file from the project, we can select that file and then press <Delete> key on the key board. Then click on “OK”. Another way is to select that file then click on “Delete” option from the menu. We can see that file is removed from the project.