A couple of clarification questions for today's class:

1. One student asked me the relationship among Python, PyCharm, Jupyter Notebook, and GitHub.

Python is a programming language that serves as the foundation. You write the code in Python to analyze data, build models, or perform other computational tasks. It is like the language we speak.

PyCharm is an Integrated Development Environment (IDE) for Python that provides a professional coding environment such as code editor, debugging tools, project navigation, version control integration, etc. It is like RStudio.

Jupyter Notebook is an interactive notebook interface for Python (It also supports R and MATLAB with additional configurations). It lets you write code in small chunks, run them immediately, and see outputs inline (tables, plots, texts). It is great for data science, teaching, or reproducible workflow in research development. It is like Markdown written in RStudio.

Finally, GitHub is cloud-based version control and collaboration platform that stores and shares your code, track changes, enable collaborations, and integrates with both PyCharm and Jupyter. It is like OneDrive for code, with additional features.

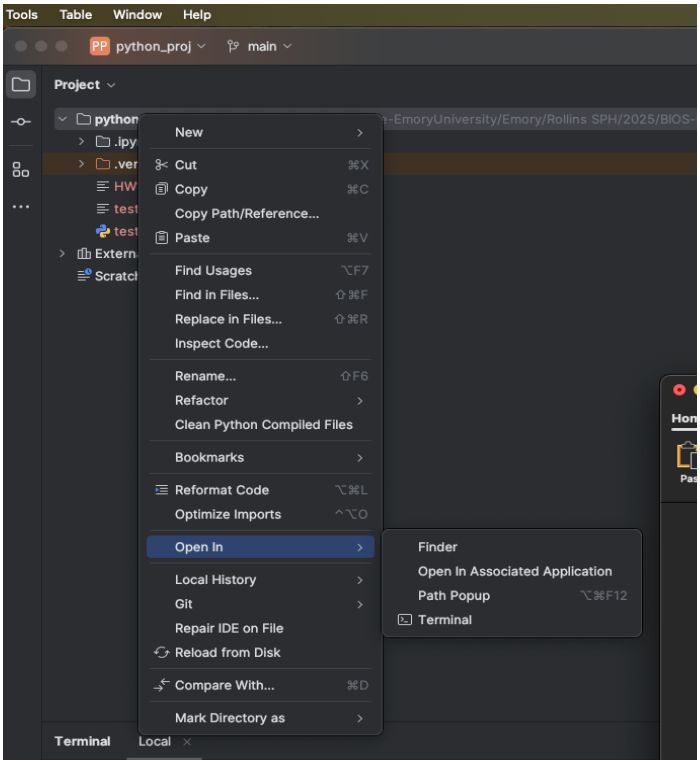
You do not necessarily need GitHub to write python code in PyCharm.

So the relationship is:

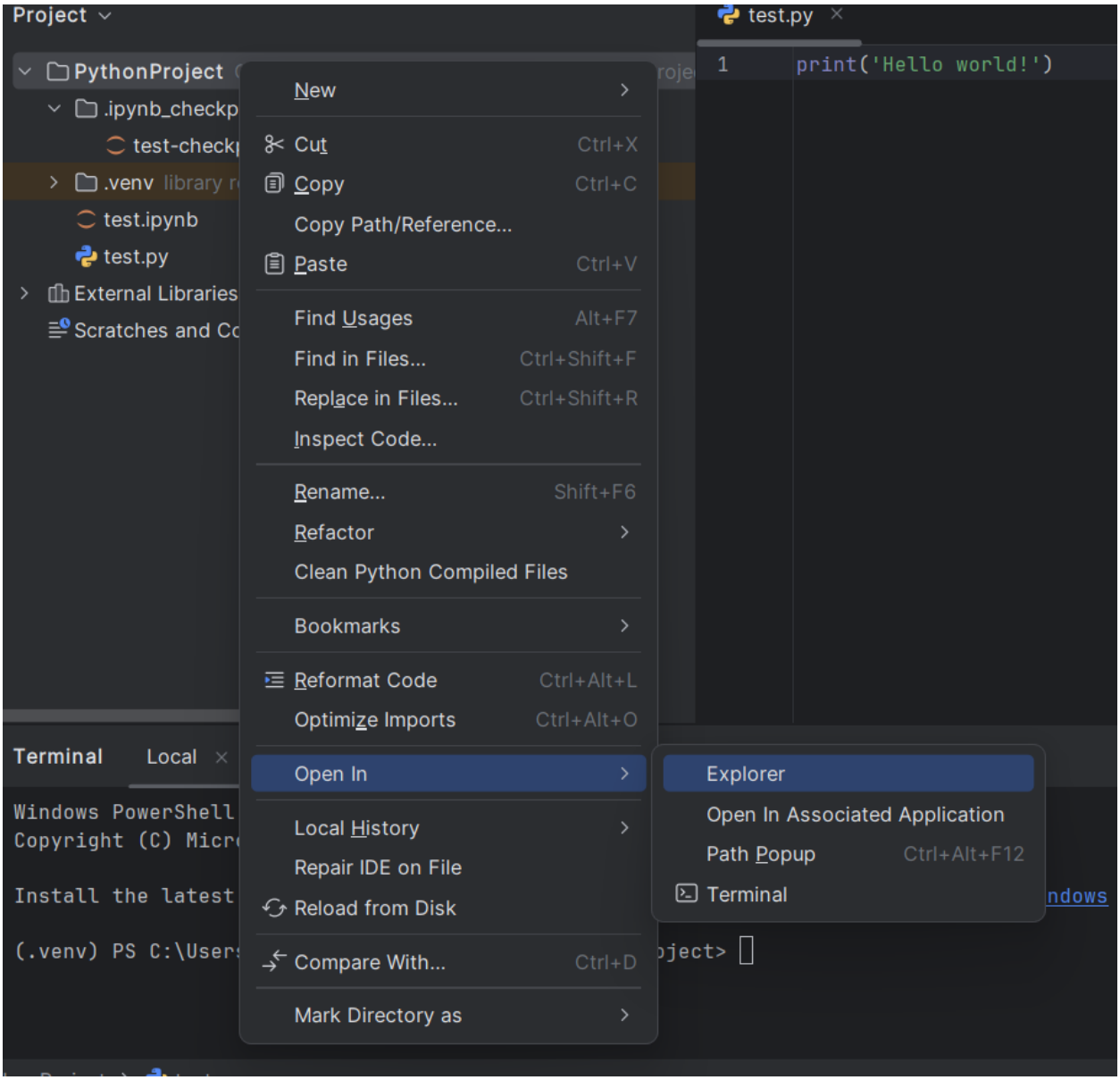
- Python = the language.
- PyCharm & Jupyter = two different tools/environments to write and run Python code.
- GitHub = where you manage and share that code with others.

2. Students are confused about the add changes in the GitHub section. For HW1, if your PyCharm project directory is different from the cloned BIOS-584 directory under GitHub folder, and you have already completed the HW1.ipynb under the PyCharm project directory, all you need to do is to copy your HW1.ipynb and paste it to the directory under the GitHub one. Anything new under GitHub repo will be detected by the GitHub Desktop automatically.
    a. How to identify your PyCharm directory? The one with a yellow arrow.

You can right click on the python project name (python_proj in my case) -> “Open In” -> Finder (for Mac)



Or right-click on the python project name (PythonProject in my case) -> “Open In” -> “Explorer” (for Windows)

