DATA
Jupyter Notebook
Tutorial

Simos Gerasimou

What are Jupyter Notebooks?

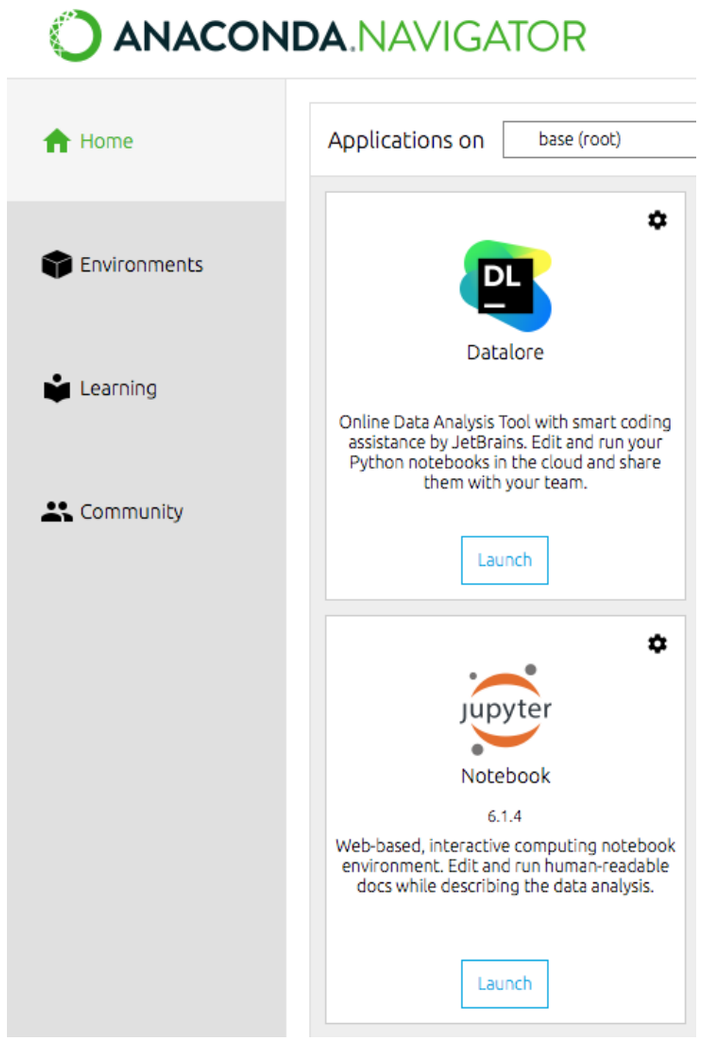
- A Jupyter notebook is an electronic file (in .ipynb format) that contains both programming code and text descriptions.
- A notebook can also contain embedded charts, plots, images, videos, equations, visualisations, links.
- A notebook runs in a web browser (e.g., Firefox, Google Chrome).
- It is maintained by Project Jupyter.
- The Python code in a notebook is the same type of Python code found in a .py file.
- Notebooks can be exported to a variety of formats including .html and .pdf.
- You can find more information at https://jupyter.org
- The text description sections of Jupyter notebooks include explanations and clarifications are provided in the markdown format.

