<https://code.visualstudio.com/docs/python/python-tutorial>

Linux/macOS: open a Terminal Window and type the following command:

python3 --version

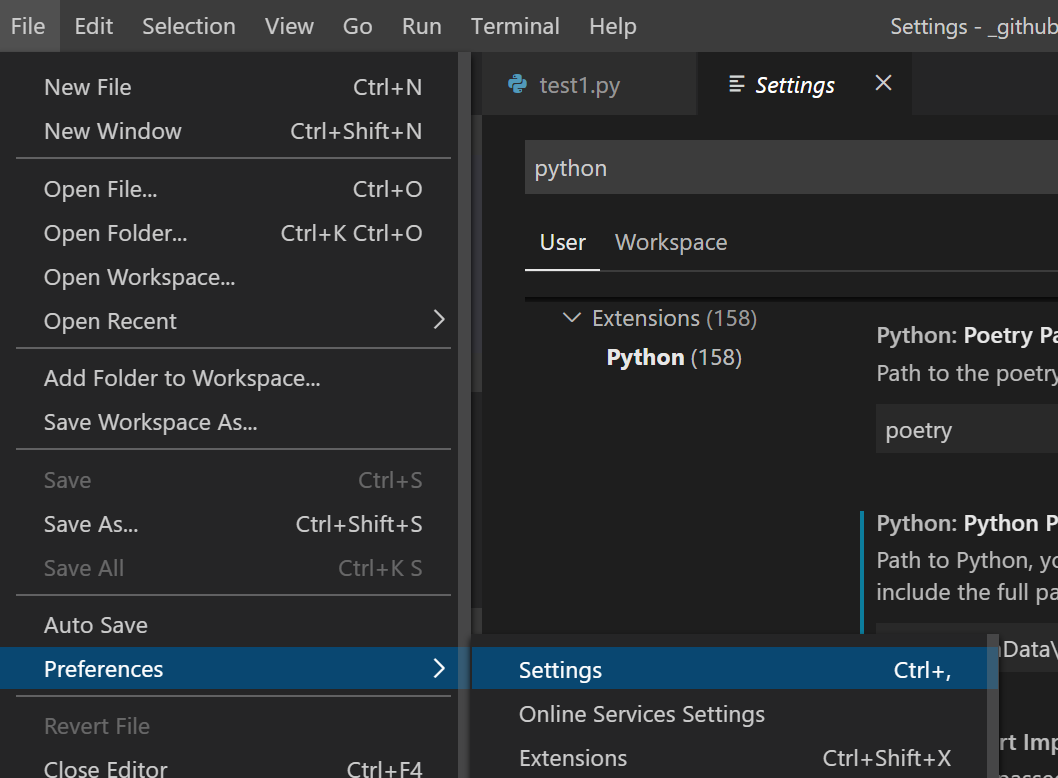
Windows: open a command prompt and run the following command:

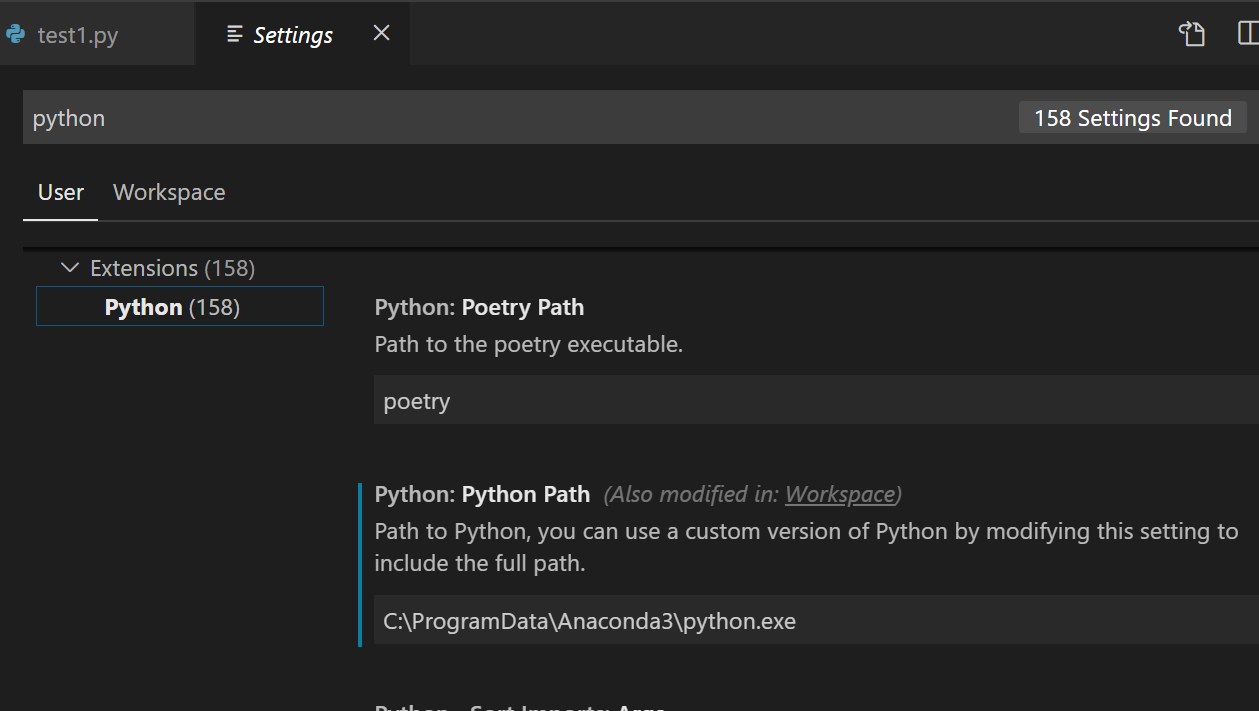
py -3 --version

Command Palette (Ctrl+Shift+P), start typing the Python: Select Interpreter

This modify the python path parameter in the workspace

We can copy the path in the user setting also





<https://stackoverflow.com/questions/29987840/how-to-execute-python-code-from-within-visual-studio-code>