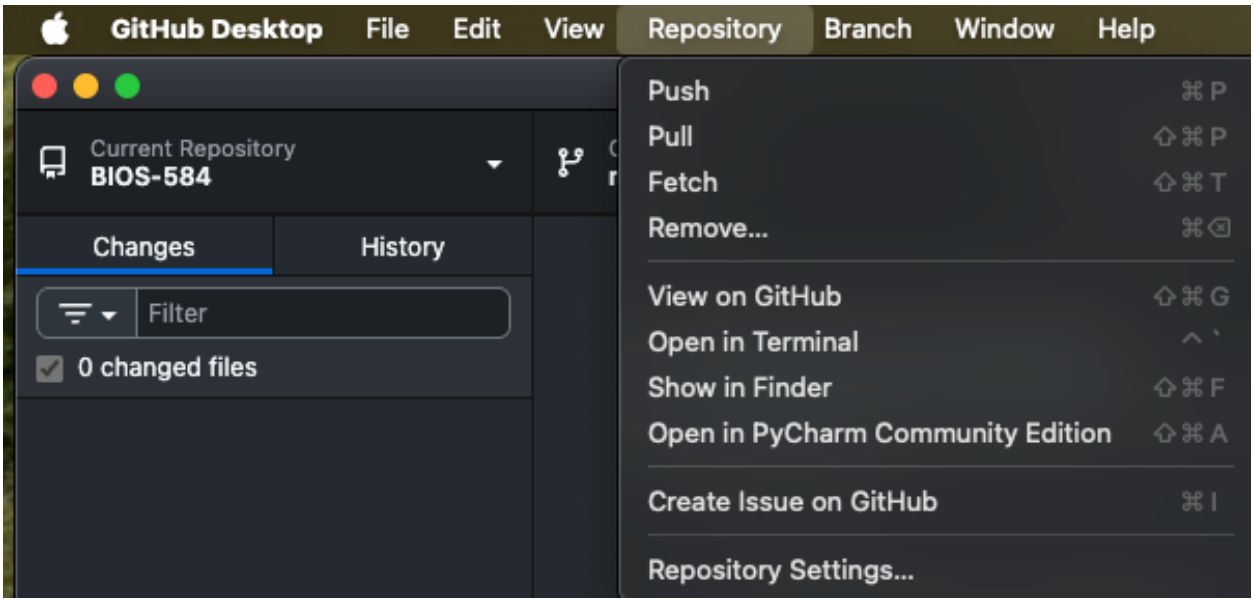


Or type “pwd” in the Terminal, and use go to the corresponding directory (both Windows and Mac)

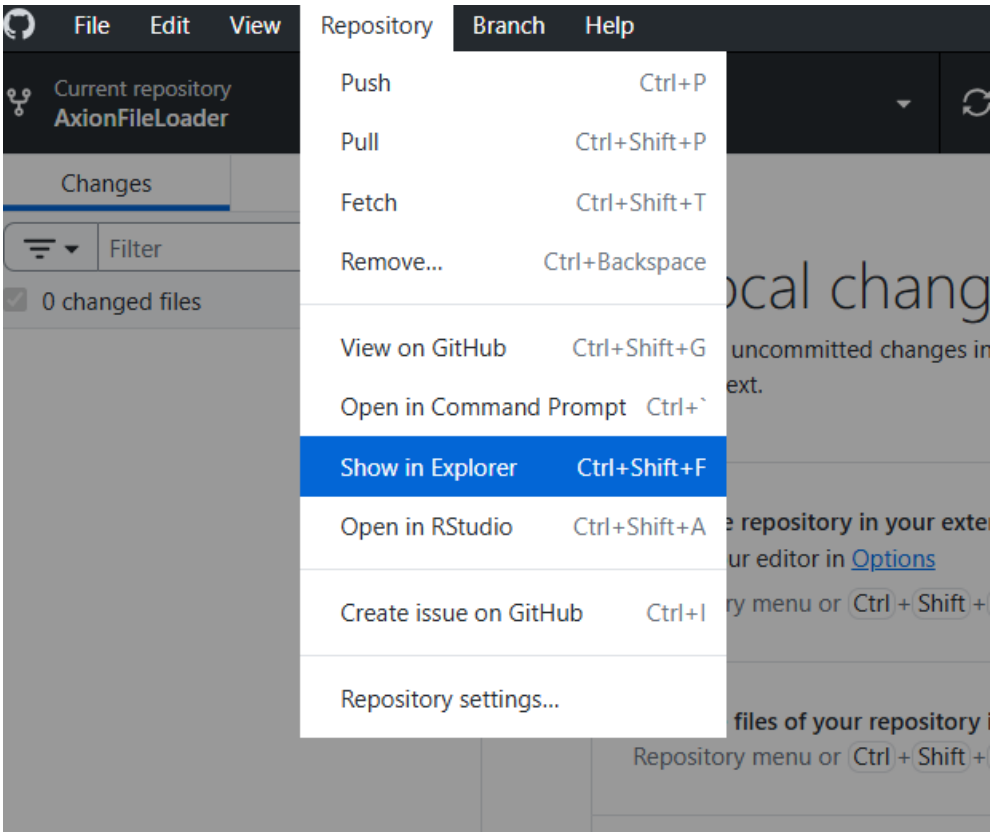
```
(.venv) tma33@BIOR6N700WRXY python_proj % pwd
/Users/tma33/Library/CloudStorage/OneDrive-EmoryUniversity/Emory/Rollins SPH/2025/BIOS-584/python_proj
(.venv) tma33@BIOR6N700WRXY python_proj %
```