- **Markdown commands**: https://guides.github.com/features/mastering-markdown

Jupyter Notebook Installation

Using Anaconda

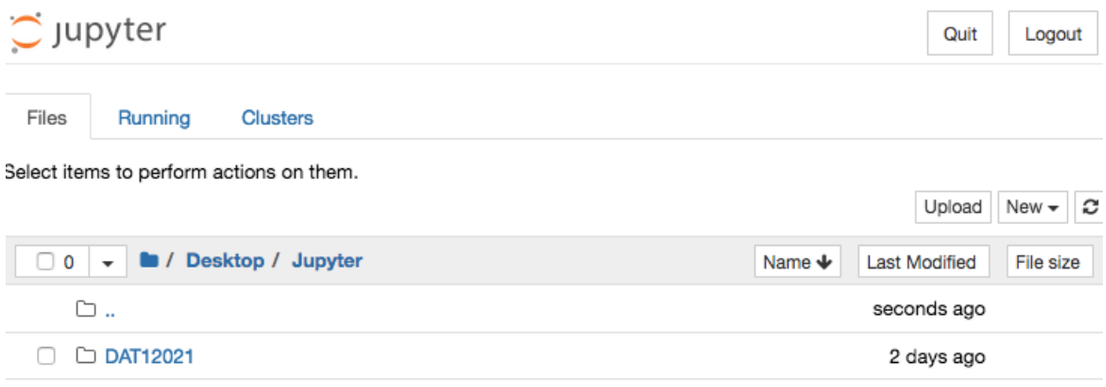
- A widely used Python distribution for data science pre-loaded with popular libraries
- Go to https://www.anaconda.com/products/navigator
- Select the installer for your operating system. The download should start shortly.
- When the download is completed, run the setup file.
- When the installation is complete, run the Anaconda Navigator
- Jupyter Notebook comes pre-installed with Anaconda
- If it is not preinstalled, then click the install button below the Jupyter Notebook sign once the Anaconda Navigator opens
- Click Launch
- **If everything went well, the Notebook Dashboard should open in a web browser tab**

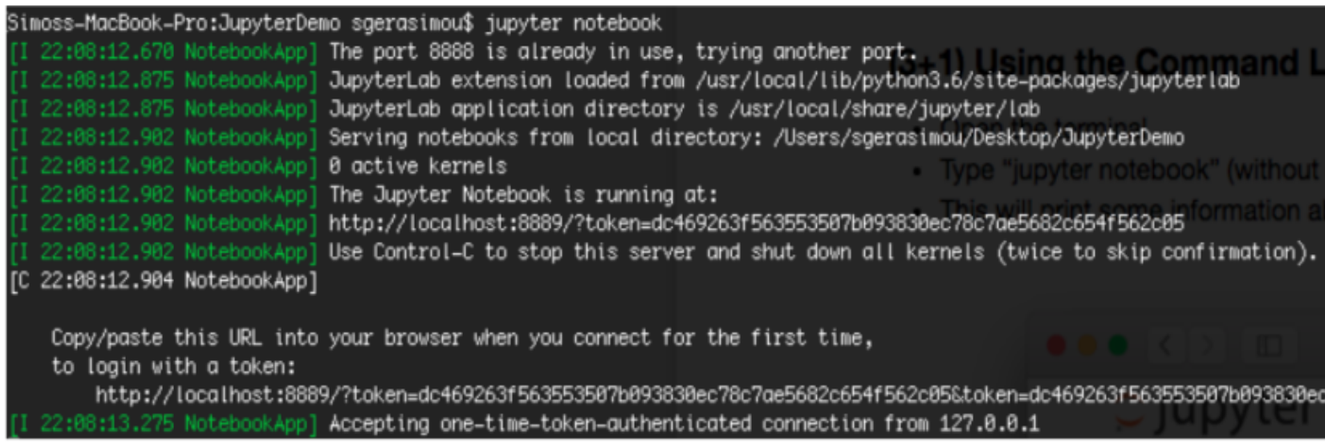


Jupyter Notebook Installation

Using pip (Python's package management system)

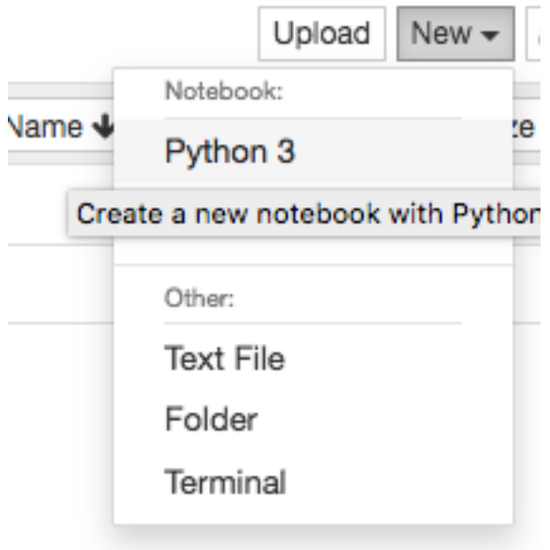
- Open terminal
- Run: pip3 install jupyter
- When installation is complete, type "jupyter notebook" (no quotes) and hit the return key
- This will print some information about the notebook server in your terminal, including the URL of the web application (by default, http://localhost:8888)
- **If everything went well, the Notebook Dashboard should open in a web browser tab**