[https://ipython.readthedocs.io/en/stable/interactive/tutorial.html#startup-files](https://ipython.readthedocs.io/en/stable/interactive/tutorial.html" \l "startup-files)

If you want some code to be run at the beginning of every IPython session, the easiest way is to add Python (.py) or IPython (.ipy) scripts to your profile\_default/startup/ directory. Files here will be executed as soon as the IPython shell is constructed, before any other code or scripts you have specified. The files will be run in order of their names, so you can control the ordering with prefixes, like 10-myimports.py.

D:\\_github\lucpython1>ipython locate

C:\Users\lucmi\.ipython

D:\\_github\lucpython1>ipython profile create

[ProfileCreate] Generating default config file: 'C:[\\Users\\lucmi\\.ipython\\profile\_default\\ipython\_config.py](smb://Users//lucmi//.ipython//profile_default//ipython_config.py)'

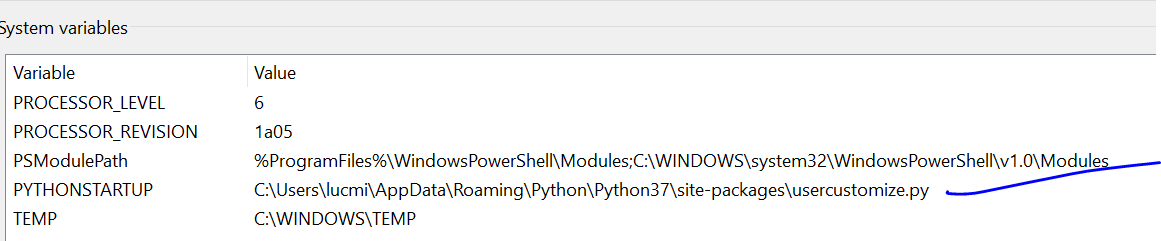
added file

20\_usercustomize.py

[https://docs.python.org/3/tutorial/appendix.html#the-interactive-startup-file](https://docs.python.org/3/tutorial/appendix.html" \l "the-interactive-startup-file)

When you use Python interactively, it is frequently handy to have some standard commands executed every time the interpreter is started. You can do this by setting an environment variable named [PYTHONSTARTUP](https://docs.python.org/3/using/cmdline.html" \l "envvar-PYTHONSTARTUP) to the name of a file containing your start-up commands.

THIS WORKS , the rest not sure....



Python provides two hooks to let you customize it: sitecustomize and usercustomize. To see how it works, you need first to find the location of your user site-packages directory. Start Python and run this code:

>>>

**>>> import** **site**

**>>>** site.getusersitepackages()

'/home/user/.local/lib/python3.5/site-packages'

Now you can create a file named usercustomize.py in that directory and put anything you want in it. It will affect every invocation of Python, unless it is started with the [-s](https://docs.python.org/3/using/cmdline.html" \l "cmdoption-s) option to disable the automatic import.

sitecustomize works in the same way, but is typically created by an administrator of the computer in the global site-packages directory, and is imported before usercustomize. See the documentation of the [site](https://docs.python.org/3/library/site.html" \l "module-site) module for more details.

----------------------------------------

import sys

print (sys.path)

>>> import site

>>> site.getusersitepackages()

'C:[\\Users\\lucmi\\AppData\\Roaming\\Python\\Python37\\site-packages](smb://Users//lucmi//AppData//Roaming//Python//Python37//site-packages)'

I created this arborescence

<https://ipython.readthedocs.io/en/stable/config/intro.html>

[https://ipython.readthedocs.io/en/stable/config/intro.html#ipythondir](https://ipython.readthedocs.io/en/stable/config/intro.html" \l "ipythondir)

ipython profile create

D:\\_github\lucpython1>ipython profile create

[ProfileCreate] Generating default config file: 'C:\\Users\\lucmi\\.ipython\\profile\_default\\ipython\_config.py'

[https://docs.python.org/3/tutorial/appendix.html#the-interactive-startup-file](https://docs.python.org/3/tutorial/appendix.html" \l "the-interactive-startup-file)

When you use Python interactively, it is frequently handy to have some standard commands executed every time the interpreter is started. You can do this by setting an environment variable named [PYTHONSTARTUP](https://docs.python.org/3/using/cmdline.html" \l "envvar-PYTHONSTARTUP) to the name of a file containing your start-up commands.

<https://docs.python.org/3/using/cmdline.html>

**PYTHONSTARTUP**

If this is the name of a readable file, the Python commands in that file are executed before the first prompt is displayed in interactive mode. The file is executed in the same namespace where interactive commands are executed so that objects defined or imported in it can be used without qualification in the interactive session.

created

sitecustomize.py

<https://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data>

A best practice among Python developers is to avoid installing packages into a global interpreter environment. You instead use a project-specific virtual environment that contains a copy of a global interpreter. Once you activate that environment, any packages you then install are isolated from other environments. Such isolation reduces many complications that can arise from conflicting package versions. To create a virtual environment and install the required packages, enter the following commands as appropriate for your operating system

[https://code.visualstudio.com/docs/python/environments#\_global-virtual-and-conda-environments](https://code.visualstudio.com/docs/python/environments" \l "_global-virtual-and-conda-environments)

python -m venv .venvluc