

Statistics 21

Python & Other Technologies for Data Science

Vivian Lew, PhD - Wednesday, Week 1

## Say Hello!

You've been randomized into new teams today

You will need to find each other - so name tags are going around

Please take a team photo and choose one person to upload it

We will inventory these tools together

## Recall: Shell Basics

### Opening Terminal on MacOS

- Probably the quickest: Open spotlight with Command + space. Start typing "Terminal". Terminal will appear as the top hit after you type the first few letters. Hit Enter to start.
- Another method: You can open Launchpad from the dock. Click "Other". Click "Terminal".

## Getting Help

- Windows cmd

*help commandname*

- Mac OS

*commandname --help*

Type `q` to exit help. Hit the space bar to scroll to the next page.

Using the command line

shell/terminal/console

We typically interact with computers through the use of GUIs (Graphical User Interfaces)

But there are times when it is better to interact directly and bypass these windows

- quicker/more efficient task completion
- easier/more efficient task automation
- lower overhead lower resource usage for the same task

Your shell skills are a good foundation for more advanced tasks

## Shell Basics: Navigation

- `pwd` will tell you where you are currently located. (present working directory)
- `cd` is the command to change your directory

Wherever you are, you can switch to your home directory with `cd`

## Shell Basics: Navigation

- Directories are listed in a hierarchy. For example, you may decide to store content for this class in:

Desktop/classes/stats21

- Let's assume this is your present working directory (the response when `pwd` is typed at the prompt).  
`cd homework` will change to the directory `homework` **IF** it exists in your current directory. If there is a `homework` folder inside the `stats21` folder, it will take you to

Desktop/classes/stats21/homework

`cd ..` will take you to the parent directory. If you are currently in `homework`, `cd ..` will take you to

Desktop/classes/stats21

`cd ../../..` will take you two levels up.

- Shell support tab completion. If you have the folder `homework` inside `stats21`, you can begin by typing `cd ho` and then hit TAB. Shell will try to complete what you are typing. If there are multiple items that start with `ho` then you can hit TAB multiple times until it finds the item you are looking for.

## Shell Basics (Mac OS)

- `ls` will list the contents of your current directory.
- `mkdir name` will create a new directory called `name` inside your current working directory.
- `cat filename` will print the contents of a file to your screen typically
- `clear` will clear the screen.
- `touch filename` will create a new file called `filename` in your current working directory
- `cp name1 name2` will copy a file or directory to a new location
- `mv name1 name2` will move or can rename a file or directory
- `rm name` will delete a file or directory

## Shell Basics (Windows )

- `dir` will list the contents of your current directory.
- `mkdir name` will create a new directory called `name` inside your current working directory.
- `cls` will clear the screen.
- `copy name1 name2` will copy a file or directory to a new location
- `move name1 name2` will move or can rename a file or directory
- `del name` will delete a file or directory called `name`
- `type name` will display the contents of a file called `name`

## Python Interpreter

- We will just do this once and not again in Stats 21
- At the command prompt, type python, if your computer is set up correctly



vivian — Python — 80x24

```
[vivian@Vivians-MacBook-Pro-3 ~ % python
Python 3.10.5 (v3.10.5:f377153967, Jun  6 2022, 12:36:10) [Clang 13.0.0 (clang-1
300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> ]
```



vivian — Python — 80x24

```
[vivian@Vivians-MacBook-Pro-3 ~ % python
Python 3.10.5 (v3.10.5:f377153967, Jun  6 2022, 12:36:10) [Clang 13.0.0 (clang-1
300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> print("Hello, World")
Hello, World
>>> ]
```

```
vivian@Vivians-MacBook-Pro-3 ~ % python
Python 3.10.5 (v3.10.5:f377153967, Jun  6 2022, 12:36:10) [Clang 13.0.0 (clang-1
300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> print("Hello, World")
Hello, World
[>>> import math
[>>> result = math.factorial(5)
[>>> print(result)
120
>>> ]
```

## Takeaways

- Python is an "interpreted" language. We can access the Python interpreter directly BUT it's inconvenient for us mortals
- So Python programmers write scripts -- e.g., .py and .ipynb files and access the interpreter differently (we will see that today)
- Python functions are called/invoked like R functions - syntax is *functionname(argument)*
- Python variable assignment *result = math.factorial(5)* uses the = only
- math is a module, part of the Python Standard Library, we import modules/packages/libraries

## Teams Please Try this!

If your team has a command prompt available, there is some code (`factorial1.py`) that you can download (you can download the others too).

- `cat factorial1.py` - reveals the contents
- `python factorial1.py` - the Python script `factorial1.py` is sent directly to the interpreter instead of typing it interactively

I would simply like you to see first hand running a Python script at the prompt BUT your computer does need to be set up correctly to do this.

```
Lecture2 -- -zsh -- 80x24
[vivian@Vivians-MacBook-Pro-3 Lecture2 % cat factorial1.py
# This is a comment
# Import the math module and all of its functions
# https://docs.python.org/3/tutorial/stdlib.html?highlight=library#mathematics
import math

# Calculate the factorial of 5 using the factorial function from the math module
# https://docs.python.org/3/library/math.html#module-math
# functions use ( ) so factorial(n)
# equal sign in this context means "assign" so we assign the result to result
result = math.factorial(5)

# Print the result
print(result)
[vivian@Vivians-MacBook-Pro-3 Lecture2 % python factorial1.py
120
vivian@Vivians-MacBook-Pro-3 Lecture2 % ]
```

## Modules, Packages and Libraries

- The module is a Python file/script/program (.py) it has collections of functions and variables. We will be importing modules very soon. (<https://docs.python.org/3/glossary.html#term-module>)
- The package is a collection of modules - best understood as a folder or directory. Later in the quarter, you will see the package pandas.  
([https://pandas.pydata.org/docs/getting\\_started/overview.html#](https://pandas.pydata.org/docs/getting_started/overview.html#))
- The library is a collection of packages with related functionality. The Python Standard Library (PSL - <https://docs.python.org/3/library/>) and Matplotlib (<https://matplotlib.org/stable/index.html>) are what they sound like, libraries.

## *pip (Pip Installs Packages or Pip Installs Python maybe preferred installer program)*

- I've had more success with pip than conda for installing and maintaining packages
- Check that you have it, at the terminal prompt you can run

```
python -m pip --version
```

- Installation using pip happens with

```
python -m pip install packagenames (note: Mac users might need to preface this with sudo)  
pip install packagenames (this works too)
```

- Updating can be done with

```
python -m pip install --upgrade packagenames
```