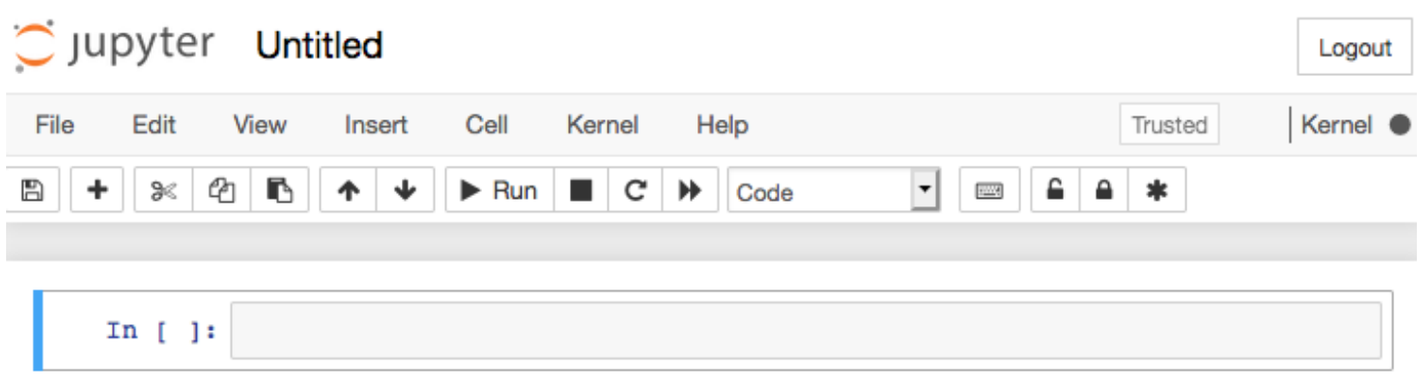


Creating a New Notebook

- Click on the New button (upper right), and it will open up a list of choices

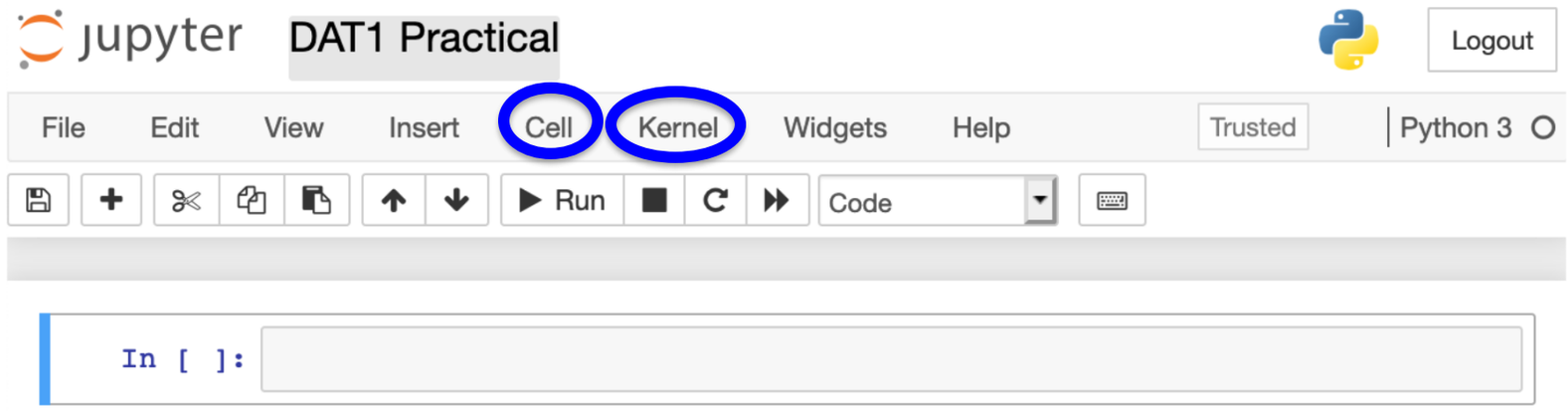


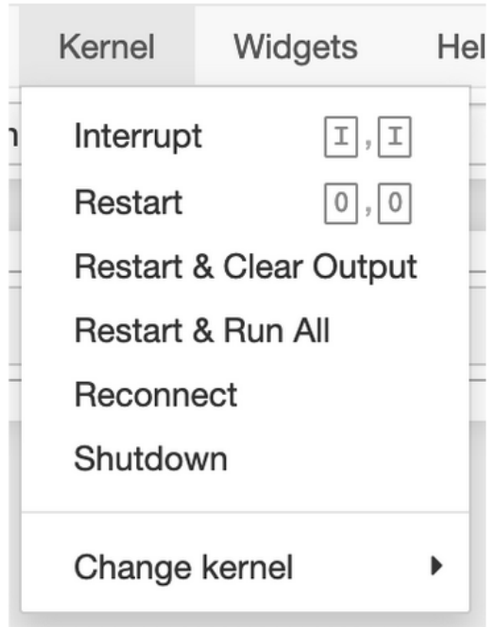
- Select Python 3
- **A new notebook will open as a new tab in your web browser.**
- Each notebook is stored as an **.ipynb file**
  - You should give it a reasonable name (e.g., DATA_Practical1); currently it is "Untitled"
    - Click on the “Untitled” text field



The Notebook Interface

- Two primary features of a notebook are: **Cell** and **Kernel**





- **Kernel:** a "computational engine" that executes the code contained in a notebook document and produces the results.
  - In this case, it is a Python kernel
  - The kernel can be restarted (e.g., if execution takes a long time)

- **Cell:** a container for text to be displayed in the notebook or code to be executed by the notebook's kernel

Code Cells

- **A notebook contains three types of cells: code cells, output cells, and markdown cells.**
- **Code cells**: contain Python code that could be executed
- Hint: If the cell-type drop-down menu shows Code, then you are at a code cell.


```
In [ ]: #Python code that imports numpy and calculates the square root of 36
        import numpy as np
        x = np.sqrt(36)
        print("the root of 36 is:", x)
```

- **Brackets next to cell:** if the **[…]** to the left of a cell shows a number then it means that the cell has been evaluated! This means that the code written in that cell has been run. If it is **[]** then the cell has not been run yet. If it is **[*]** then the code is still evaluated.

- **Executing the code in a cell:** Click the [Run] button or type [Shift]+[Enter].

Output Cells

- **A notebook contains three types of cells: code cells, output cells, and markdown cells.**
- **Output cells**: Contain the output from running the code cells as well as charts, plots, command line output, and images
- Not all code produces output, so not all code cells produce output cells.
- The results in output cells can't be edited.

```
In [4]: #Python code that imports numpy and calculates the square root of 36
        import numpy as np
        x = np.sqrt(36)
        print("the root of 36 is:", x)
```

