Introduction to Python

Mahdi Shafiee Kamalabad

How is your programming skill? ☺

- I do not know how to write a program at all.
- I know it just a little bit (very basic).
- I am an intermediate programmer (some experience).
- I am a professional programmer.

Utrecht University

Please go https://www.menti.com/ and answer,

Code: 2310 4691

https://www.mentimeter.com/s/2c3817841872ff05dd4d8d8c101e7094/60e73b8a9081/edit

Utrecht University

The content of this workshop is as follows

1. Python Overview

- Brief History
- Usage of Python

2. Installing Python

- Install Anaconda Distribution for Python
- Briefly running Jupyter Notebook
- Exploring “no install” online options

3. Running Python Code

- Text Editors and Full IDEs
- Notebook Environments
- Jupyter Notebook
- A Quick Tour in Jupyter Notebook

4. Python Coding

- Python Object
- Python Data Structure Basics
- Python Comparison Operators
- Python Statements
- Methods and Functions
- A Tour in NumPy Library
- A Tour in Pandas Library
- A Tour in Matplotlib Library

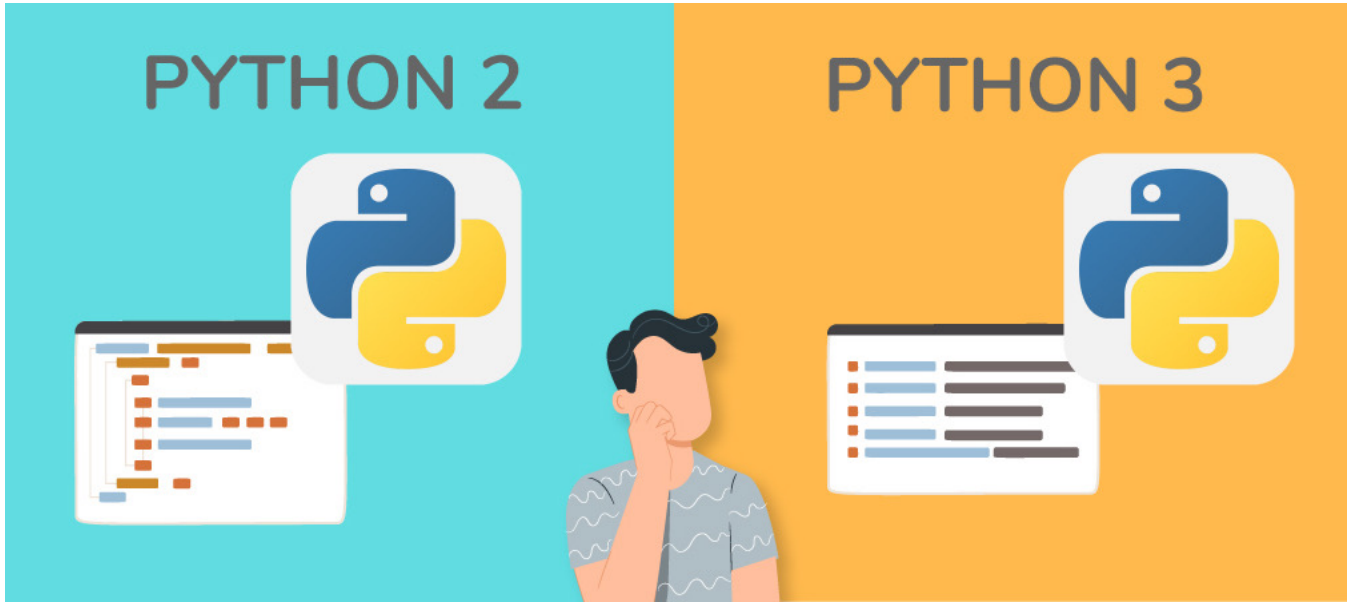
Utrecht University

This workshop in particular is geared towards people new to programming.

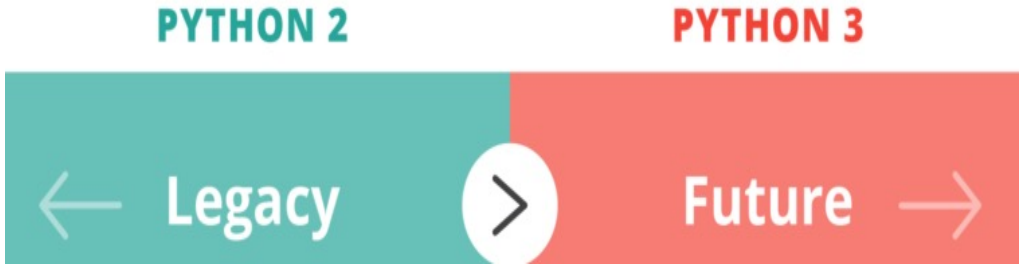
Utrecht University

First let's see which one:

Python 2 vs ***Python 3***



- Now every major external python package has been updated to support Python 3!
- This workshop now focuses on Python 3.
- If need be, going back to Python 2 syntax is a very easy jump once you know Python 3.
- Python 3 is indeed the future of Python.



Python Overview

Brief History of Python

- Created in 1990 by Guido van Rossum
- Python 3 released in 2008
- Specifically designed as an easy-to-use language

Why Choose Python?

- Designed for clear and logical code that is easy to read and learn.
- It has a simple syntax.
- Lots of existing libraries and frameworks written in Python allow users to apply Python to a wide variety of tasks.
- Can be easily written and executed much faster than other programming languages.
- Great documentation online:

  docs.python.org/3

We first focus on “**base**” Python,

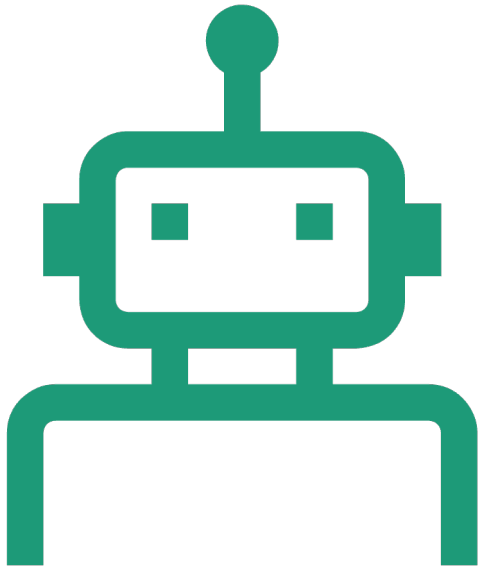
which consists of the core components of the language, and

writing scripts and small programs.