- Check what you have

```
python -m pip list
```

pythonProject2 > factorial1.py

Project Terminal: Local + main.py factorial1.py

Current File ▾

Notifications

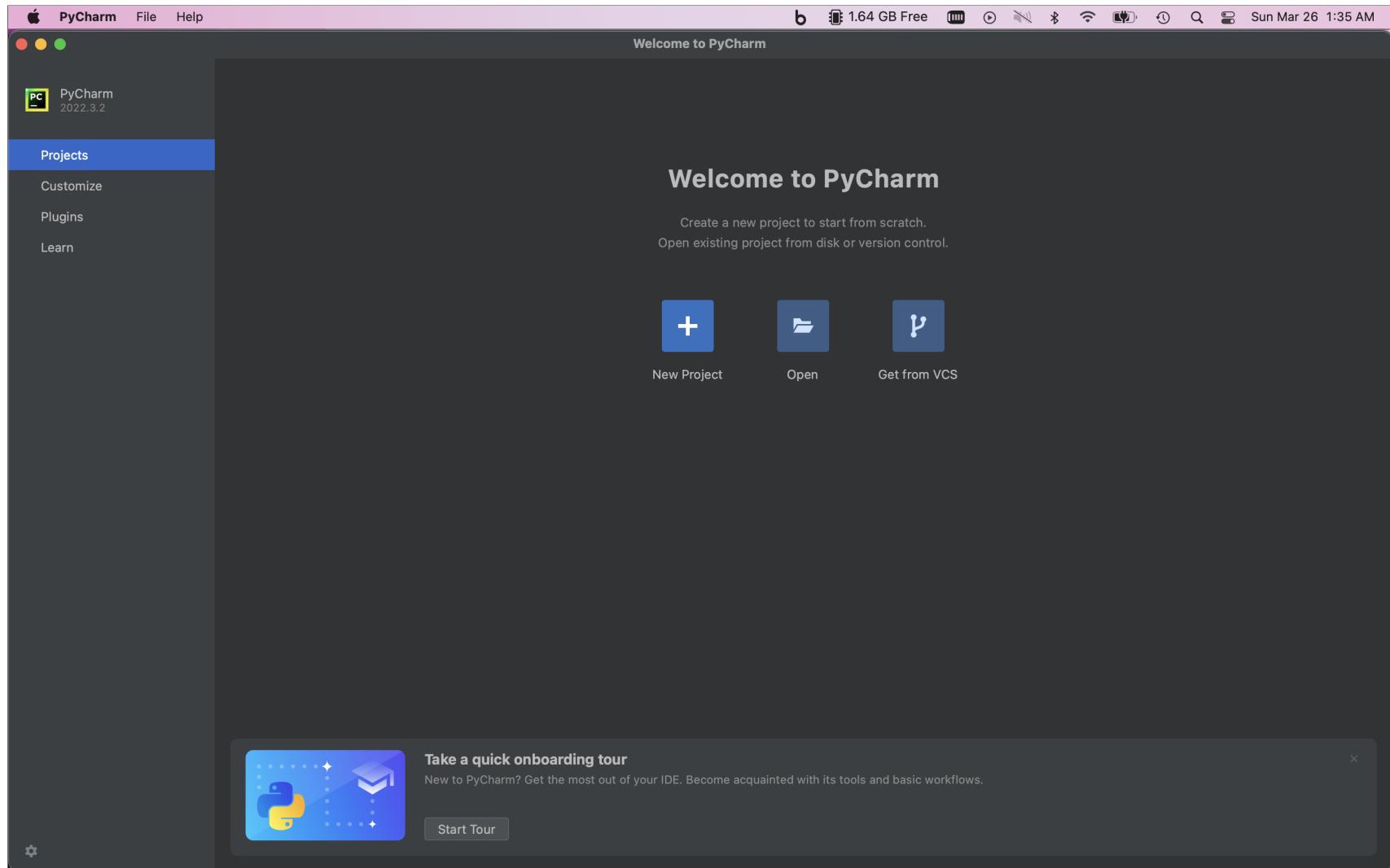
Terminal: Local

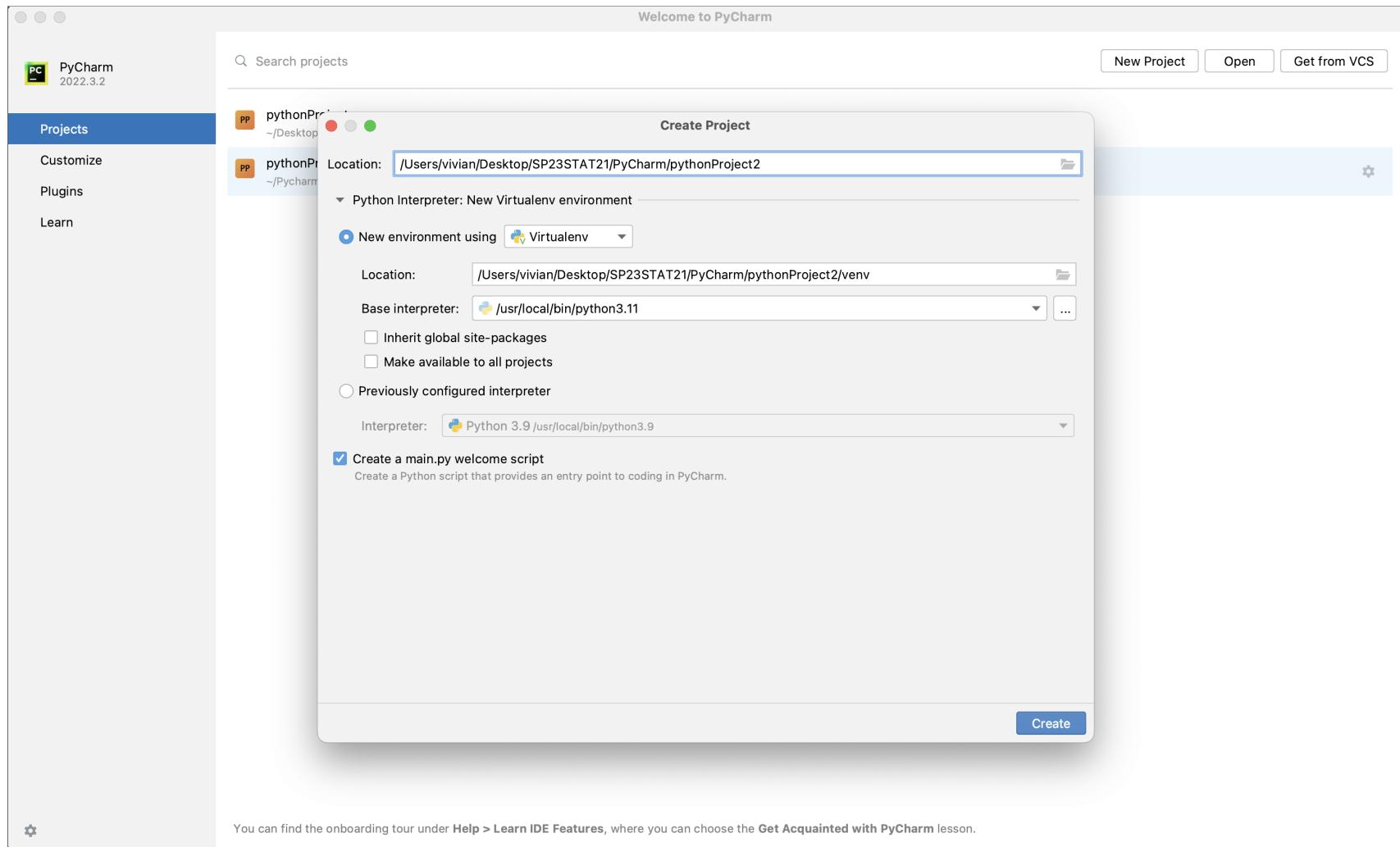
```
(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 % pip list
Package      Version
-----
pip          23.0.1
pytube       12.1.3
setuptools   67.5.1
wheel         0.38.4
(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 % pip install pandas
Collecting pandas
  Downloading pandas-2.0.0-cp311-cp311-macosx_10_9_x86_64.whl (11.6 MB)
    11.6/11.6 MB 19.0 MB/s eta 0:00:00
Collecting python-dateutil>=2.8.2
  Using cached python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
Collecting pytz>=2020.1
  Downloading pytz-2023.3-py2.py3-none-any.whl (502 kB)
    502.3/502.3 kB 7.4 MB/s eta 0:00:00
Collecting tzdata>=2022.1
  Downloading tzdata-2023.3-py2.py3-none-any.whl (341 kB)
    341.8/341.8 kB 7.1 MB/s eta 0:00:00
Collecting numpy>=1.21.0
  Using cached numpy-1.24.2-cp311-cp311-macosx_10_9_x86_64.whl (19.8 MB)
Collecting six>=1.5
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, tzdata, six, numpy, python-dateutil, pandas
```

## PyCharm

(<https://www.jetbrains.com/pycharm/>)

- is an IDE (integrated development environment) for Python development
- we can use the free community edition
- It has features and tools like code highlighting, code completion, and error checking.
- It automatically saves your work, has numerous features
- makes it easier to find/fix mistakes - step through code line by line, inspect variables, and set breakpoints (designated pause/stop)





pythonProject2 > main.py

pythonProject2 – main.py

Project

pythonProject2 ~/Desktop/SP23STAT21/PyCharm/p

main.py

External Libraries

Scratches and Consoles

main.py

```
# This is a sample Python script.

# Press ^R to execute it or replace it with your code.

# Press Double ↑ to search everywhere for classes, files, tool windows, actions, and settings.

def print_hi(name):
    # Use a breakpoint in the code line below to debug your script.

    print(f'Hi, {name}') # Press ⌘F8 to toggle the breakpoint.

# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    print_hi('PyCharm')

# See PyCharm help at https://www.jetbrains.com/help/pycharm/
```