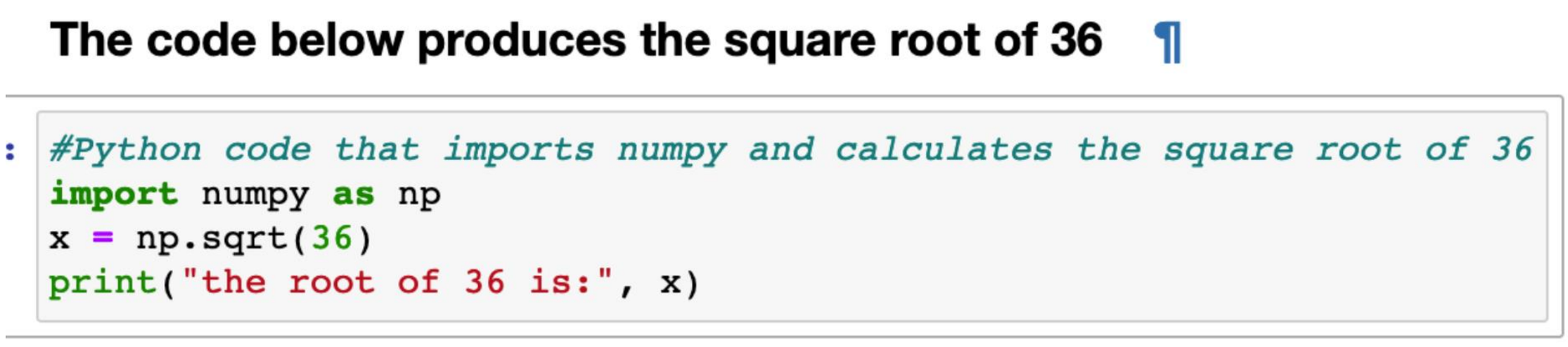
```
the root of 36 is: 6.0
```

- **Brackets next to cell:** if the **[...]** to the left of a cell shows a number then it means that the cell has been evaluated! This means that the code written in that cell has been run. If it is **[]** then the cell has not been run yet. If it is **[*]** then the code is still evaluated.

Markdown Cells

- **A notebook contains three types of cells: code cells, output cells, and markdown cells.**
- **Markdown cells**: Contain text-like descriptions that conform to the Markdown syntax (https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet)
- These cells are typically used to explain the code or output (before or above a Markdown cell). They don't contain Python code

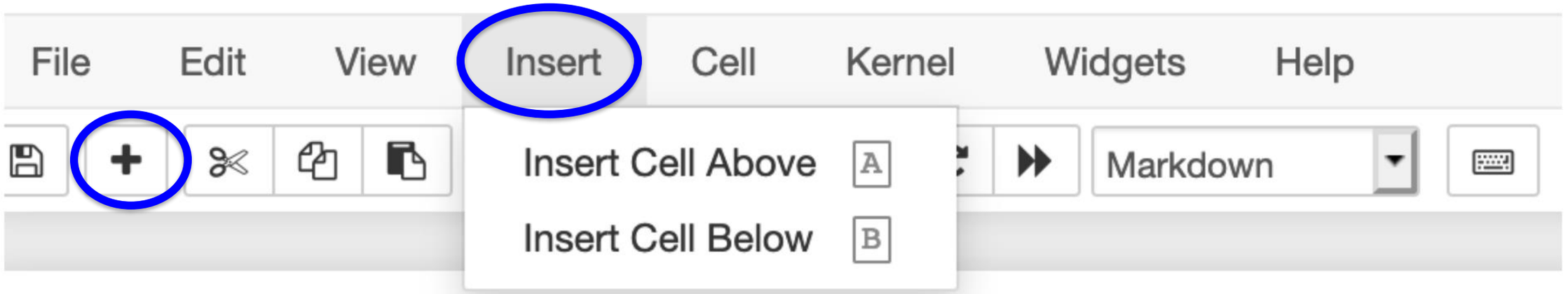
- **Executing markdown in a cell:** Click the [Run] button or type [Shift]+[Enter]



Creating a New Cell

- You can create a new cell by clicking the [+] button in the upper menu.
- Clicking [+] creates a new code cell below the active cell.

- You can also create a new cell by clicking the **Insert** option from the toolbar
- A cell is inserted above or below the currently selected cell



Getting Help within a Notebook

- There are several ways we can get help within a notebook

- **Using the dir command**
  - Type dir() and provide as input a function, method, variable or object
  - Shows the possible object, method and function calls available to that object
  - e.g., ***dir(math)*** shows the contents of the math module

```
In [11]: import math
         dir(math)

Out[11]: ['__doc__',
          '__file__',
          '__loader__',
          '__name__',
          '__package__',
          '__spec__',
          'acos',
          'acosh',
          'asin',
          'asinh',
          'atan',
          'atan2',
          'atanh',
```

- **Using the Tab completion functionality**
  - Use the [Tab] key after typing the name of a variable, object or function to view the available options.
  - Scroll through the list
  - Use [Enter] to select the desired option.

```
In [12]: import math
         math.a
                  acos
                  acosh
                  asin
                  asinh
                  atan
                  atan2
                  atanh
In [ ]:
In [ ]:
In [ ]:
```

