Anaconda (for Python)

Installing Anaconda
Working with Jupyter Notebooks

Installing Anaconda

Anaconda is available for Windows, MacOS, and Linux.

It is advisable to update conda and all packages after installation. Open the Anaconda Prompt and in the prompt, run the following commands:

pip install --upgrade pip

conda upgrade conda

conda upgrade --all

and answer yes (y) when asked if you want to install the packages. Note that it might take a while.

1

Download Anaconda from

https://www.anaconda.com/download

2

Find installation instructions here.

3

Validate your installation here.

Installing Packages in Anaconda Prompt

- To install a package: *conda install PACKAGE*
- For example, to install pandas: *conda install pandas.*
- You can install multiple packages at the same time. For example: *conda install numpy pandas matplotlib seaborn*
- Some packages are dependent on another.
- For example, pandas uses and requires numpy.
- If you install just pandas (*conda install pandas*), Conda will also install numpy if it isn't already installed.
- To see all installed packages: *conda list*

Updating and Removing Packages in Anaconda Prompt

- To update a package, use: *conda update PACKAGE*
- If you want to update all packages: *conda update –all*
- To remove a package: *conda remove PACKAGE*

Managing Conda Environments

Conda can be used to create environments to isolate your projects.

To create an environment, use the following command in the prompt:

conda create -n ENV_NAME python=#.# LIST_OF_PACKAGES

For example:

conda create -n uranus_env python=3.9 numpy pandas matplotpib

1

-n env_name is to set the environment name (-n for name)

2

(Optional) You can specify the Python version, e.g., python=3.9.

3

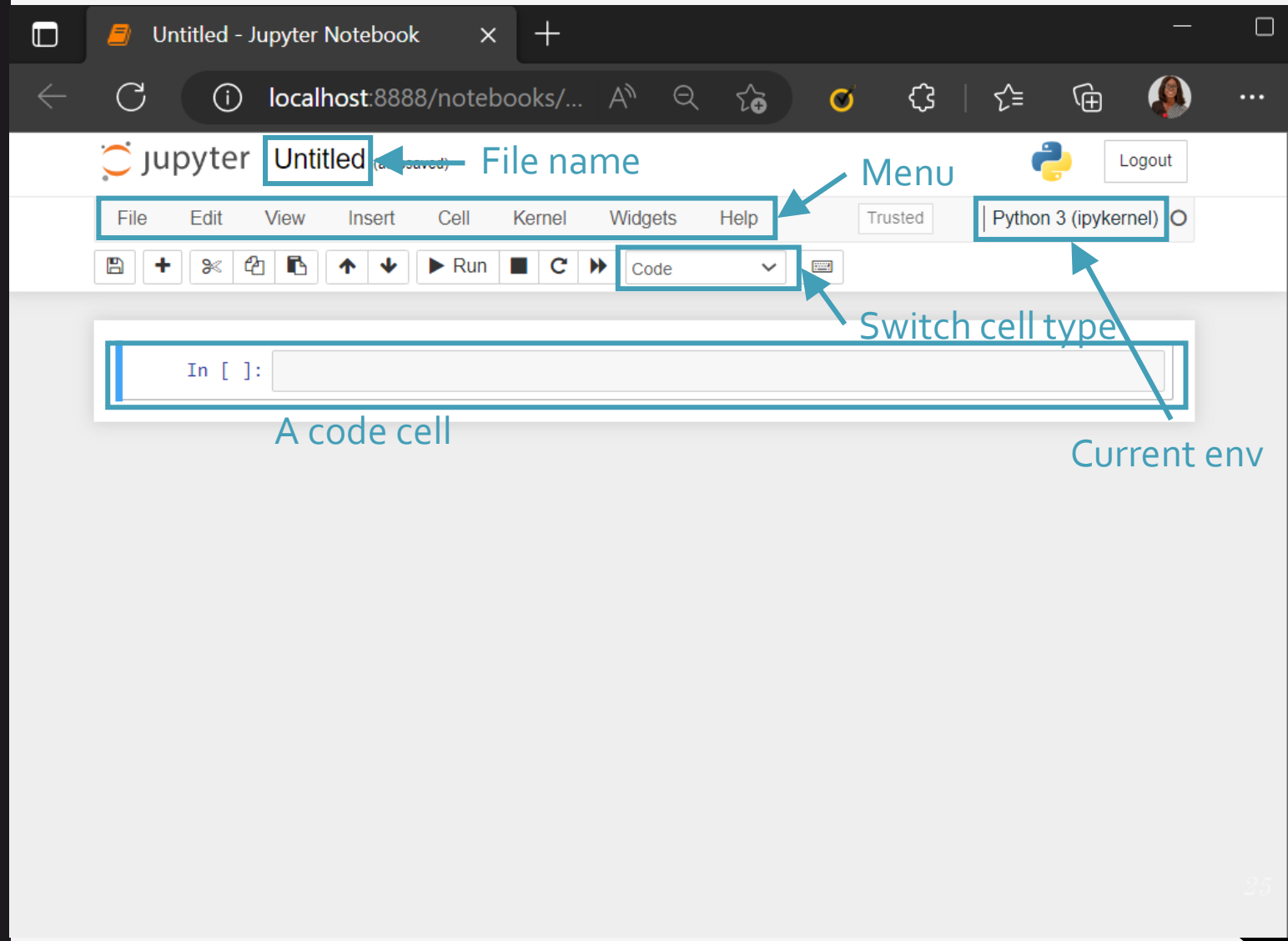
(Optional) You can list packages you want in the environment with LIST_OF_PACKAGES