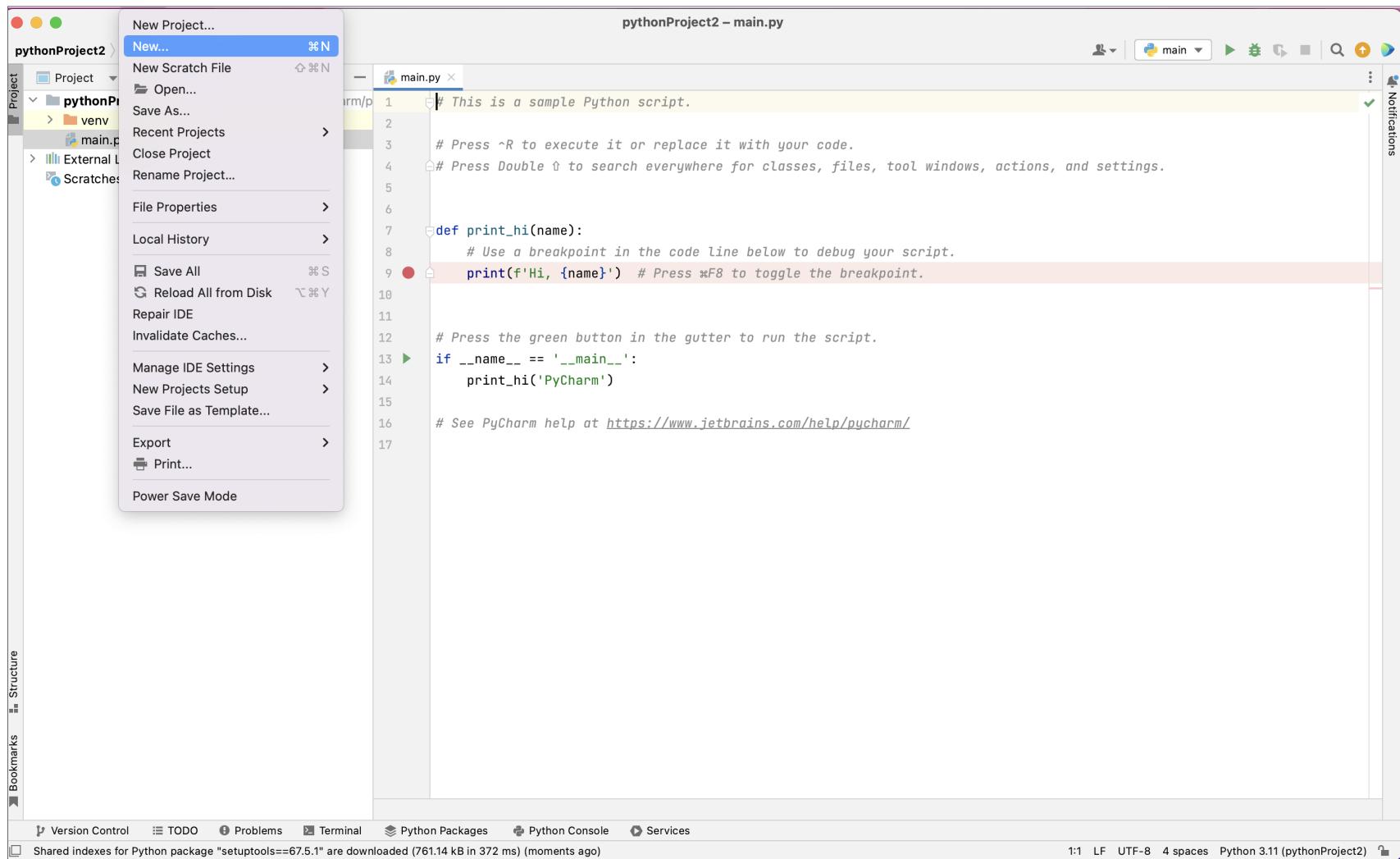
Structure

Bookmarks

Version Control    TODO    Problems    Terminal    Python Packages    Python Console    Services

Shared indexes for Python package "setuptools==67.5.1" are downloaded (761.14 kB in 372 ms) (moments ago)

1:1 LF UTF-8 4 spaces Python 3.11 (pythonProject2)



PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

76 MB Free Tue Apr 4 11:23 PM

pythonProject2 – main.py

pythonProject2 / main.py

Project

pythonProject2 ~/Desktop/SP23STAT21/PyCharm/p  
venv  
main.py

External Libraries  
Scratches and Consoles

main.py

# This is a sample Python script.

New

File  
New Scratch File ⌘N  
Directory  
Python Package

Python File  
HTML File  
Resource Bundle  
EditorConfig File

replace it with your code.  
everywhere for classes, files, tool windows, actions, and settings.

the code line below to debug your script.  
Press ⌘F8 to toggle the breakpoint.

# Press the green button in the gutter to run the script.

if \_\_name\_\_ == '\_\_main\_\_':  
 print\_hi('PyCharm')

# See PyCharm help at <https://www.jetbrains.com/help/pycharm/>

Structure

Bookmarks

Version Control TODO Problems Terminal Python Packages Python Console Services

Creates a Python file from the specified template

1:1 LF UTF-8 4 spaces Python 3.11 (pythonProject2)

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

pythonProject2 – main.py

pythonProject2 > main.py

Project v pythonProject2 ~/Desktop/SP23STAT21/PyCharm/p > venv main.py External Libraries Scratches and Consoles

main.py x

```
1 # This is a sample Python script.
2
3 # Press ^R to execute it or replace it with your code.
4 # Press Double ↑ to search everywhere for classes, files, tool windows, actions, and settings.
5
6
7 def print_hi(name):
8     # Use a breakpoint in the code line below to debug your script.
9     print(f'Hi, {name}') # Press *F8 to toggle the breakpoint.
```

10
11
12 # Press the green button in the gutter to run the script.
13 if \_\_name\_\_ == '\_\_main\_\_':
14 print\_hi('PyCharm')
15
16 # See PyCharm | factorial| PyCharm/
17

New Python file

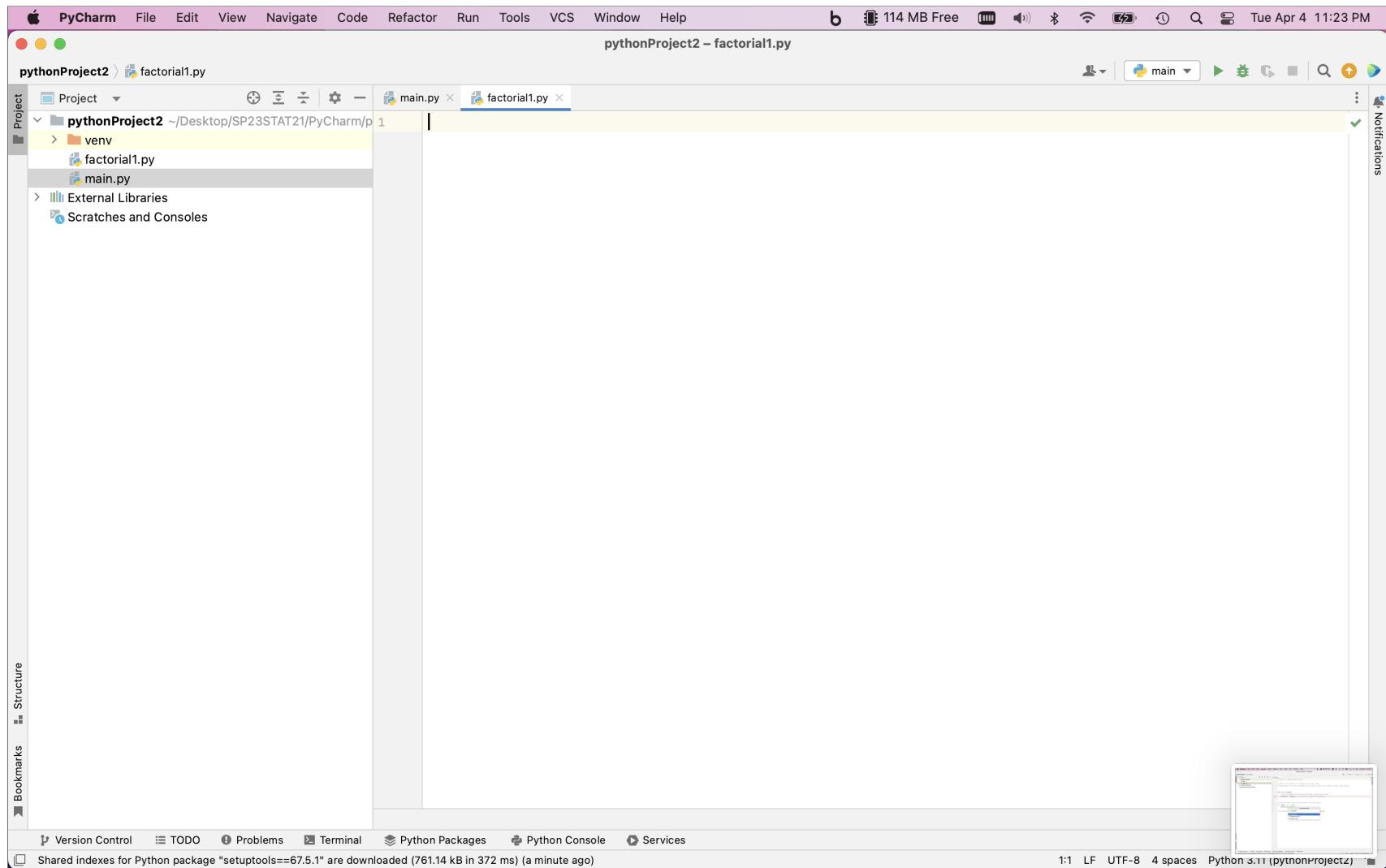
- factorial|
- Python file
- Python unit test
- Python stub