Getting Help within a Notebook

- There are several ways we can get help within a notebook
- **Using the help() function**
  - Type help() and provide as input a function, method, variable or object
  - e.g., ***help(math.cos)*** shows the documentation of the cos function

```
In [14]: import math
         help(math.cos)
```

```
Help on built-in function cos in module math:

cos(x, /)
    Return the cosine of x (measured in radians).
```

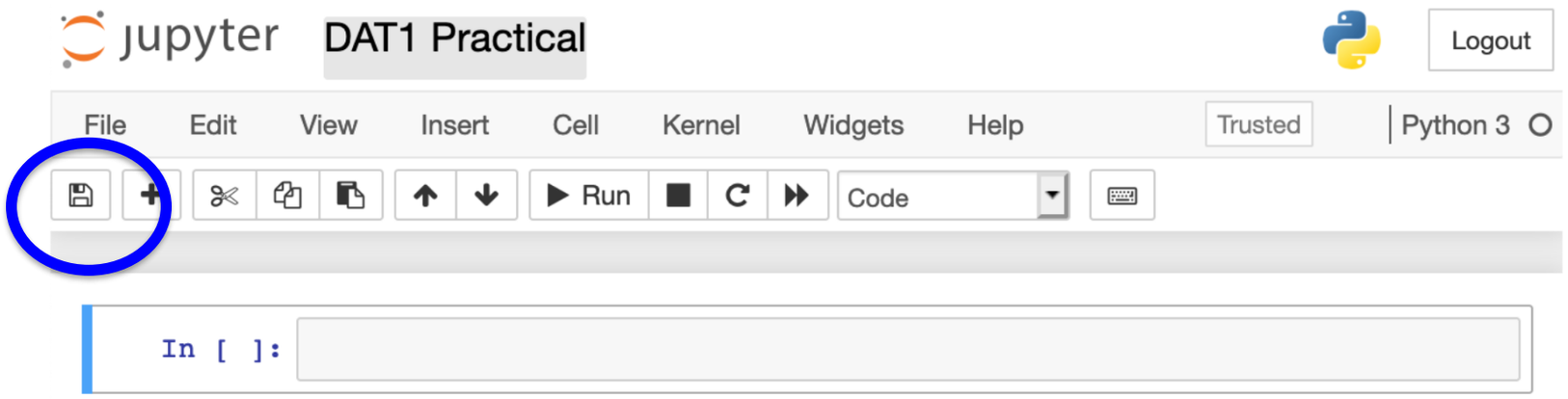
- **Using the question mark ?**
  - Use the [?] after a function to get help and view its source code

```
In [16]: import numpy as np
         np.mean?
```

