Python Environnement

Python version

Python 2 or Python 3 ? Definitely 3

- https://www.python.org/
- https://pythonclock.org/

Python distributions

Why Python ?

- The rich ecosystem of libraries and tooling, and the convenience of the language itself, make Python an excellent choice.

Many distributions of Python, such as:

- WinPython,
- ActivePython,
- Anaconda,
- Enthought Canopy,
- Python(x,y),
- Pyzo

Python distributions

Why Python ?

For this course, let's install Anaconda for python 3

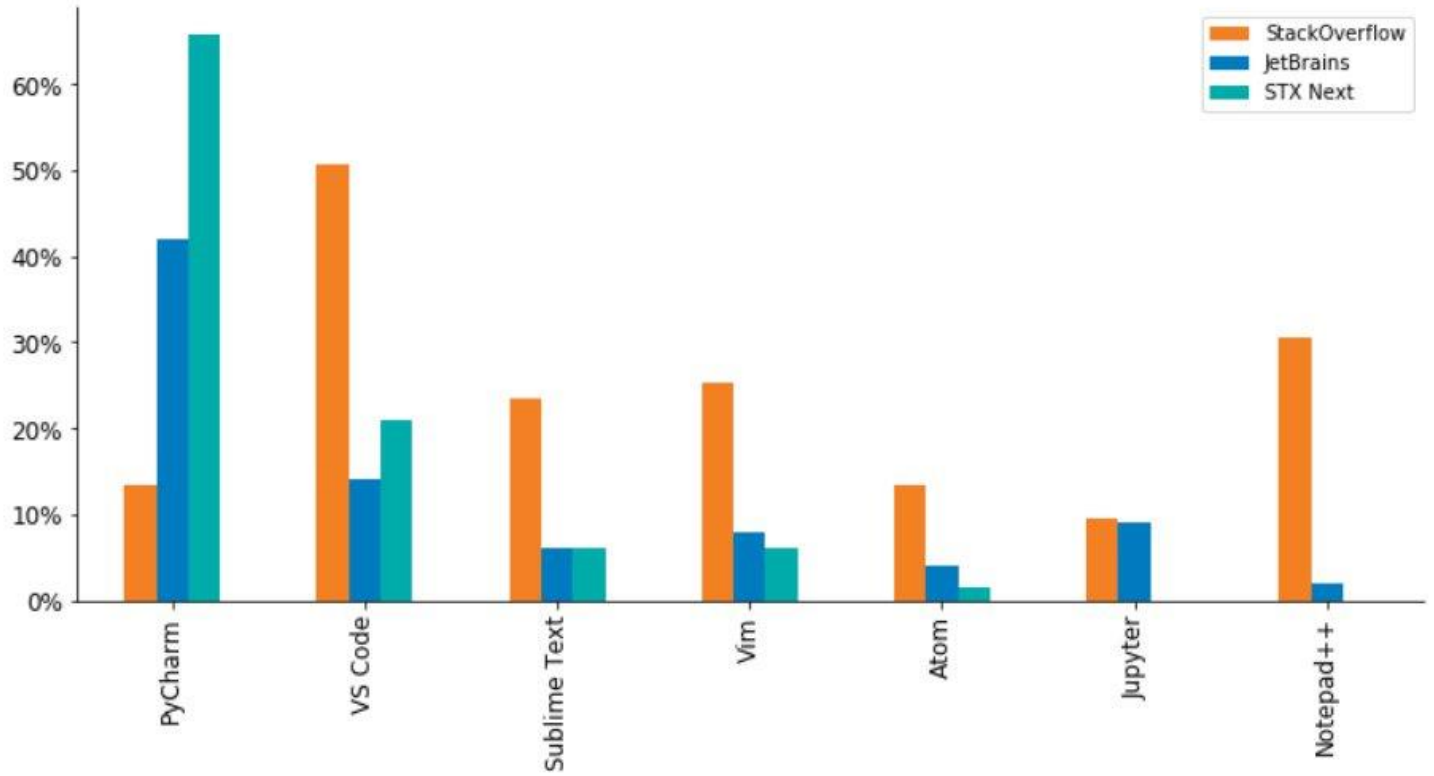
- https://www.anaconda.com/products/individual

Python IDE

Which Python IDE?