[De]Activating Environments

- To activate an environment, use: *conda update ENV_NAME*
- To deactivate the current enviroment: *conda deactivate*
- Deactivating an environment takes you back to the default environment

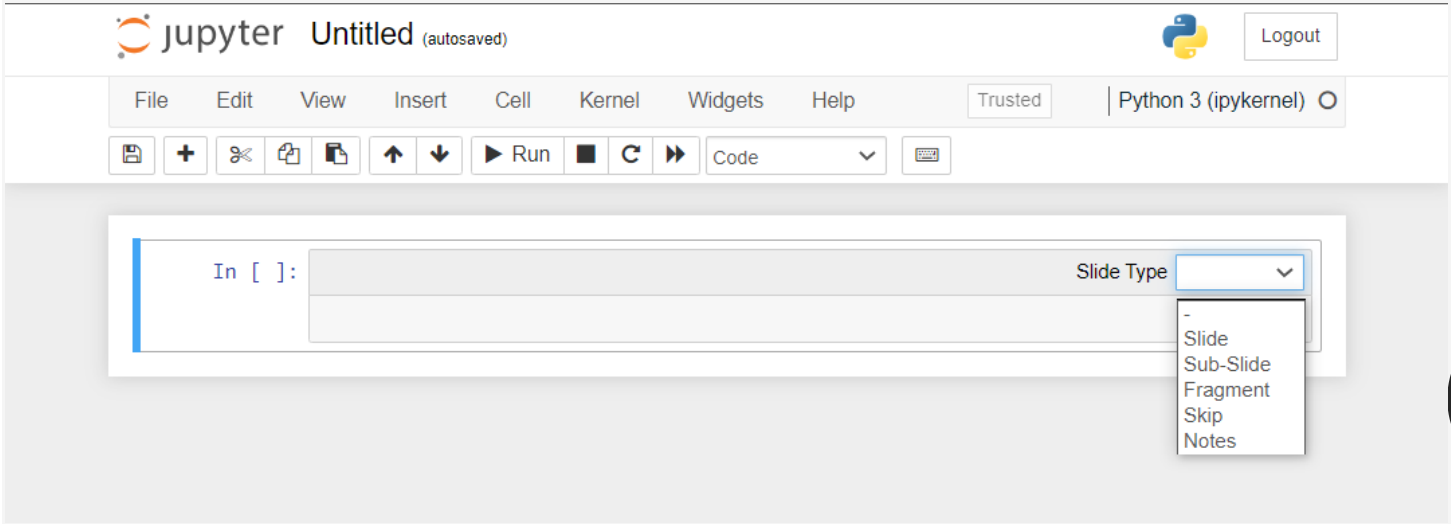
Jupyter

Working with Jupyter Notebooks



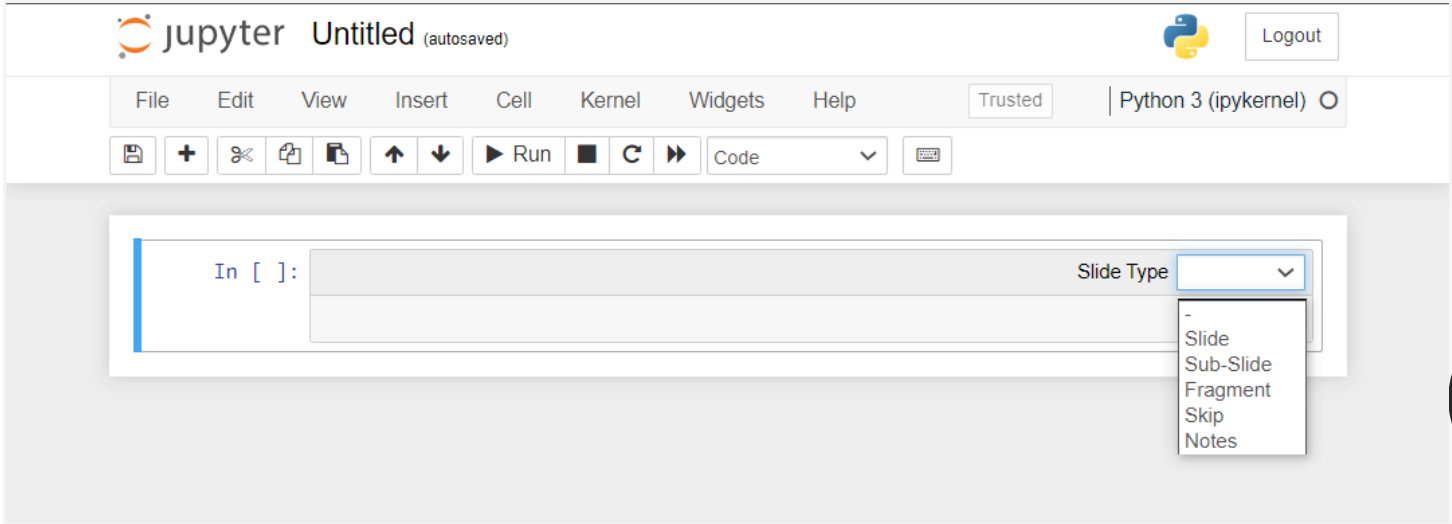
Turning Jupyter Notebooks to Slides

- You can view an example here.
- Click on View on the Menu bar
- Click on Cell Toolbar and click on Slideshow
- This will display a toolbar on each cell to choose how they appear on the slideshow



Turning Jupyter Notebooks to Slides

- Slides are full slides that you move through left to right.
- Sub-slides are animated in the slideshow by pressing up or down.
- Fragments appear with a button press.
- You can skip cells with Skip and
- Notes leaves the cell as speaker notes.



Turning Jupyter Notebooks to Slides

- To create the slideshow from the notebook file, you'll need to use nbconvert: *jupyter nbconvert notebook.ipynb --to slides*
- This converts the notebook to an html file that can be presented.
- To open the file in browser immediately after conversion: *jupyter nbconvert notebook.ipynb --to slides --post serve*