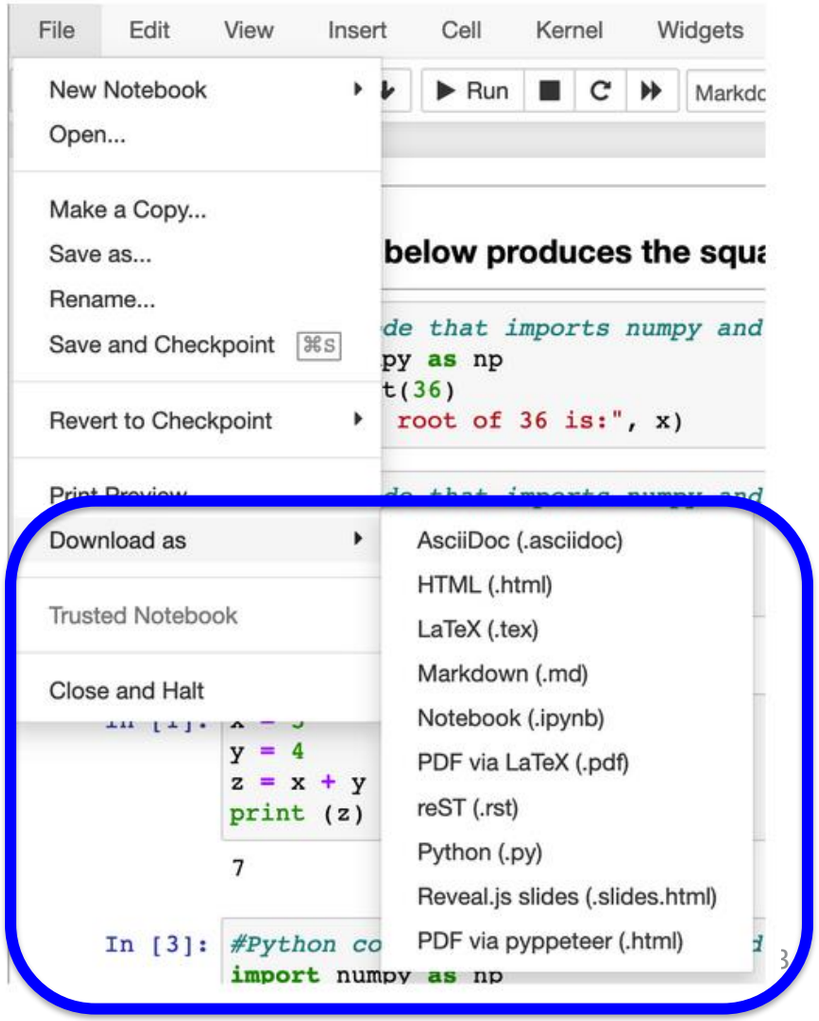
```
Signature: np.mean(a, axis=None, dtype=None, out=None, keepdims=<no value>)
Docstring:
Compute the arithmetic mean along the specified axis.

Returns the average of the array elements.  The average is taken over
the flattened array by default, otherwise over the specified axis.
`float64` intermediate and return values are used for integer inputs.

Parameters
----------
a : array_like
    Array containing numbers whose mean is desired. If `a` is not an
```

Saving the Jupyter Notebook

- In Jupyter Notebook format (.ipynb)