What can you do with Python?

Automate simple tasks

- Searching for files and editing them
- Scraping information from a website
- Reading and editing excel files
- Working with PDFs
- Writing emails and text messages
- Filling out forms



What can you do with Python?

Data Science and Machine Learning

- Analyze large data files
- Create visualizations
- Perform machine learning tasks
- Create and run predictive algorithms

What can you do with Python?

Create websites

- Use web frameworks such as Django and Flask to handle the backend of a website and user data
- Create interactive dashboards for users

Once you understand the base Python and begin working with a few libraries, you'll quickly start to see Python's vast potential for your own projects!

Utrecht University

Strongly recommended to

quick ***Google*** or ***StackOverflow*** search!

Installing Python

- There are many ways to run Python!
- Later, we will explore the difference between running a Python *.py* script or running Python code in a *Notebook* environment.

Either way, we will still want to install Python!

Installation

- Install ***Anaconda*** Distribution for Python.

  Anaconda installs Python. It is an easy-to-use development environment and navigator launch tool.

- Briefly run ***Jupyter Notebook***.

- Explore "*no install*" online options.

Quick Note:

- Many online “no install” Python environments that can run in the browser (given that you have internet).
- We will give you a brief tour of these online “no install” options at the end.

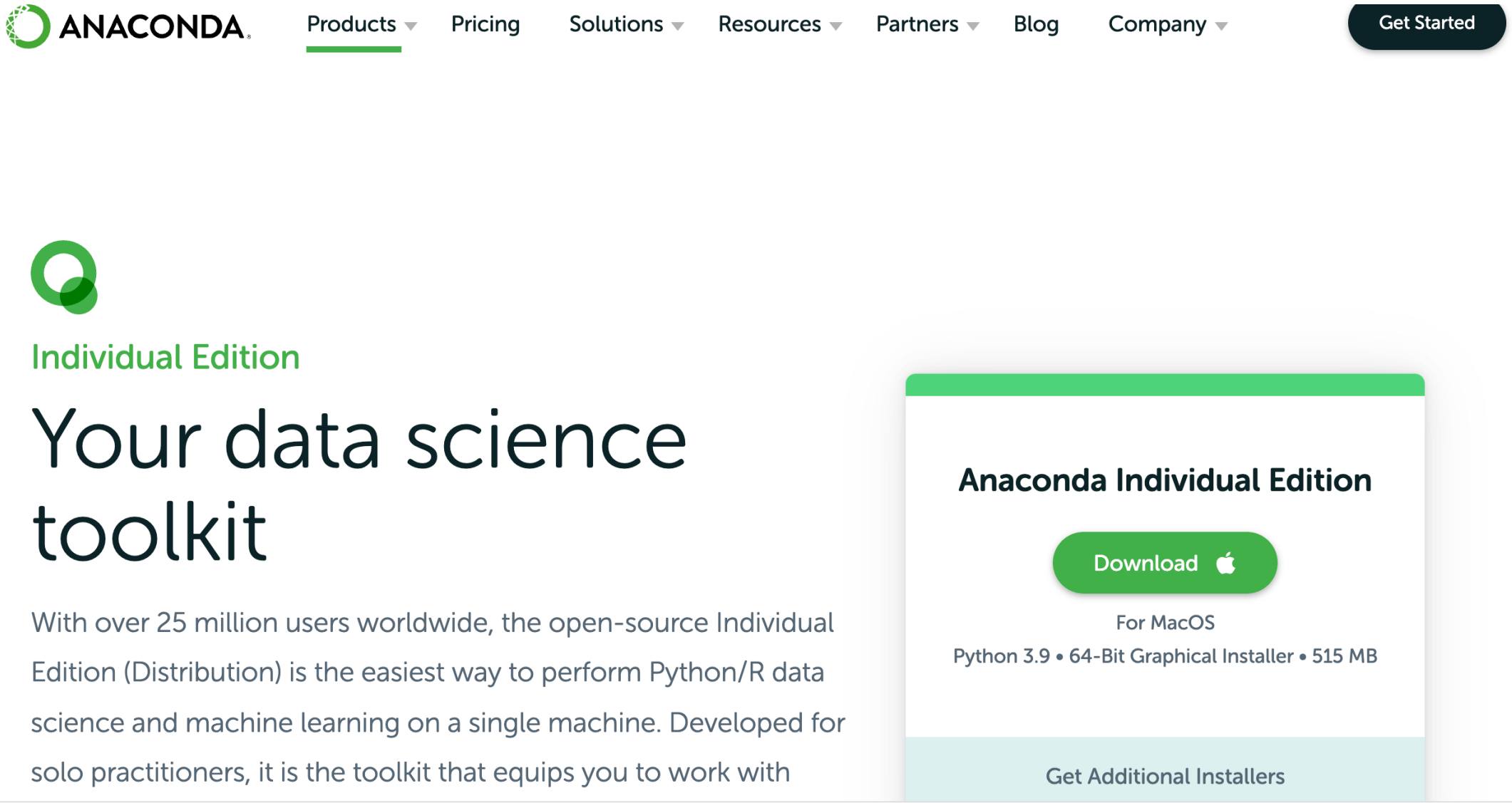
Utrecht University

Install Anaconda

- To install Python, we will use the free Individual Anaconda distribution.
- This distribution includes Python as well as many other useful libraries, including the Jupyter Notebook environment.
- Anaconda can also easily be installed on any major OS, Windows, MacOS, or Linux.

www.anaconda.com/downloads

Utrecht University



Select the installer as per your operating system.