b. How to identify your directory of your GitHub repo?

Suppose that you already open your “BIOS-584” repo, Click “Repository” -> “Show in Finder” (for Mac)

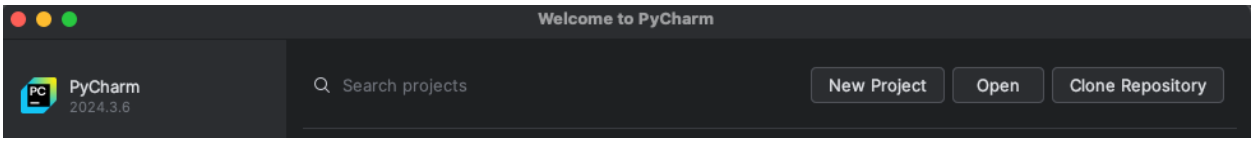


Or Repository -> Show in Explorer (for Windows)

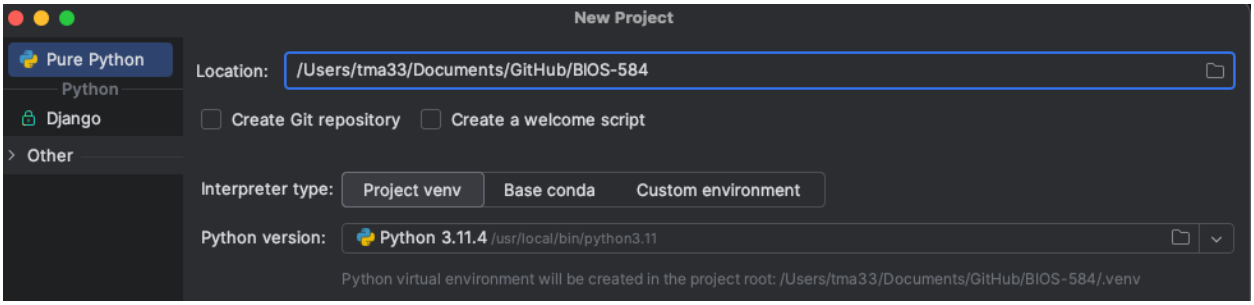


3. Since we introduce PyCharm and GitHub separately, it is more convenient to create a PyCharm project such that any changes you made will be automatically detected by the GitHub Desktop (and to commit a change and push it) instead of manually copying and pasting files all the time.