Structure Bookmarks

Version Control TODO Problems Terminal Python Packages Python Console Services

Shared indexes for Python package "setuptools==67.5.1" are downloaded (761.14 kB in 372 ms) (a minute ago)

1:1 LF UTF-8 4 spaces Python 3.11 (pythonProject2)



If your team has PyCharm, please try it

- If no PyCharm, you can use whatever you want (R Studio will work)

Just type 3 lines (or copy the online code to your project folder):

```
import math  
  
result = math.factorial(5)  
  
print(result)
```

You can add your own comments (those lines start with #)

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

212 MB Free Tue Apr 4 11:24 PM

pythonProject2 – factorial1.py

pythonProject2 > factorial1.py

Project main.py factorial1.py

```
1 import math
2
3 result = math.factorial(7)
4
5 print(result)
```

Notifications

Scratches and Consoles

External Libraries

Structure

Bookmarks

Version Control TODO Problems Terminal Python Packages Python Console Services

PEP 8: W292 no newline at end of file 5:14 LF UTF-8 4 spaces Python 3.11 (pythonProject2)

The screenshot shows the PyCharm IDE interface. The project is named 'pythonProject2' and contains files 'factorial1.py' and 'main.py'. The 'main.py' file is open and displays the following code:

```
1 import math
2
3 result = math.factorial(7)
4
5 print(result)
```

The code imports the 'math' module, calculates the factorial of 7, and prints the result. A warning icon is visible in the gutter next to line 1. The bottom status bar indicates the file is 5:14, uses LF line endings, is in UTF-8 encoding, has 4 spaces, and is using Python 3.11 (pythonProject2).

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

197 MB Free Tue Apr 4 11:24 PM

pythonProject2 – factorial1.py

pythonProject2 > factorial1.py

Project main.py factorial1.py

```
1 import math
2
3 result = math.factorial(7)
4
5 print(result)
```

Edit Configurations...  
Current File  
Run Configurations  
main

Notifications

Scratches and Consoles

External Libraries

Structure

Bookmarks

Version Control TODO Problems Terminal Python Packages Python Console Services

Run the file that is opened in the focused editor

5:14 LF UTF-8 4 spaces Python 3.11 (pythonProject2)

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

144 MB Free Tue Apr 4 11:24 PM

pythonProject2 – factorial1.py

pythonProject2 > factorial1.py

Project

pythonProject2 ~/Desktop/SP23STAT21/PyCharm/p  
> venv  
factorial1.py  
main.py

External Libraries  
Scratches and Consoles

main.py factorial1.py

```
import math
result = math.factorial(7)
print(result)
```

Run 'factorial1.py' ^R

Notifications

Structure

Bookmarks

Version Control TODO Problems Terminal Python Packages Python Console Services

Run selected configuration

5:14 LF UTF-8 4 spaces Python 3.11 (pythonProject2)

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

126 MB Free Tue Apr 4 11:25 PM

pythonProject2 – factorial1.py

pythonProject2 > factorial1.py

Project main.py factorial1.py

```
1 import math
2
3 result = math.factorial(7)
4
5 print(result)
```

Notifications

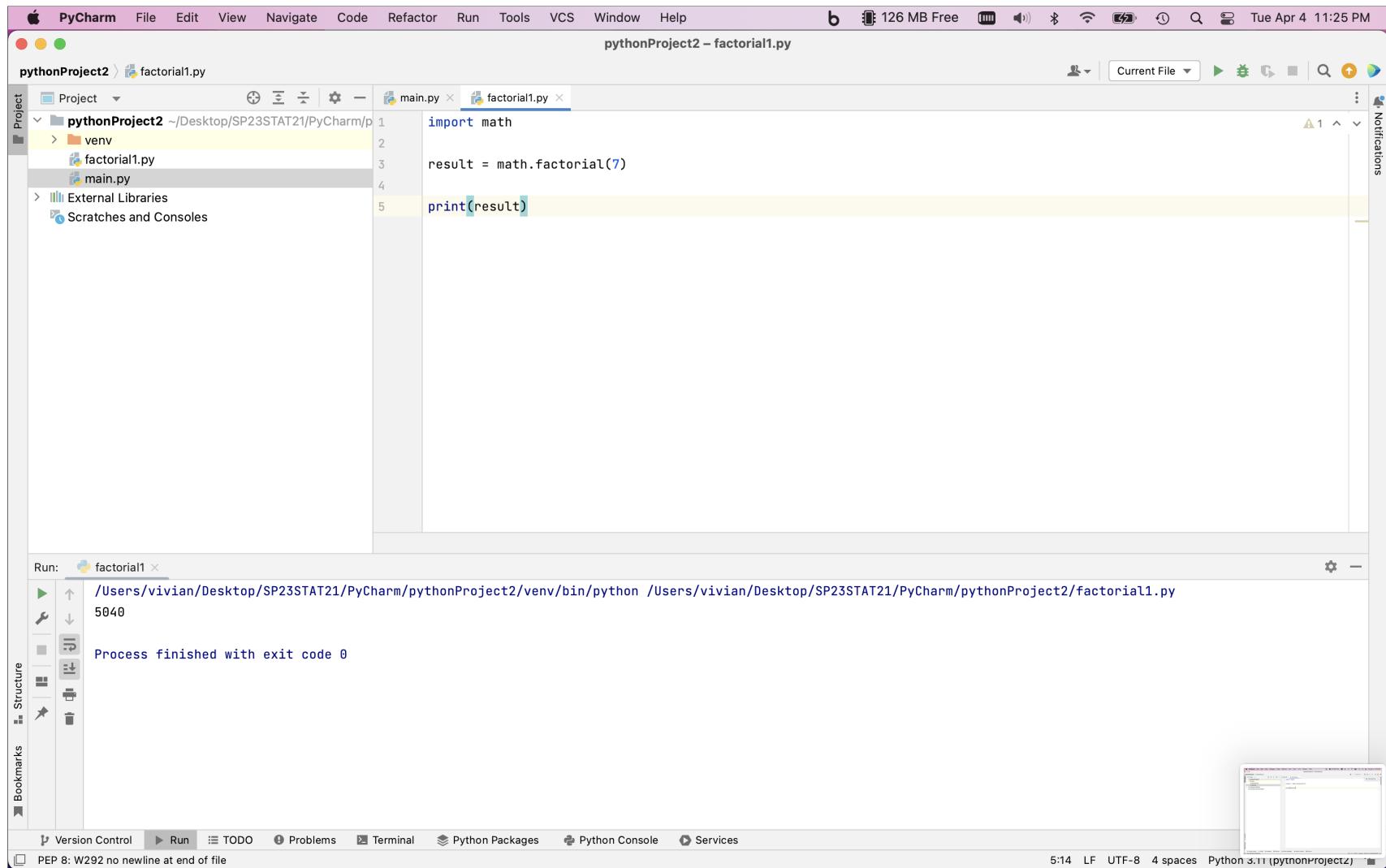
Run: factorial1

```
/Users/vivian/Desktop/SP23STAT21/PyCharm/pythonProject2/venv/bin/python /Users/vivian/Desktop/SP23STAT21/PyCharm/pythonProject2/factorial1.py
5040
Process finished with exit code 0
```

Structure Bookmarks

Version Control Run TODO Problems Terminal Python Packages Python Console Services

PEP 8: W292 no newline at end of file 5:14 LF UTF-8 4 spaces Python 3.11 (pythonProject2)



PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

pythonProject2 – factorial1.py

pythonProject2 > factorial1.py

Project

pythonProject2 ~/Desktop/SP23STAT21/PyCharm/p  
> venv  
factorial1.py  
main.py

External Libraries  
Scratches and Consoles

main.py factorial1.py

1 import math  
2  
3 result = math.factorial(7)  
4  
5 print(result)

Notifications

Terminal: Local +

(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 % python -V  
Python 3.11.2  
(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 %

Structure

Bookmarks

Version Control Run TODO Problems Terminal Python Packages Python Console Services

PEP 8: W292 no newline at end of file

5:14 LF UTF-8 4 spaces Python 3.11 (pythonProject2)

The screenshot shows the PyCharm IDE interface. The top menu bar includes PyCharm, File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The title bar indicates the project is "pythonProject2" and the current file is "factorial1.py". The left sidebar has a "Project" view showing the directory structure: "pythonProject2" containing "venv", "factorial1.py", and "main.py". Below this are sections for "External Libraries" and "Scratches and Consoles". The main editor area displays the code for "main.py": "import math", "result = math.factorial(7)", and "print(result)". The terminal below shows a Python 3.11.2 environment with the command "python -V" run. The bottom navigation bar includes Version Control, Run, TODO, Problems, Terminal, Python Packages, Python Console, and Services. A status bar at the bottom shows PEP 8 compliance, file encoding (LF), character width (4 spaces), Python version (3.11), and the current time (5:14).