Utrecht University

Anaconda Installers

Windows

Python 3.9

64-Bit Graphical Installer (510 MB)

32-Bit Graphical Installer (404 MB)

MacOS

Python 3.9

64-Bit Graphical Installer (515 MB)

64-Bit Command Line Installer (508 MB)

Linux

Python 3.9

64-Bit (x86) Installer (581 MB)

64-Bit (Power8 and Power9) Installer (255 MB)

64-Bit (AWS Graviton2 / ARM64) Installer (488 M)

64-bit (Linux on IBM Z & LinuxONE) Installer (242 M)

In case, you have trouble installing

- Tutorial video for installing anaconda on *Windows*

https://www.youtube.com/watch?v=aN6OVm0mTHo

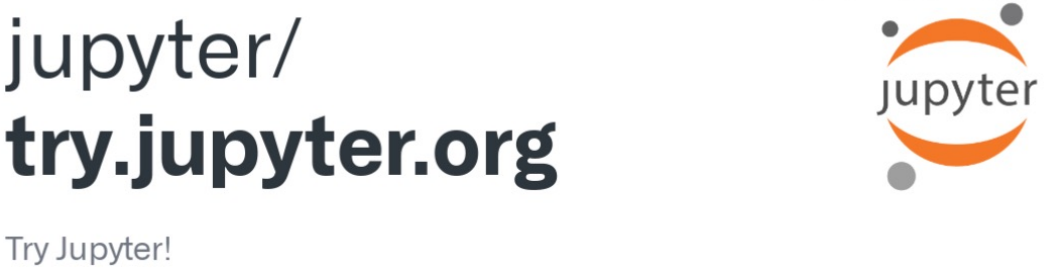
- Tutorial video for installing anaconda on *MacOS*

https://www.youtube.com/watch?v=2JeoNlCcLOM

Free “No Install” Options:

- *https://jupyter.org/try*
- *Google Colab* Online Notebooks
- *Repl.it*
- Google Search:

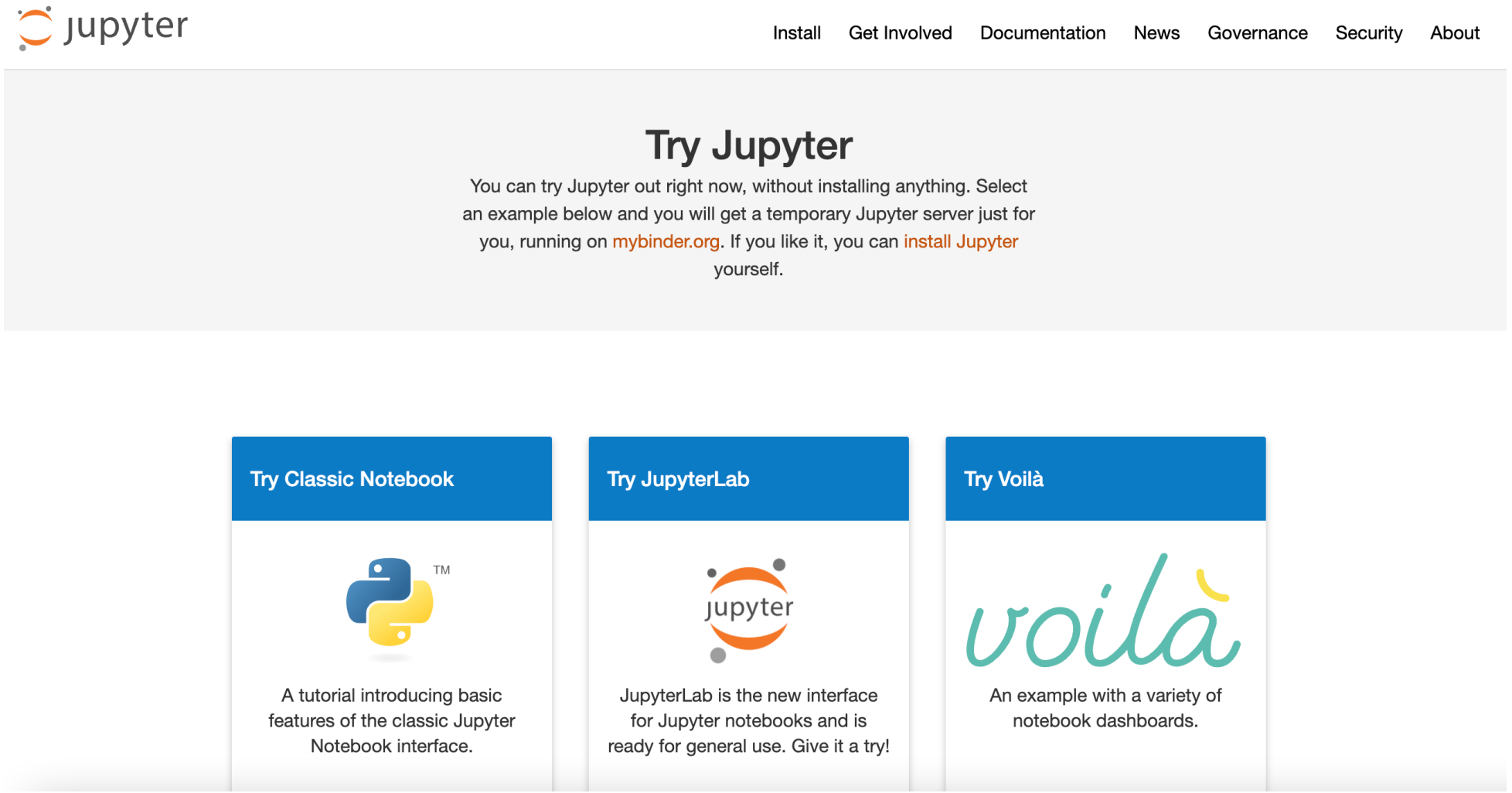
  “Python Interpreter Online”