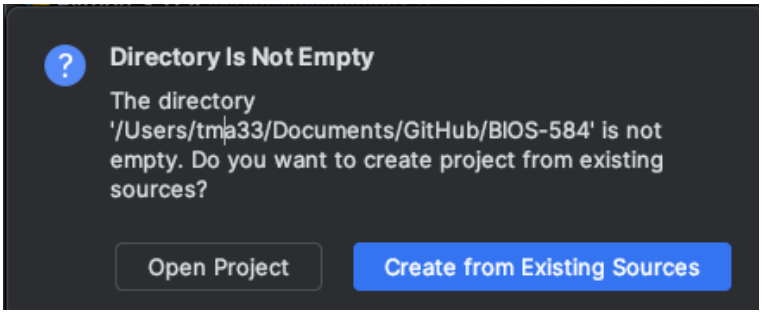
For future work, you can save your Python code under your own GitHub BIOS-584 repo. Suppose that you have already cloned a BIOS-584 repo on your local GitHub Desktop, when you open PyCharm, click “New Project”



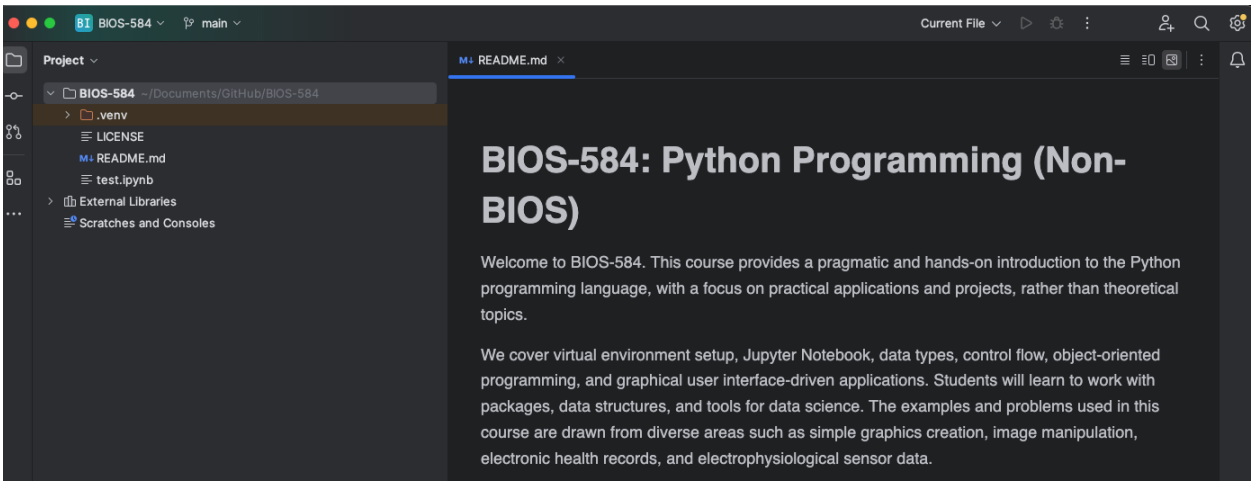
In “Location”, type the directory of the local GitHub repo and click “Create” below (snapshot did not show it).



A warning pops up. You click “Create from Existing Sources”



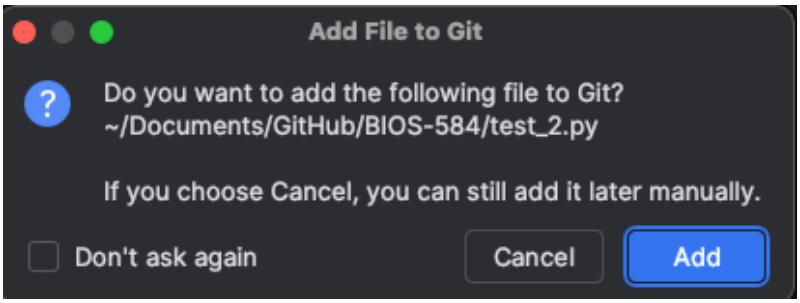
And it will create a new PyCharm project under the GitHub “BIOS-584” repo with a “.venv” folder.