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

pythonProject2 – factorial1.py

pythonProject2 / factorial1.py

Project main.py factorial1.py

```
1 import math
2
3 result = math.factorial(7)
4
5 print(result)
```

Notifications

Terminal: Local

```
(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 % python -V
Python 3.11.2
(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 % pip list
Package Version
-----
pip    23.0.1
setuptools 67.5.1
wheel   0.38.4
(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 %
```

Structure Bookmarks

Version Control Run TODO Problems Terminal Python Packages Python Console Services

PEP 8: W292 no newline at end of file

294 MB Free Tue Apr 4 11:25 PM

PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help

pythonProject2 – factorial1.py

pythonProject2 > factorial1.py

Project

pythonProject2 ~/Desktop/SP23STAT21/PyCharm/p  
> venv  
factorial1.py  
main.py

External Libraries  
Scratches and Consoles

main.py factorial1.py

1 import math  
2  
3 result = math.factorial(7)  
4  
5 print(result)

Notifications

Terminal: Local

Package	Version
pip	23.0.1
setuptools	67.5.1
wheel	0.38.4

(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 % pip install pytube

Collecting pytube

Using cached pytube-12.1.3-py3-none-any.whl (57 kB)

Installing collected packages: pytube

Successfully installed pytube-12.1.3

(venv) vivian@Vivians-MacBook-Pro-3 pythonProject2 %

Structure Bookmarks

Version Control Find Run TODO Problems Terminal Python Packages Python Console Services

PEP 8: W292 no newline at end of file

383 MB Free Tue Apr 4 11:26 PM

5:14 LF UTF-8 4 spaces Python 3.11 (pythonProject2)

## Recap

- PyCharm is just one of many IDEs. Using an IDE is optional for Stats 21
- It has more features than we can use, it is well suited for developers
- It utilizes a virtual environment so it's like a new Python install with each new project
- Great for writing/testing code

## Changing the subject - the Notebook

- A different development environment, very interactive
- Code (Python) and text (markdown) can be mixed together
- A notebook is almost (almost!) designed to be executed in a non-linear manner

## JupyterLab

<https://jupyter.org/install>

- First install and then run from the terminal prompt.

```
pip install jupyterlab
```

```
jupyter lab
```

- A browser tab or new window should open
- Your lecture slides are written using Jupyter and then converted out to html for presentation
- And printed to PDF for distribution

localhost:8888/lab

File Edit View Run Kernel Tabs Settings Help

+ Week3\_4\_Notebook\_firstname.ipynb Week01\_01\_ChatGPT.ipynb Week\_1.ipynb

Filter files by name / Week01 / Lecture 1/

Name	Last Modified
anaconda.png	a year ago
AnacondaWor...	a year ago
ChatGPTansw...	6 hours ago
Fall_Example...	a year ago
hw1example....	a year ago
miles2.mp4	a year ago
MRS_LEW.JPG	a year ago
Screen Recor...	a year ago
Screen Shot 2...	a year ago
Screen Shot 2...	a year ago
Screen Shot 2...	a year ago
Screen Shot 2...	a year ago
Screen Shot 2...	a year ago
terminal1.png	a year ago
terminal2.png	40 minutes ago
terminal3.png	40 minutes ago
terminal4.png	29 minutes ago
terminal5.png	28 minutes ago
Week_1-1 slid...	a year ago
Week_1.ipynb	2 minutes ago
Week_1-1.slid...	2 minutes ago
Window1.JPG	a year ago
window1.png	a year ago
winInstall.png	a year ago

## JupyterLab (<https://jupyter.org/install>)

- I will discuss this further in the coming days. But if you just want to start playing with Jupyter Lab from the terminal prompt.

```
pip install jupyterlab
```

```
jupyter lab
```

## Google Colab (<https://colab.research.google.com/>)

- Google Colab (short for "collaboratory") is a cloud-based development environment that provides a Jupyter Notebook interface.
- Colab allows you to run and execute Jupyter Notebooks on Google's cloud servers,
- Benefit allows you to perform high-performance computations for machine learning and other data-intensive tasks.

## See you next time

```
[5]: from IPython.display import IFrame
IFrame(src="https://www.youtube.com/embed/hlqdW24ZGS0",
       width="560", height="315")
```