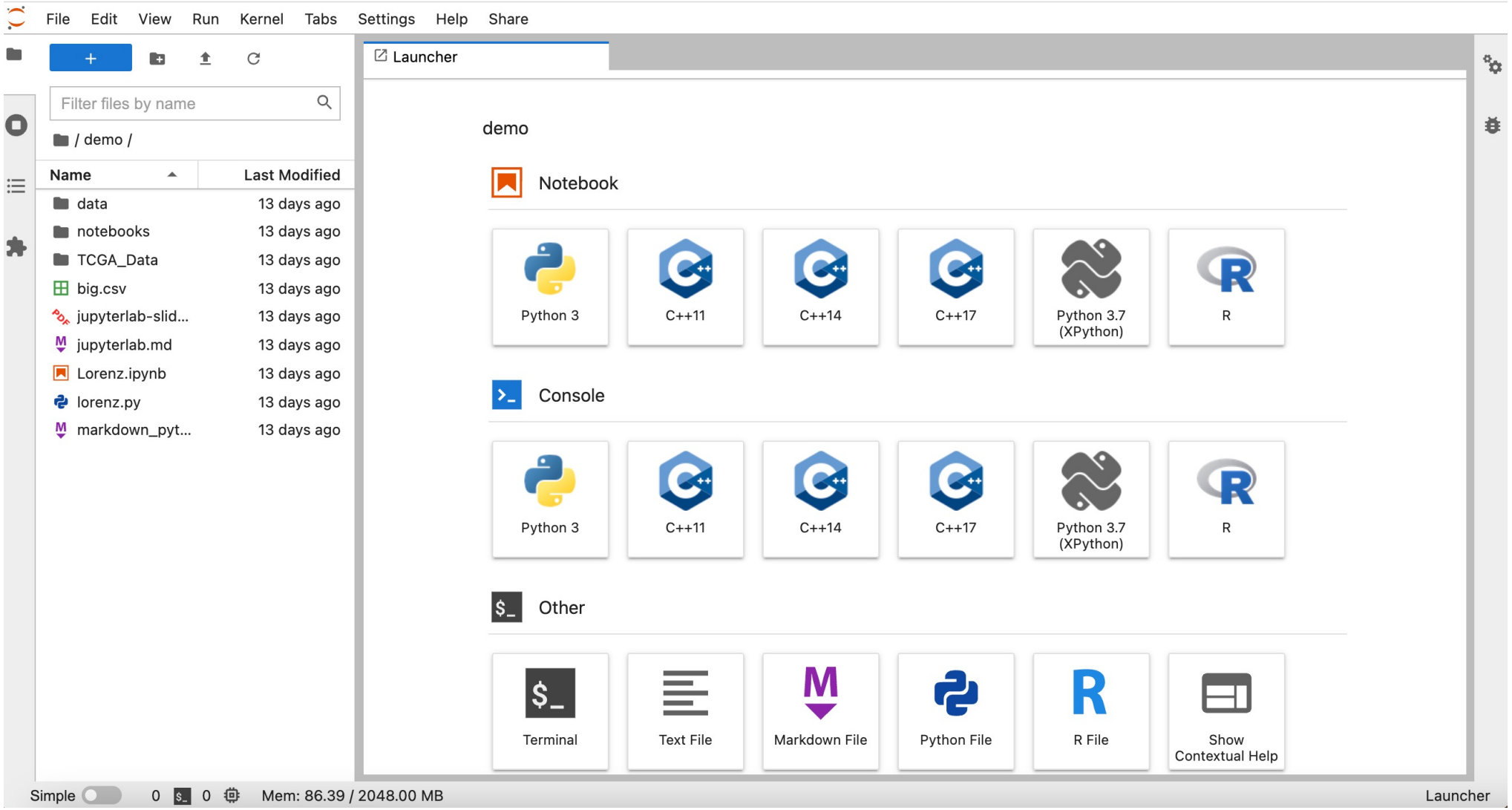
try.jupyter.org

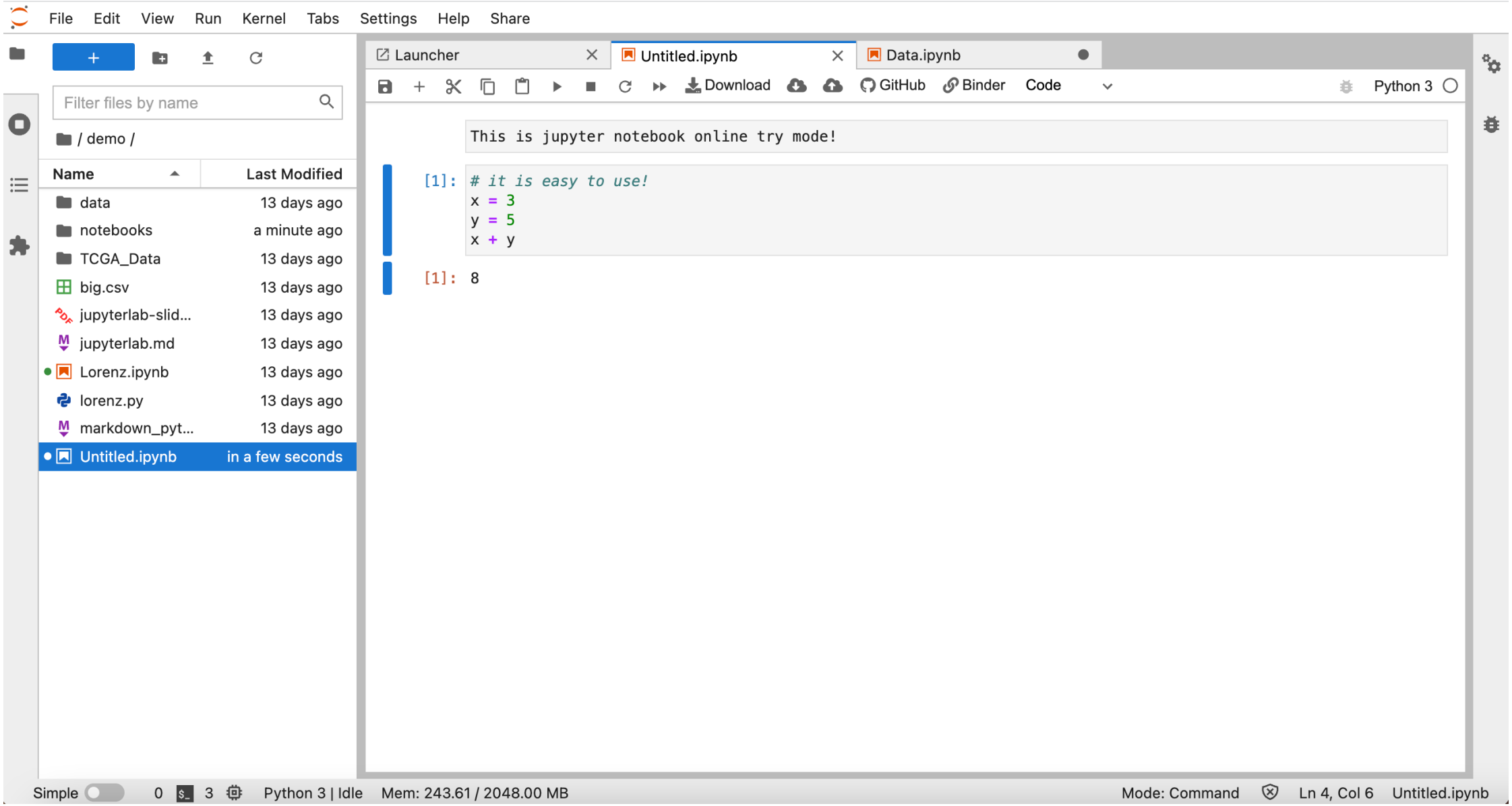
Utrecht University

Click “*Try JupyterLab*”