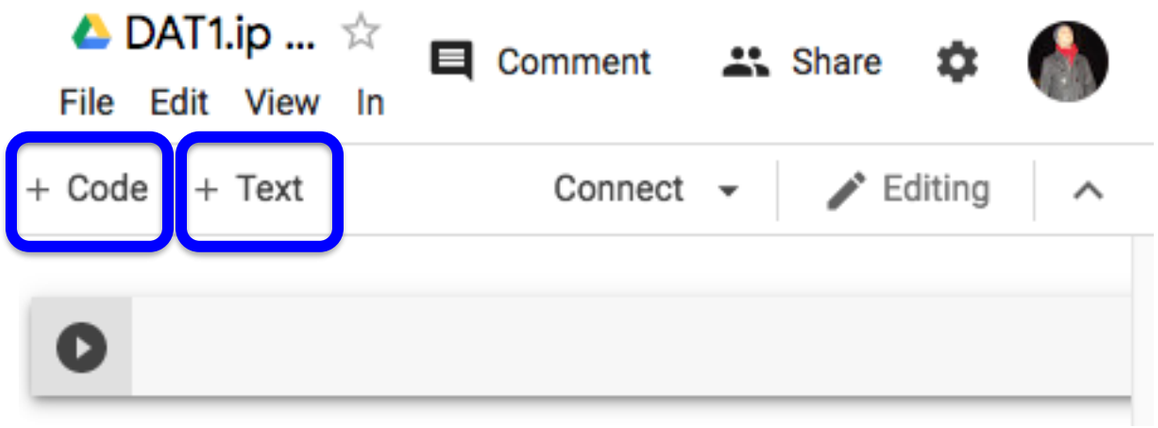


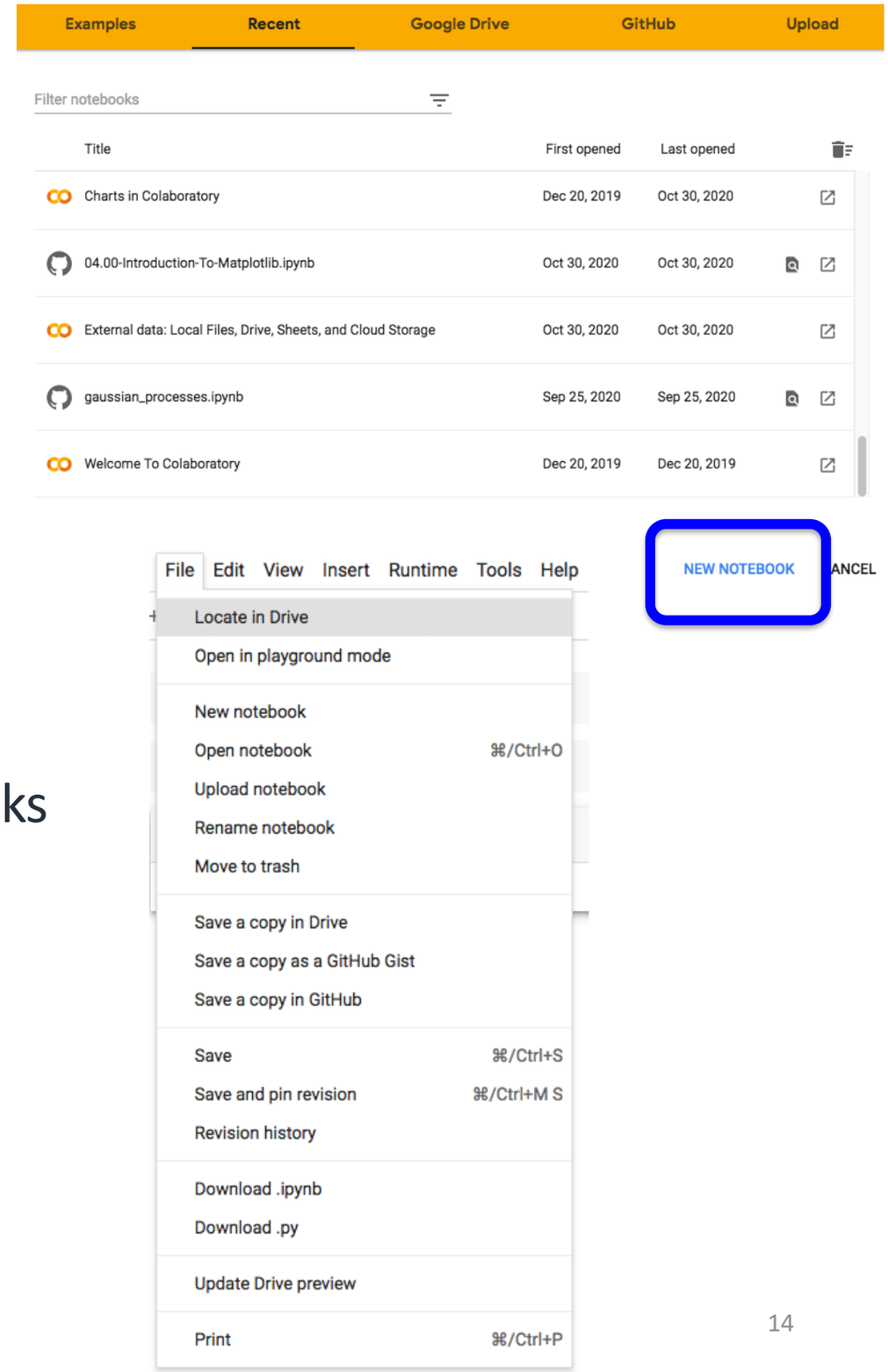
- Downloading in other formats
  - File > Download as
  - e.g., pdf, html, latex
  - You can also export the Python code only



Using Google Colab Notebook

- Go to https://colab.research.google.com
- Select “NEW NOTEBOOK”
- A new tab will open





- Follows the same principles with Jupyter Notebooks
- You can download the .ipynb or Python files
- You can save it to you Google Drive etc

Useful Resources

- Video: Jupyter Notebook Tutorial: Introduction, Setup, and Walkthrough
  https://www.youtube.com/watch?v=HW29067qVWk

- Introduction to Google Colaboratory
  https://colab.research.google.com/notebooks/intro.ipynb
  https://colab.research.google.com/notebooks/basic_features_overview.ipynb