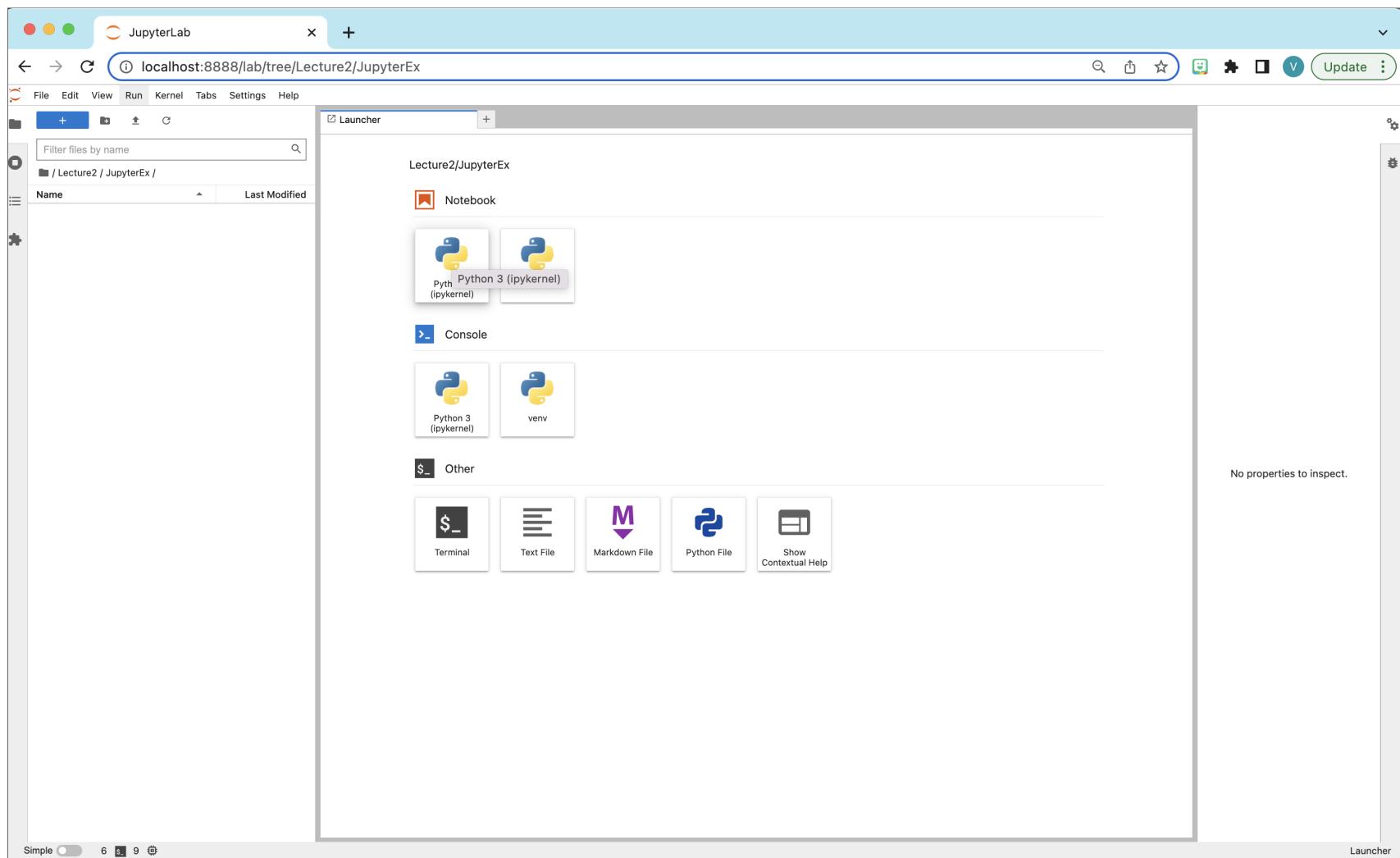
D Sleep Token - The Summoning Ending

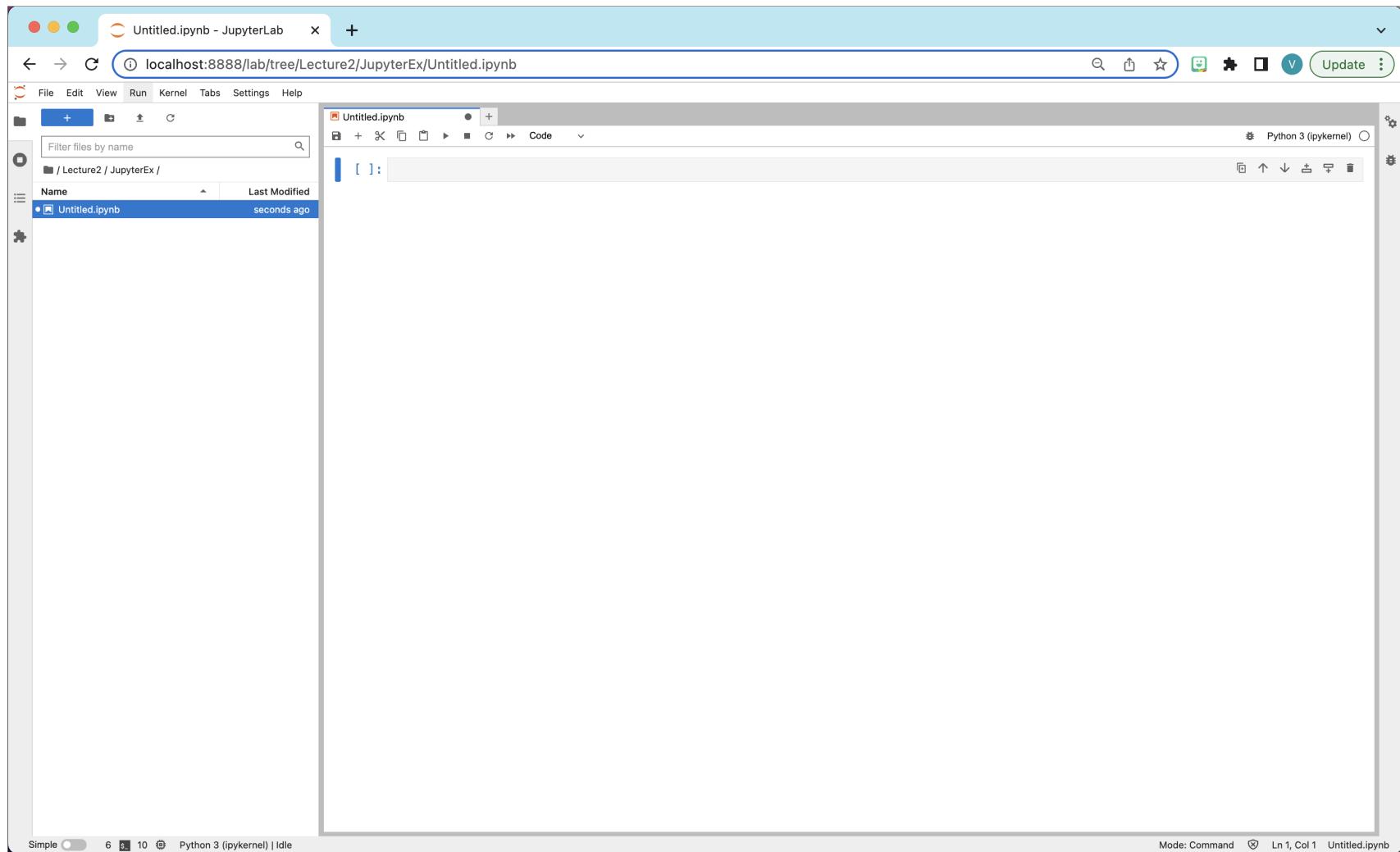


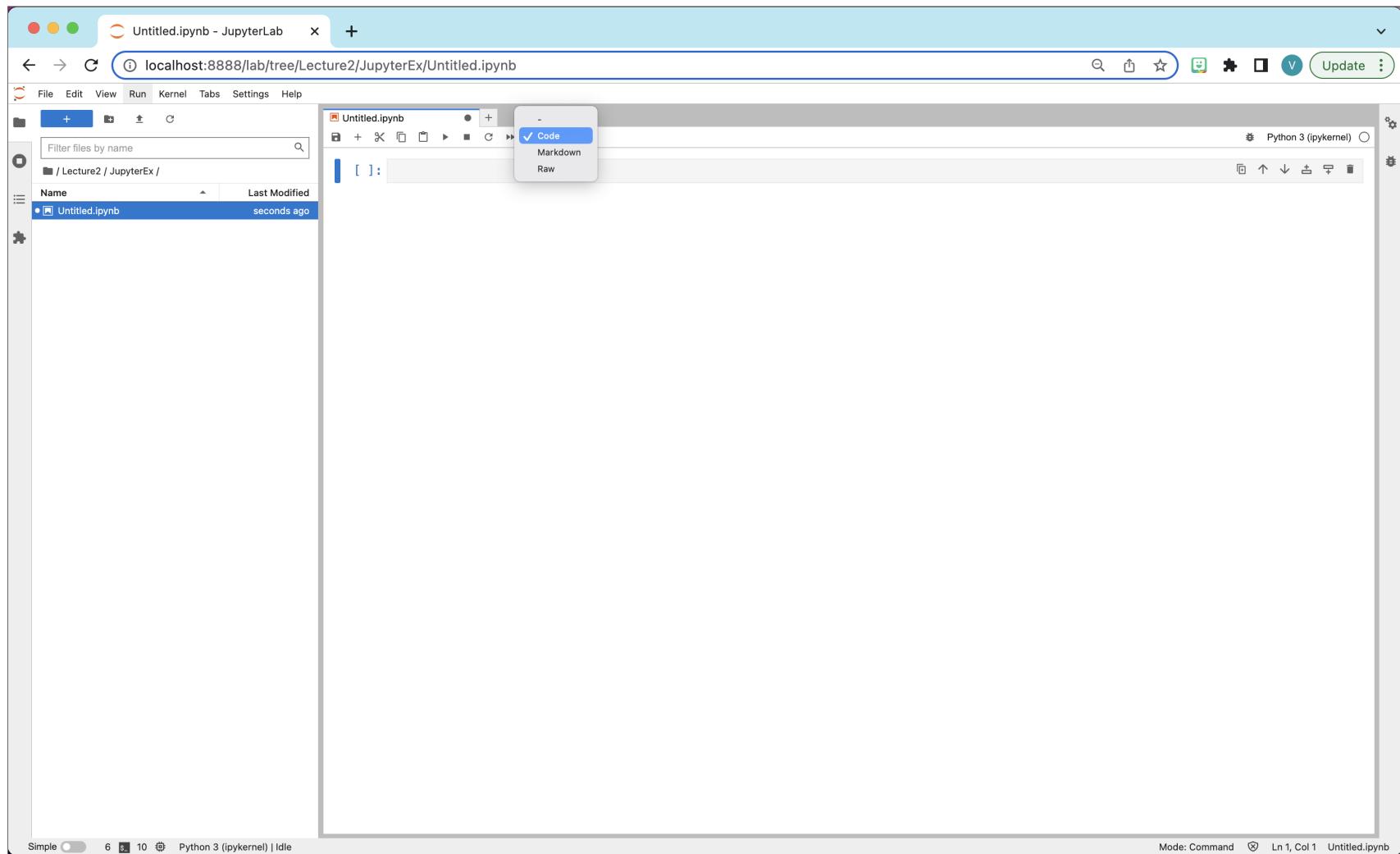
```
<script>
setBackgroundImage("Window1.jpg", "black");
```

Watch on YouTube Watch later Share

Simple 0 s 3 Python 3 (ipykernel) | Idle Mode: Command Ln 2, Col 72 Week\_1.ipynb







The screenshot shows the JupyterLab interface. The top bar displays the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The menu bar includes File, Edit, View, Run, Kernel, Tabs, Settings, and Help. The left sidebar features a file browser with a search bar and a list of files under "/Lecture2 / JupyterEx /", showing "Untitled.ipynb" as the selected file. The main workspace contains a code editor tab titled "Untitled.ipynb" with the following Python code:

```
[ ]: import math  
result = math.factorial(5)  
print(result)
```

The status bar at the bottom indicates "Simple" mode, "Python 3 (ipykernel) | Idle", "Mode: Edit", "Ln 5, Col 14", and "Untitled.ipynb".