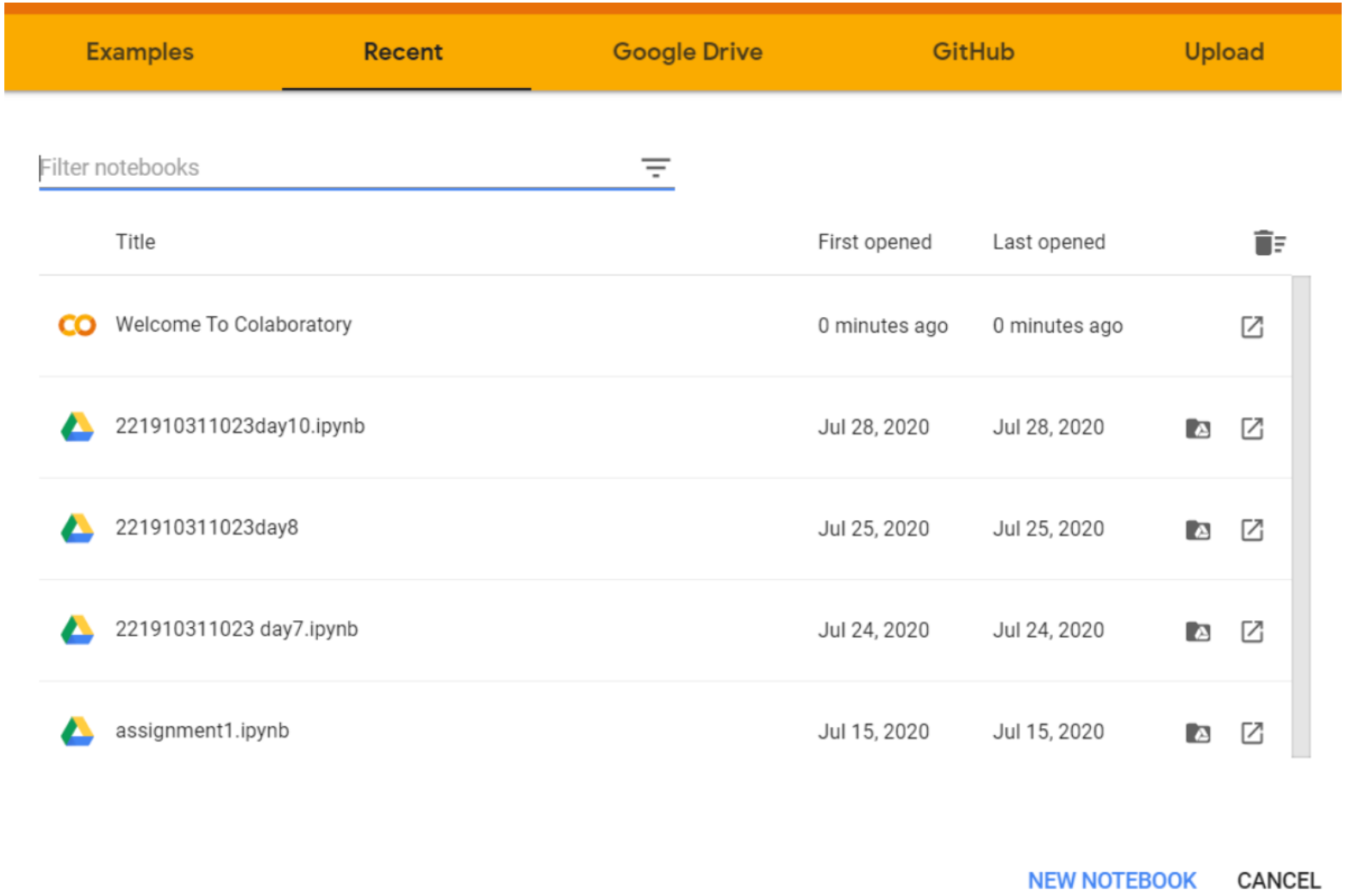
Once you've landed in the *Launcher* page, you can start working on your own file by clicking:
"File → New → e.g. Notebook"




File Edit View Run Kernel Tabs Settings Help Share
Launcher
Untitled.ipynb
Data.ipynb
Download
GitHub
Binder
Code
Python 3
Filter files by name
/ demo /
Name
Last Modified
data 13 days ago
notebooks a minute ago
TCGA_Data 13 days ago
big.csv 13 days ago
jupyterlab-slid... 13 days ago
jupyterlab.md 13 days ago
Lorenz.ipynb 13 days ago
lorenz.py 13 days ago
markdown_pyt... 13 days ago
Untitled.ipynb in a few seconds
This is jupyter notebook online try mode!
[1]:
# it is easy to use!
x = 3
y = 5
x + y
[1]: 8
Simple
0
3
Python 3 | Idle
Mem: 243.61 / 2048.00 MB
Mode: Command
Ln 4, Col 6
Untitled.ipynb

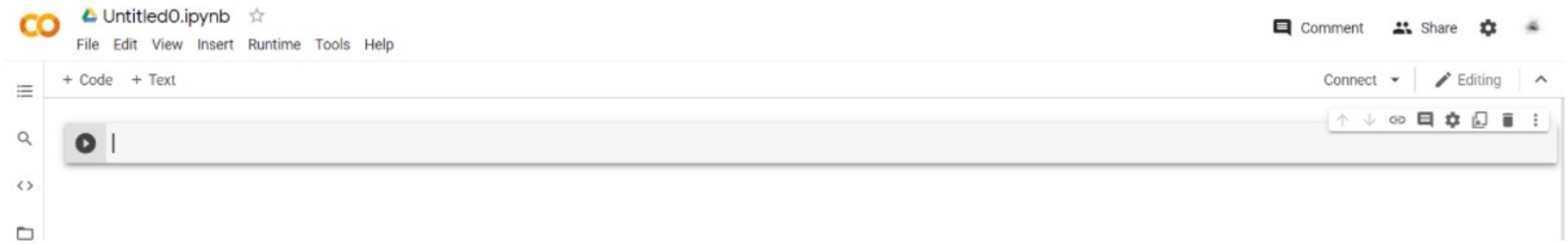
"Google Colab" https://colab.research.google.com/

- To start using Google Colab, you first have to log in to your Google account.
- The home screen of Google Colab will look like this.