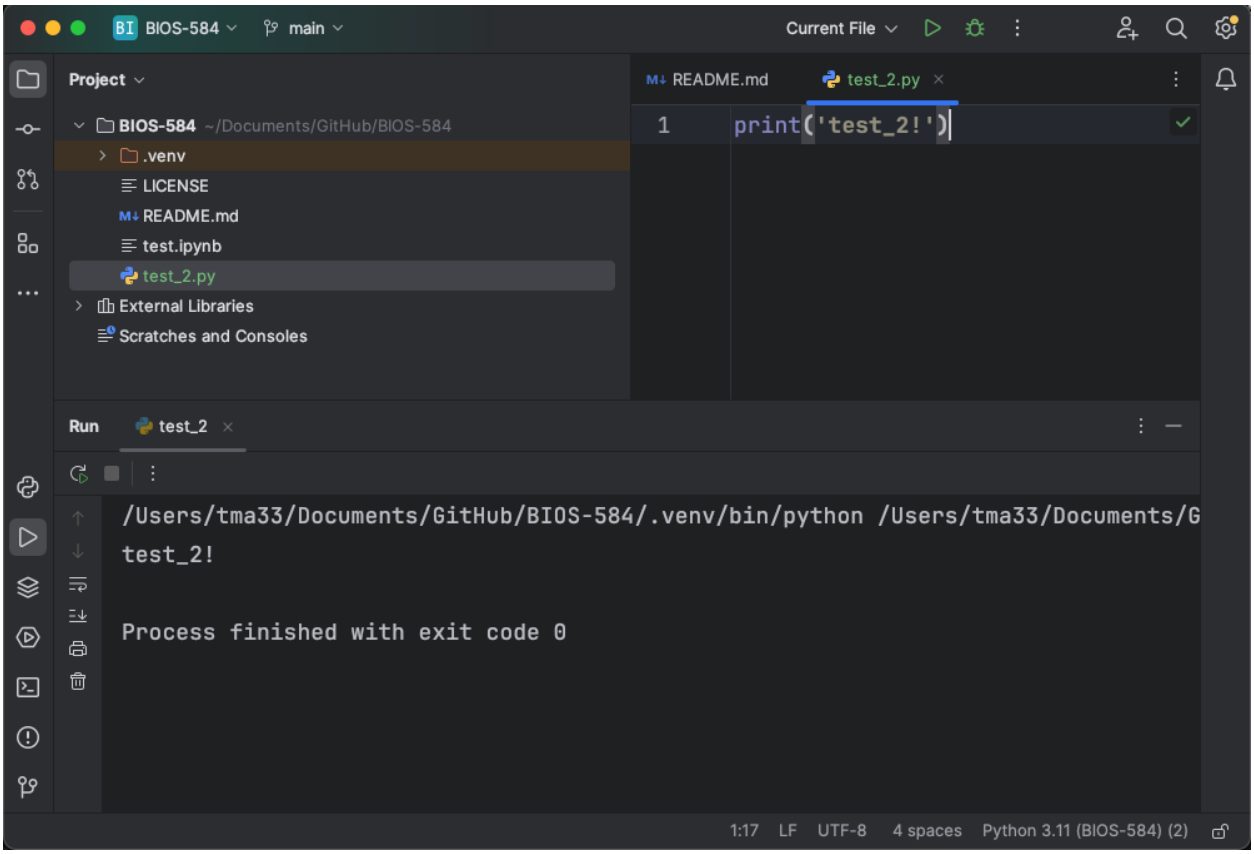
When you create a new python file, PyCharm asks you whether you want to add it to Git, click “Add”.

Every time you wrote something new, the GitHub desktop should detect it immediately.

For example, if I would like to create a new python file named “test_2.py” with a printing message, the PyCharm interface will look like:



By the way, exit code 0 means your code does not have any python-related syntax error.


BIOS-584
main
Current File
Project
BIOS-584 ~/Documents/GitHub/BIOS-584
.venv
LICENSE
README.md
test.ipynb
test_2.py
External Libraries
Scratches and Consoles
README.md
test_2.py
1 print('test_2!')
Run
test_2
/Users/tma33/Documents/GitHub/BIOS-584/.venv/bin/python /Users/tma33/Documents/G
test_2!
Process finished with exit code 0
1:17 LF UTF-8 4 spaces Python 3.11 (BIOS-584) (2)

Meanwhile, the GitHub desktop will look like