A screenshot of the JupyterLab interface. The top bar shows the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The toolbar includes standard browser controls (back, forward, search) and JupyterLab specific icons (refresh, update, etc.). A context menu is open over a code cell, listing options like "Run Selected Cells", "Run All Above Selected Cell", and "Restart Kernel and Run All Cells...". The main workspace contains a code cell with the following Python code:

```
import math
result = math.factorial(5)
print(result)
```

The status bar at the bottom shows "Simple" mode, line numbers 6, 8, 10, and "Python 3 (ipykernel) | Idle". It also indicates "Mode: Command" and the current position "Ln 5, Col 14 Untitled.ipynb".

The screenshot shows the JupyterLab interface. The top navigation bar displays the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The left sidebar contains a file tree with a single item: "Untitled.ipynb" under the path "/Lecture2 / JupyterEx /". The main workspace is a code editor with a tab titled "Untitled.ipynb". A code cell at index [1] contains the following Python code:

```
import math
result = math.factorial(5)
print(result)
120
```

The code cell has just been run, and its output, "120", is displayed below it. The status bar at the bottom indicates "Mode: Edit" and "Ln 1, Col 1 Untitled.ipynb".

A screenshot of the JupyterLab interface. The top bar shows the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The left sidebar has a file tree and a search bar. A context menu is open over a cell, with "Move Cells Down" highlighted. The main area contains a code cell [1]:

```
[1]: import math  
result = math.factorial(5)  
print(result)  
120
```

The status bar at the bottom shows "Simple" mode, 6 cells, Python 3 (ipykernel) | Idle, Mode: Command, Ln 5, Col 14, Untitled.ipynb.

The screenshot shows the JupyterLab interface with a single tab titled "Untitled.ipynb - JupyterLab". The address bar indicates the URL is "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The left sidebar displays a file tree under "/Lecture2 / JupyterEx / Untitled.ipynb". The main workspace contains a code cell with the following content:

```
[1]: import math  
      result = math.factorial(5)  
      print(result)  
      120
```

The cell is currently set to "Code" mode, indicated by the blue button in the toolbar above the cell area. The status bar at the bottom shows "Simple" mode, "Python 3 (ipykernel) | Idle", and "Mode: Command".

The screenshot shows the JupyterLab interface. The top navigation bar displays the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The left sidebar contains a file tree with a single item: "Untitled.ipynb" under the path "/Lecture2 / JupyterEx /". The main workspace is a code editor with a Python 3 kernel. A code cell is visible, containing the following Python code:

```
[1]: import math  
      result = math.factorial(5)  
      print(result)  
      120
```

The status bar at the bottom indicates "Mode: Command" and "Ln 1, Col 1 Untitled.ipynb".

The screenshot shows the JupyterLab interface. The top bar displays the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The left sidebar shows a file tree with "Untitled.ipynb" selected. The main area contains a code cell with the following Python code:

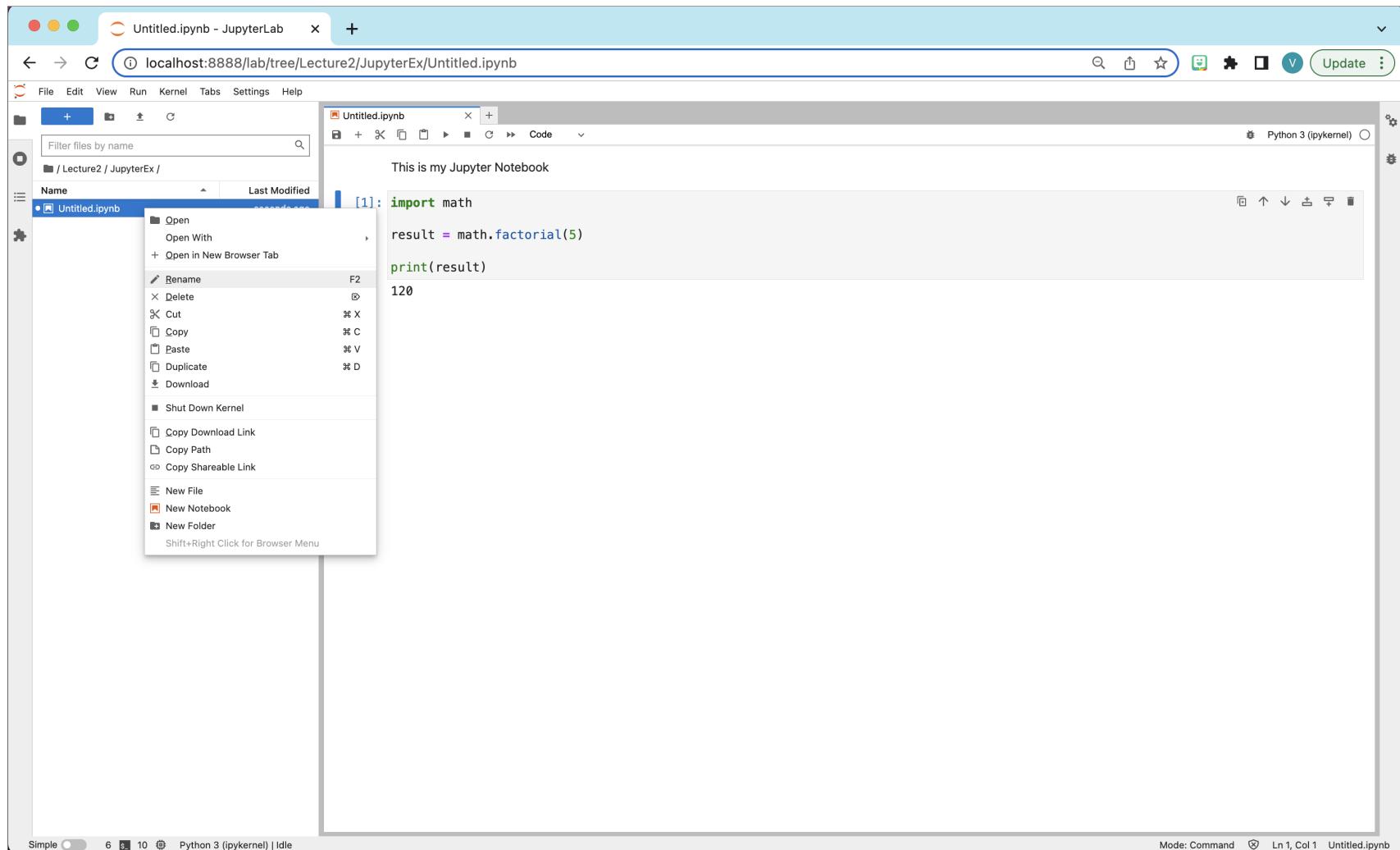
```
[1]: import math  
result = math.factorial(5)  
print(result)  
120
```

The status bar at the bottom indicates "Mode: Command" and "Ln 1, Col 28 Untitled.ipynb".

The screenshot shows the JupyterLab interface. The top bar displays the title "Untitled.ipynb - JupyterLab" and the URL "localhost:8888/lab/tree/Lecture2/JupyterEx/Untitled.ipynb". The left sidebar shows a file tree with a single file named "Untitled.ipynb" under the path "/Lecture2 / JupyterEx /". The main area contains a Jupyter Notebook cell titled "Untitled.ipynb". The cell content is as follows:

```
This is my Jupyter Notebook
[1]: import math
      result = math.factorial(5)
      print(result)
      120
```

The status bar at the bottom indicates "Simple" mode, "Python 3 (ipykernel) | Idle", "Mode: Command", "Ln 1, Col 1", and "Untitled.ipynb".