- To open a new Python notebook, click ‘new notebook’ in the bottom right corner.

- The opened notebook will look like this.



Quick Note

Free “No Install” Options:

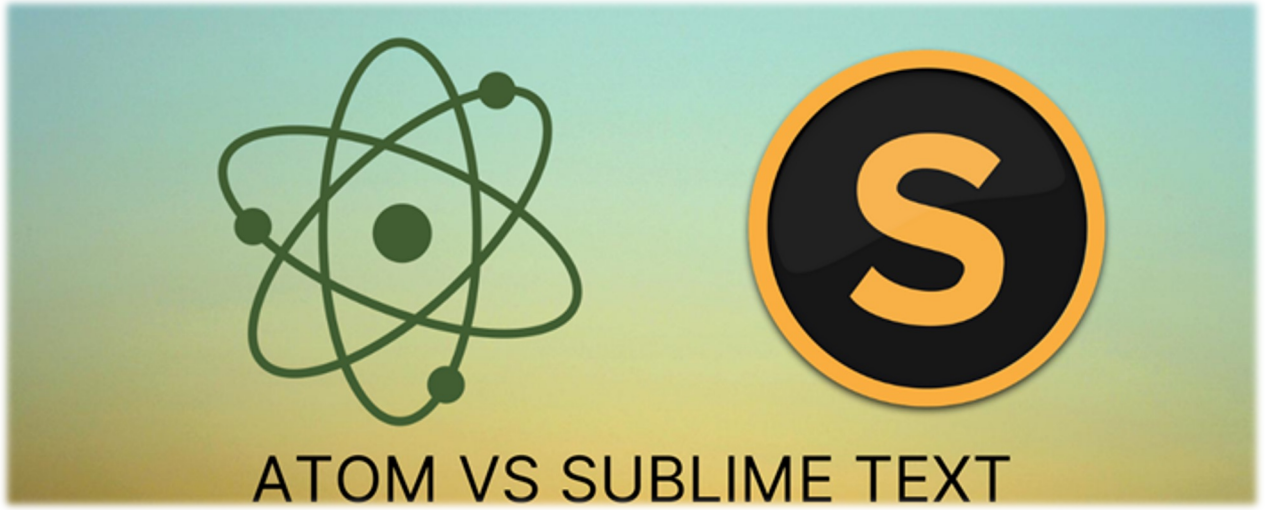
- *Hard to upload your own code, data, or notebooks!*
- *May not save your code in the free version!*

Running Python Code

- There are several ways to run Python code.
- First, let's discuss the various options for development environments.
- There are 3 main types of environments:
  - Text Editors
  - Full IDEs
  - Notebook Environments

Text Editors

Most popular : ***Sublime Text*** and ***Atom***



Full IDEs

- Development Environments explicitly designed for Python.
- Larger programs.
- Designed specifically for Python, with lots of extra functionality.

Most popular: ***PyCharm*** and ***Spyder***

*__Quick note__:

PyCharm is available in three editions: *"Professional", "Community", and "Edu".*

Only the *"Community" and "Edu"* editions are **free**, but they have relatively fewer features available.

Utrecht University

Notebook Environments

- Great for learning.
- See input and output next to each other.
- Support in-line markdown notes, visualizations, videos, and more.
- Special file formats that are *not .py*

Most popular: ***Jupyter Notebook***

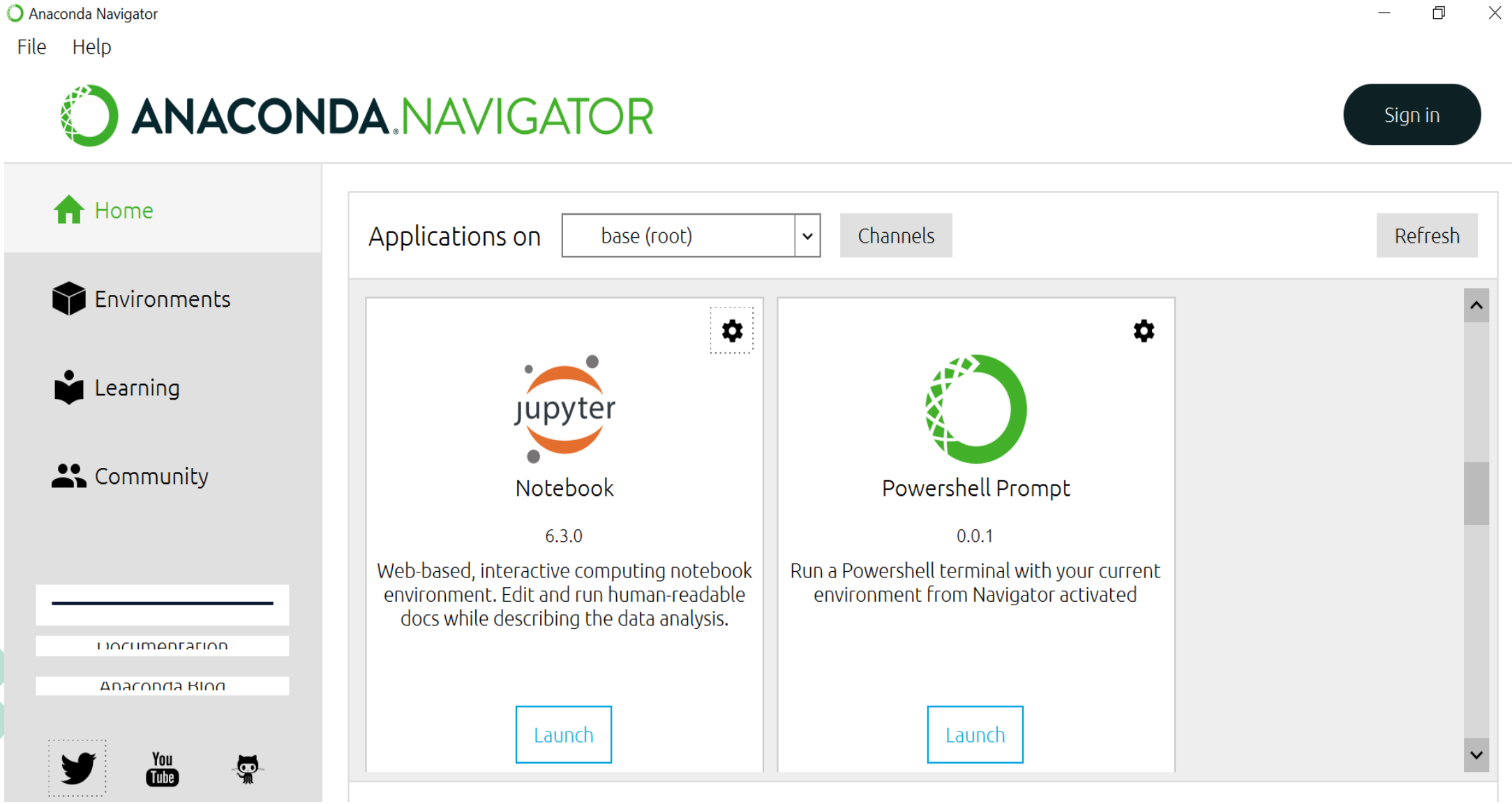
Utrecht University

Choose whichever development environment you prefer!