An IDE (Integrated Development Environment) is a software application used by developers for creating programs.

- Python IDEs for Data Science
  - Spyder
  - **PyCharm**
  - Atom
  - **Jupyter Notebook**
  - **Jupyter Lab**
  - **Visual Studio Code**
  - **PyDev**



Source: https://www.stxnext.com/blog/best-python-ides-code-editors/

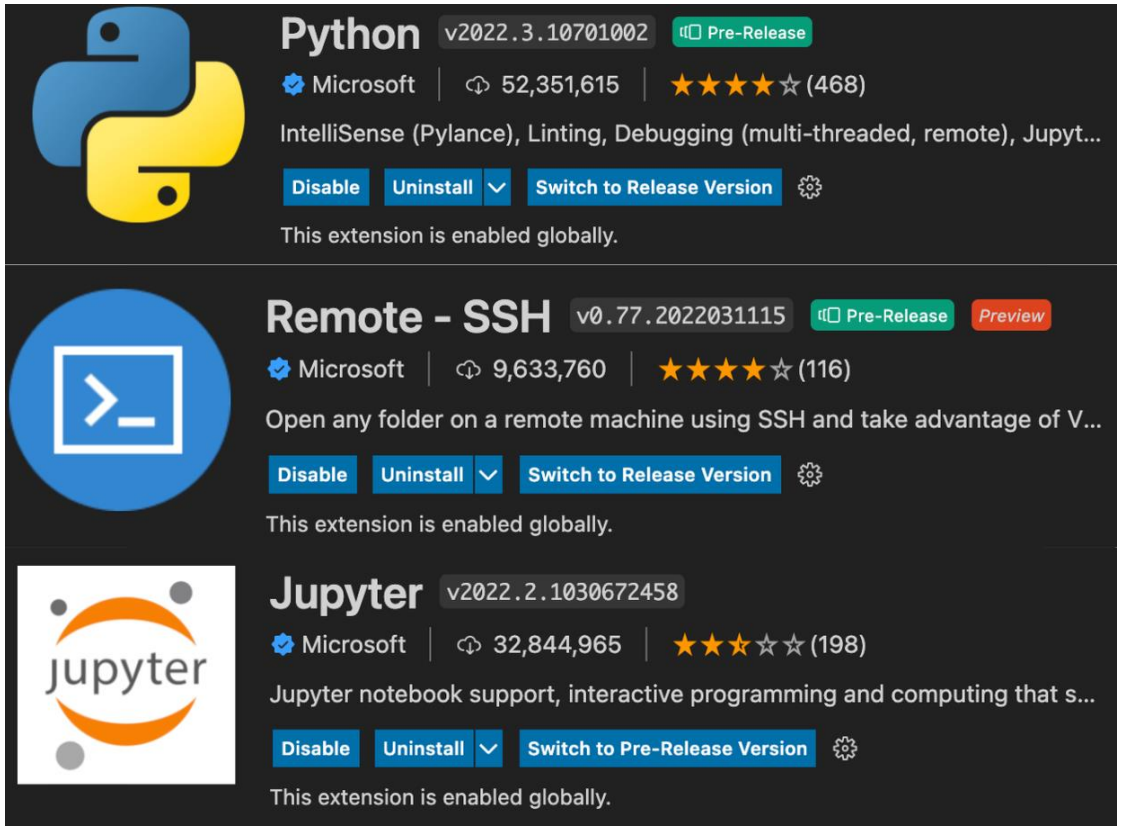
Python IDE

Python IDEs for Data Science

- http://www.jupyter.org/

and

- try it with python in http://jupyter.org/try
  - Code Cell
  - Markdaown Cell



Next, we will user Jupyter Notebooks Extension for **Visual Studio Code**

Python Virtual Environment

Why Virtual Environment?