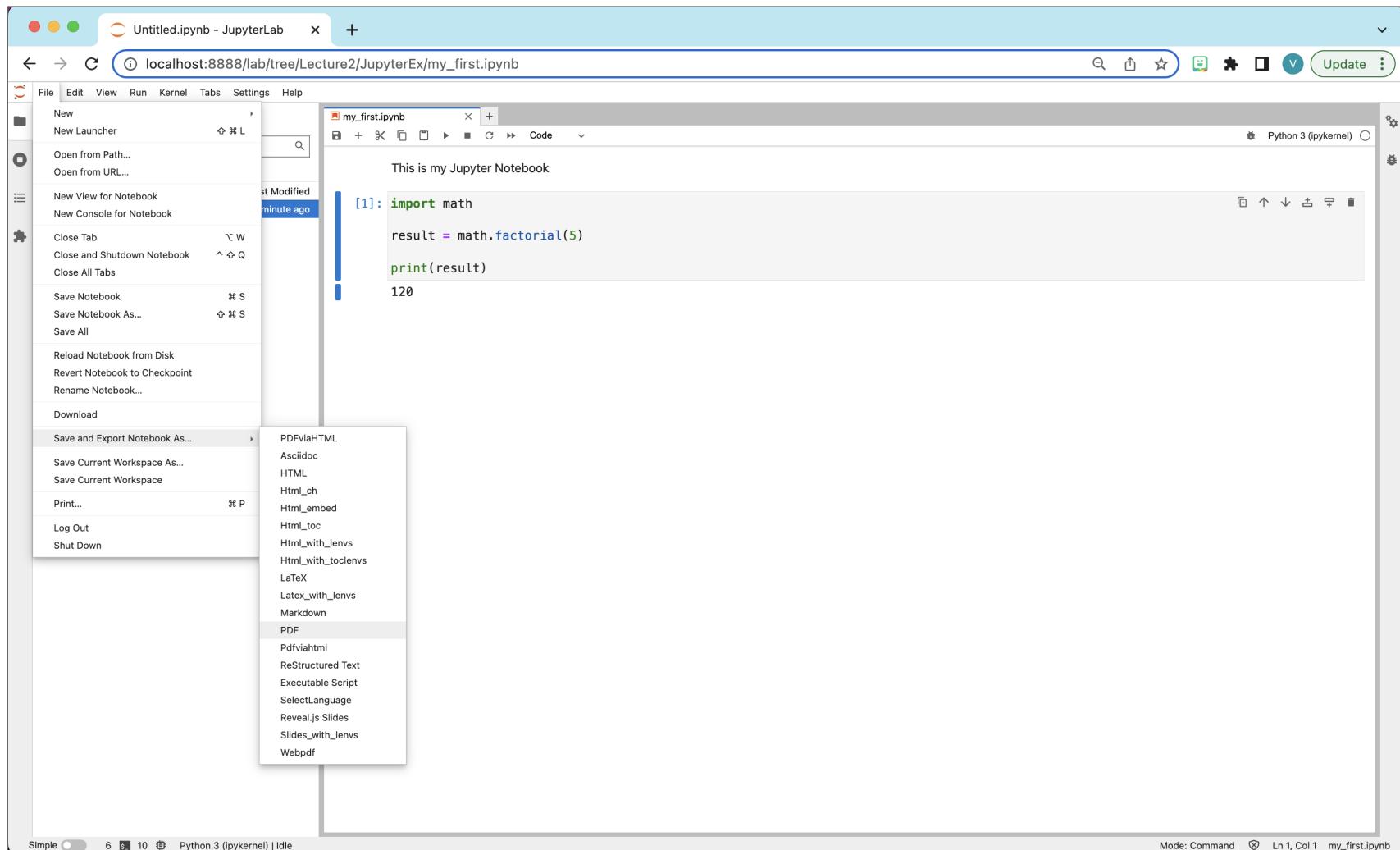


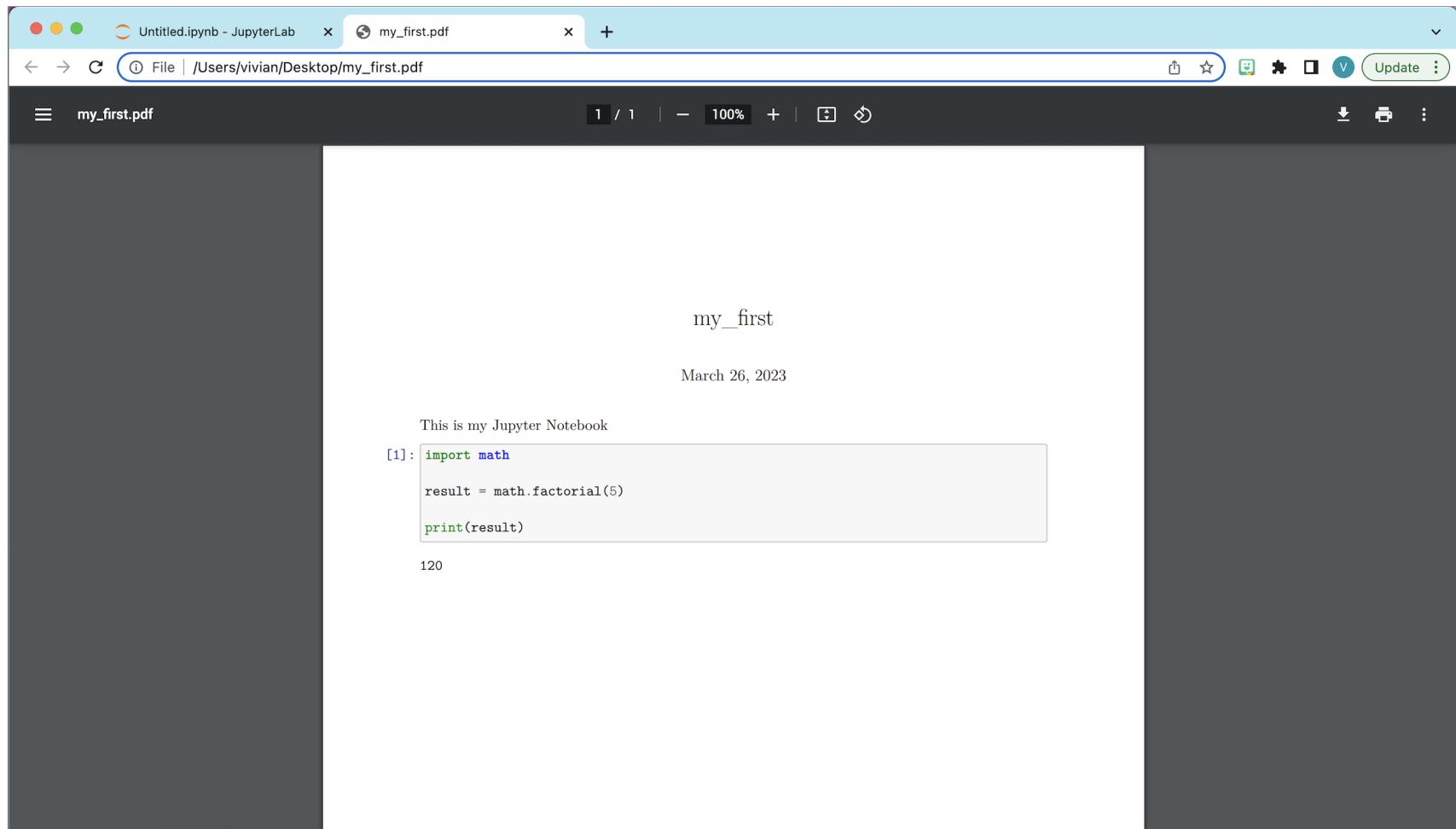
The screenshot shows the JupyterLab interface. At the top, there's a header bar with a tab labeled "Untitled.ipynb - JupyterLab" and a URL "localhost:8888/lab/tree/Lecture2/JupyterEx/my\_first.ipynb". Below the header is a navigation bar with links for File, Edit, View, Run, Kernel, Tabs, Settings, and Help. To the right of the navigation bar are several icons for search, refresh, and other functions.

The main area consists of two panes. On the left is a file tree pane titled "my\_first.ipynb" which shows a directory structure under "/Lecture2 / JupyterEx /" with a single file "my\_first.ipynb" listed. The right pane is a notebook editor titled "my\_first.ipynb" with a status bar indicating "Python 3 (ipykernel)". The notebook content is as follows:

```
This is my Jupyter Notebook
[1]: import math
      result = math.factorial(5)
      print(result)
      120
```

At the bottom of the interface, there are several small icons and a status bar with the text "Simple" and "Mode: Command".





## Jupyter Lab Takeaways

- It is great for creating reports or web pages etc. anything that mixes text and code (like this presentation)
- Jupyter Lab has fewer features than IDEs like PyCharm (e.g., it does not automatically save your files)
- It may not warn you about a mistake (but it will stop if there are errors)
- It can create .py files BUT we use it primarily to create .ipynb files (interactive Python notebook)
- Most of your homework will be in the form of a .ipynb

## Google Colab

(<https://colab.research.google.com/>)

- Google Colab (short for "collaboratory") is a cloud-based development environment that provides a Jupyter Notebook interface.
- Colab allows you to run and execute Jupyter Notebooks on Google's cloud servers,
- One Benefit allows you to perform high-performance computations for machine learning and other data-intensive tasks.
- You can use either, but you will need to figure out your own solutions if you choose Colab

← → ⌛ colab.research.google.com

File Edit View Insert Runtime Tools Help

Share ⚙️ V Update

Table of contents + Code + Text Copy to Drive Connect ^

Getting started  
Data science  
{x} Machine learning  
More Resources  
Featured examples  
Section

## Welcome to Colab!

If you're already familiar with Colab, check out this video to learn about interactive tables, the executed code history view, and the command palette.



### What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

- Zero configuration required
- Access to GPUs free of charge
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!

#### Getting started

See You Next Time