Here we work with *Jupyter notebook*.

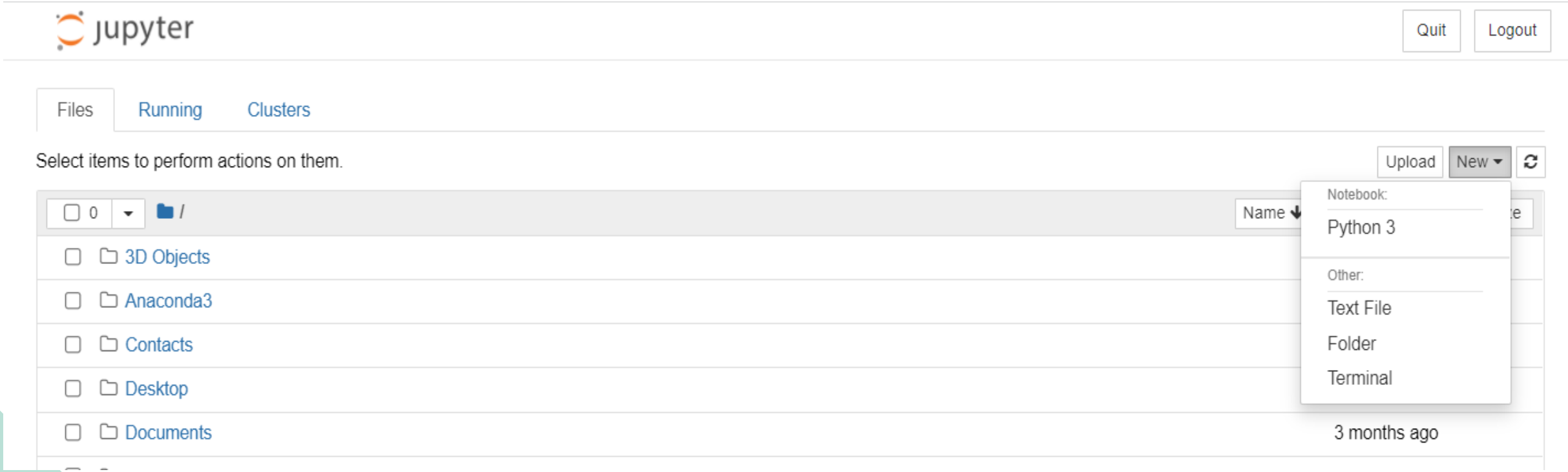
Course Notebooks

Initiate Anaconda and launch Jupyter Notebook.



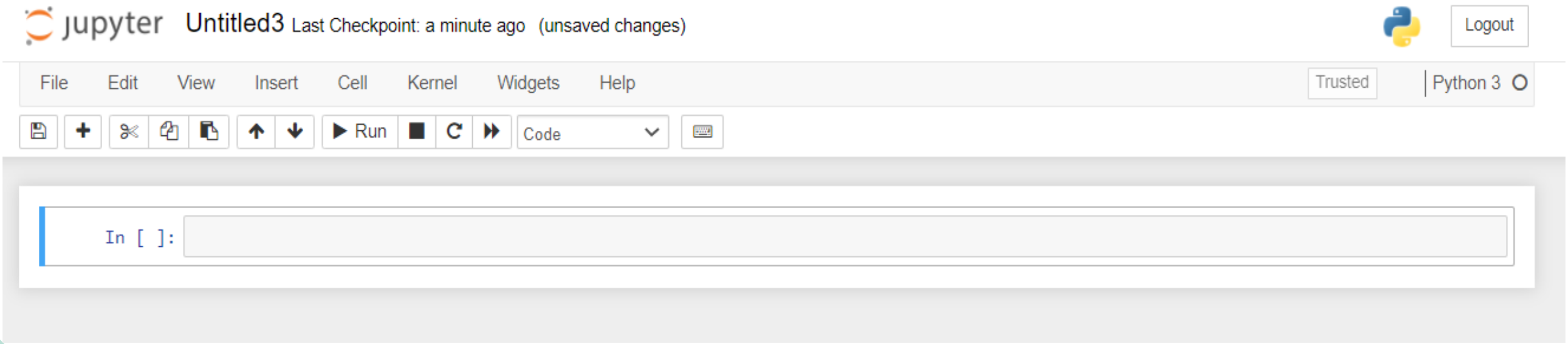
You will see the following Juypter Notebook page open in your browser.

Click "*New → Python3*" to start with a new notebook file.