- Virtual environment = a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages

- Allows virtual installations of Python and libraries on your computer
  - Multiple versions of Python or libraries and easily activate or deactivate these environments
  - Sometimes you'll want to program in different versions of a library. For example:
    - You develop a program with SciKit-Learn 0.17
    - SciKit-Learn 0.18 is released
    - You want to explore 0.18 but don't want you old code to break

Python Virtual Environment

Why Virtual Environment?

- Several ways to create a virtual environment
  - Using venv module of python (default)
    - **python3 -m venv tutorial-env**
    - **source tutorial-env/bin/activate**
    - **pip3 search astronomy**
    - **pip3 install novas or pip install novas=2.1.0**
    - **source deactivate**
  - Using Virtualenv Environment
  - Using Pipenv Environment

Python Virtual Environment

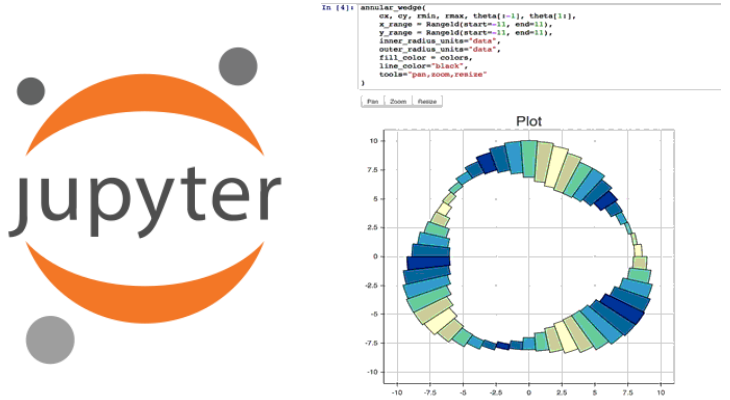
Why Virtual Environment?

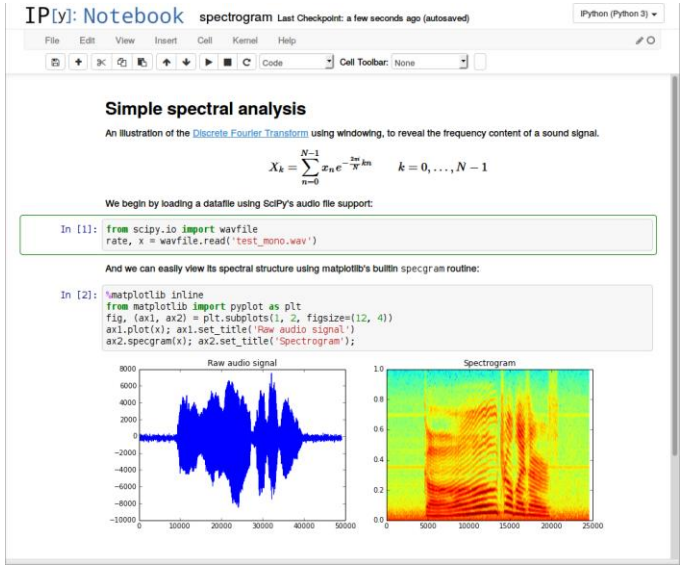
- Several ways to create a virtual environment
  - Using Anaconda (Conda) a Environment
    - https://conda.io/docs/user-guide/tasks/manage-environments.html
    - **conda create --name myenv**
    - **conda install numpy**
    - **conda install anaconda**
    - **conda activate myvenv**
    - **conda deactivate**

Jupyter Notebooks and Lab

Jupyter Notebooks

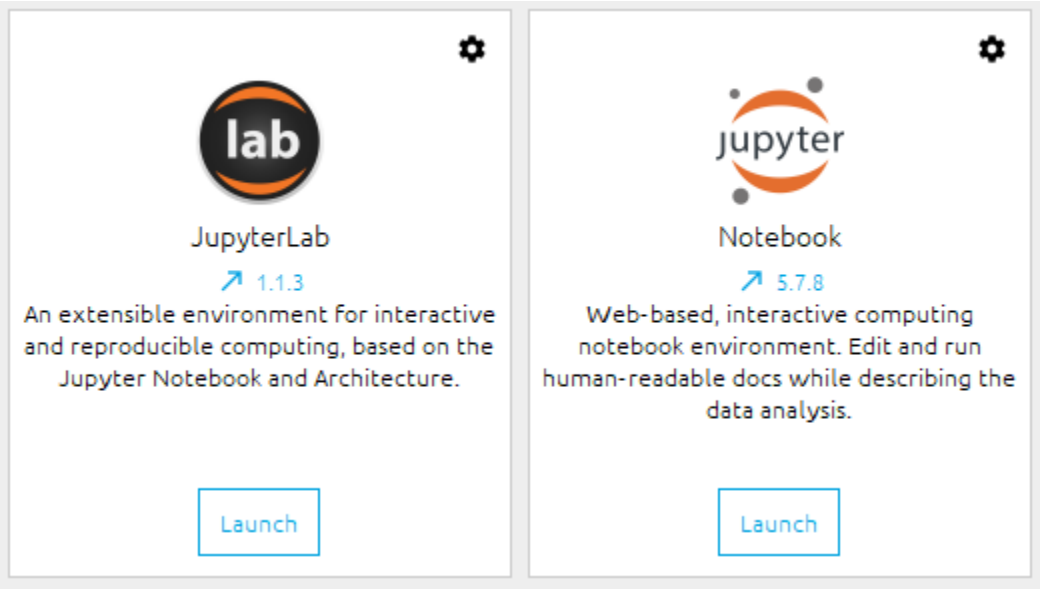
- Notebooks are an excellent way to document an engineering process because they allow you to combine nicely formatted text with code and results.
- Jupyter notebooks use a serialized data storage format, JSON, to store the document.
- Integrate many different text processing formats (HTML, Markdown, LaTex, …).
- Jupyter notebook use a browser-like interface and is integrated into Anaconda





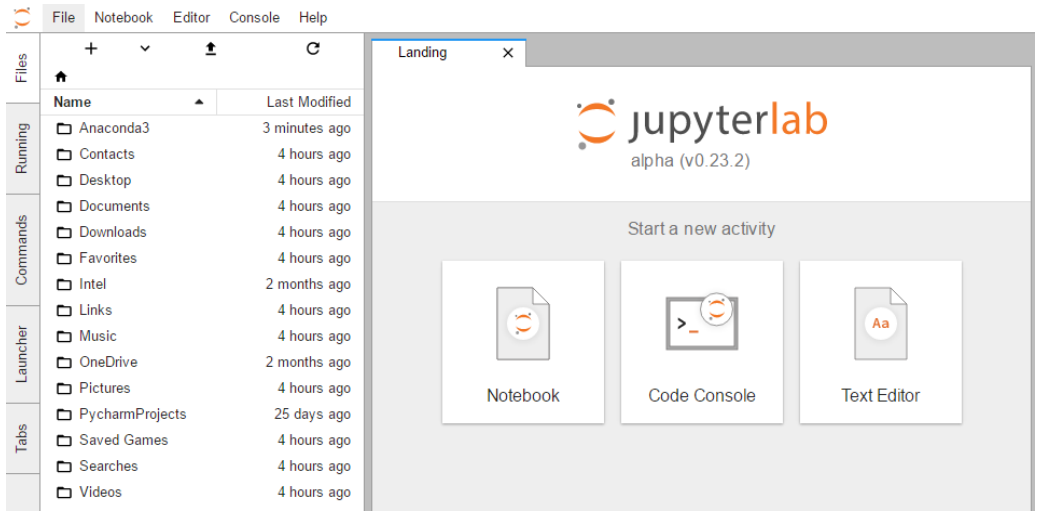
JupyterLab

- JupyterLab runs in a single tab, with sub-tabs displayed within that one tab, while Jupyter Notebook opens new notebooks in new tabs.
  - JupyterLab feels more like an IDE
  - Jupyter Notebook feels more standalone



JupyterLab

- Installation
  - Jupyter Notebook comes with Anaconda
  - **conda install -c conda-forge jupyterlab**
- Running
  - **jupyter-lab**
  - **jupyter notebook**



pip install jupyterlab

JupyterLab

- Jupyter cells
  - **Code**
  - **Markdown/Text**
- See Demo