Utrecht University

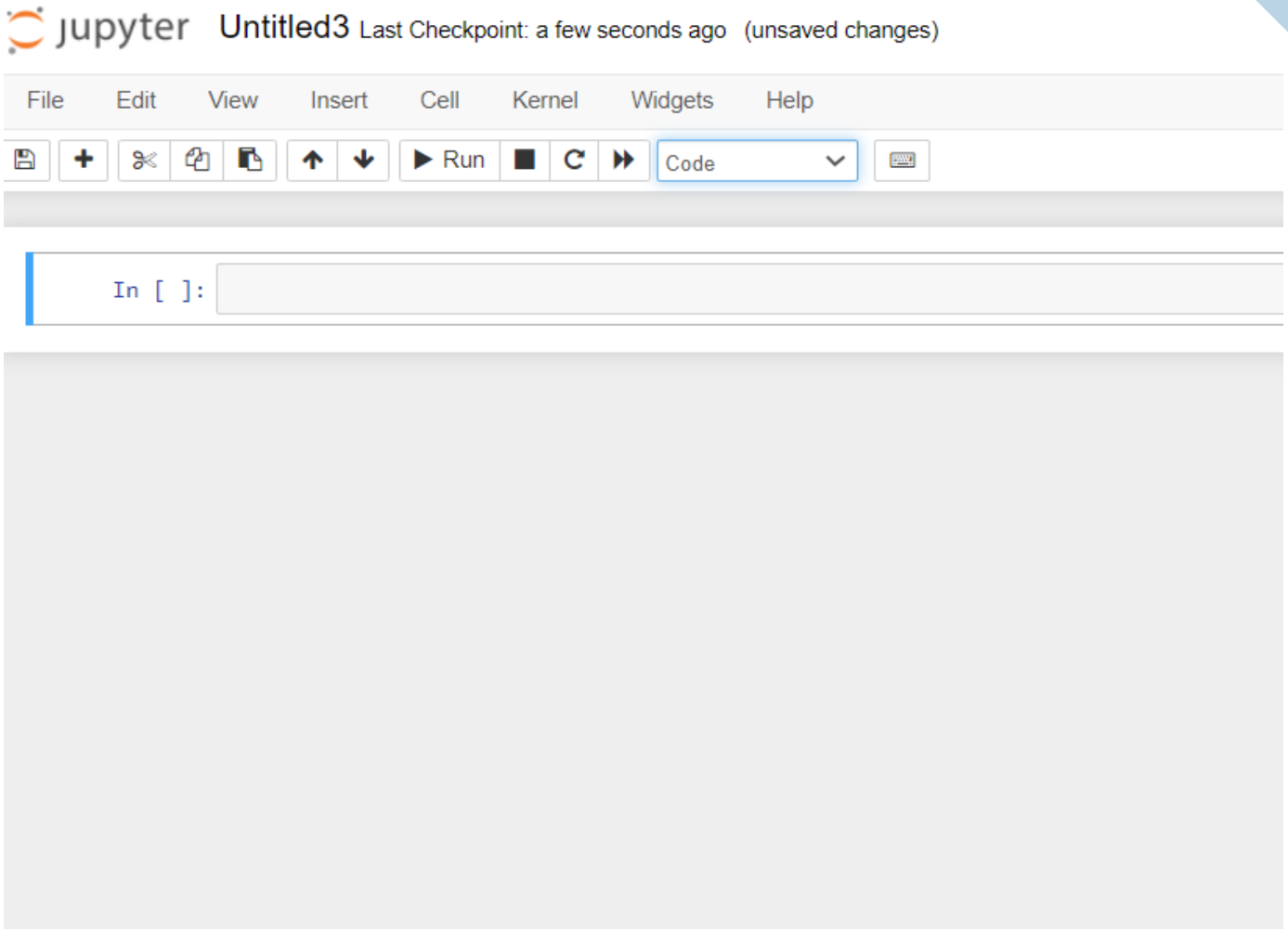
You will see an Untitled notebook page with an empty cell like the one below.



CELLS IN JUPYTER NOTEBOOK

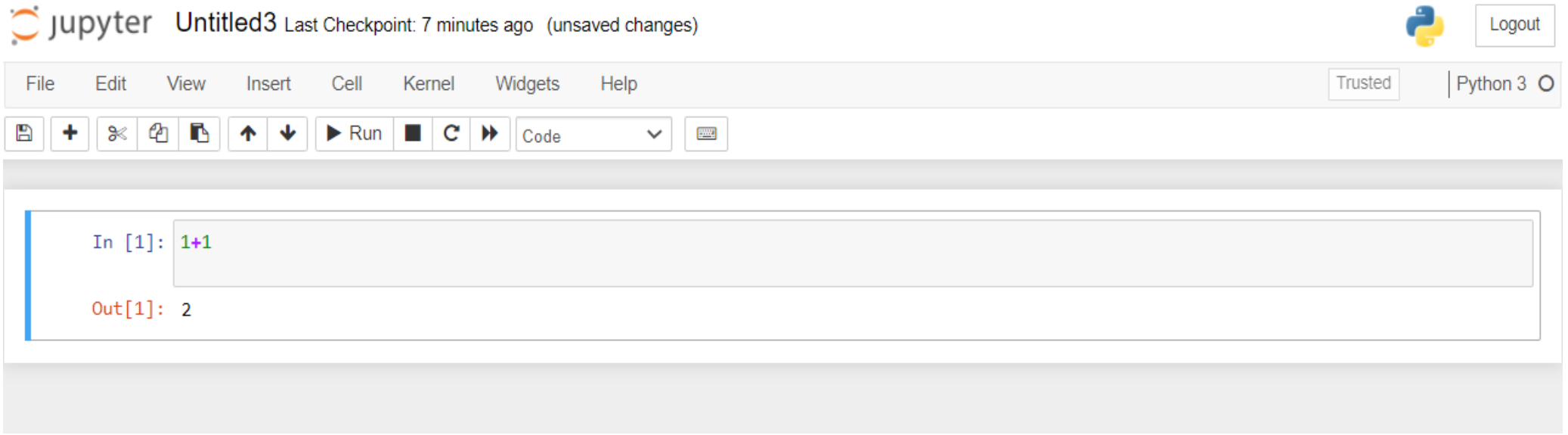
- Two types of cells: **Markdown** and **Code** cells

- Markdown cells : For text
- Code cells: For python code



RUN PYTHON

- Write some code in a code cell (the default one) and click the “run (play) button” (shortcut: “*Ctrl+Enter*”).



“Running” TAB

- In the Jupyter Notebook dashboard, there is a tab named “*Running*” that shows all running notebooks (i.e. kernels) and allows shutting them down (by clicking on a “*Shutdown*” button).

WHY *JUPYTER NOTEBOOK* ?

- Allow for interactive use.
- Can combine text, code, and plots easily.
- *Jupyter notebook* allows you to do fancy things. For instance :

  Simply hit the "*Tab*" key while writing code. This will open a menu with suggestions. Hit "*Enter*" to choose the suggestion.

- Help if you type "*?*" (example in next cell)
- Magic cells like "*%matplotlib inline*" makes the plot inside the notebooks.
- Shortcuts: "*Ctrl+Enter*": Run cell. "*Shift+Enter*": Run cell and go to the next one.
- Check the link below for other useful tricks :

https://towardsdatascience.com/15-tips-and-tricks-to-use-jupyter-notebook-more-efficiently-ef05ede4e4b9