

# Jupyter

- Notebooks
  - a vehicle to communicate your thoughts to others
  - a container for code, text, graphics
  - a Web-based IDE
- Jupyter is a notebook server
  - Server either local or in cloud
  - client usually on local machine
  - Multi-language
    - Jupyter is short for: **J**ulia, **P**ython, **R**

# Preliminaries

## Jupyter setup

Jupyter is part of the Anaconda distribution, which you already installed.

Let's finish setting up Jupyter by creating a directory for notebooks

```
mkdir Notebooks; cd Notebooks
```

And setting a password for the notebook server (optional on local machine; MANDATORY for cloud-based)

```
jupyter notebook --generate-config  
jupyter notebook password
```

## Start jupyter

*anaconda-navigator*

or

*cd Notebooks*

*jupyter notebook*

Jupyter runs in your browser.

If you installed it on your local machine, the URL is `localhost:8888`

If you installed it on a cloud machine, the URL is `your_server_ip:8888`

where `your_server_ip` is the IP address of your cloud based machine.

## Jupyter extensions

Jupyter has many useful extensions. It is NOT required for you to do this step but here are some extensions that I'm currently using

- Install
  - `conda install -c conda-forge jupyter_contrib_nbextensions`
- Enable

```
jupyter nbextension enable toc2/main
jupyter nbextension enable
collapsible_headings/main
jupyter nbextension enable
livemdpreview/livemdpreview
```

- Disable/Enable - <http://localhost:8888/nbextensions> or via tab on Jupyter Home page - check-box for which extensions to enable - You can disable/enable extensions any time

# Jupyter: a vehicle for communication (NOT just coding)

- Code and "mark-down"
- Lectures via Notebooks !

- Assignments
  - Your notebooks are your "lab notebook"
    - The final result is not always the most interesting part !
      - Process and what you learned on the journey is important
    - Define the problem you are working on
    - Describe and explore the data
      - what were the challenges ? Cleaning ? Transformation ?
    - Overview of your methodology/research method
    - Experiments conducted/results, both success and failure
    - Describe your steps in English, followed by code
- Code-only: limited credit !

**\*\*Tip\*\***: It's a movie not a photograph !



# Jupyter tour

- [Jupyter dashboard \(external/ipython-in-depth/examples/Notebook/Notebook%20Basics.ipynb#The-Dashboard\)](#)
- [Header and body \(external/ipython-in-depth/examples/Notebook/Notebook%20Basics.ipynb#The-Notebook\)](#)
- Command mode/edit mode
  - [Keyboard shortcuts \(external/ipython-in-depth/examples/Notebook/Notebook%20Basics.ipynb#Keyboard-Navigation\)](#)
- [Types of Cells \(external/mltutorial/notebooks/IPython-Tutorial/1%20%20Notebooks%20%26%20Cells.ipynb\)](#)
  - Cells can contain either code or markdown (e.g., text)
    - Code shows your solution
    - Markdown used to tell the story of your journey

## Jupyter markdown

- [Markdown \(external/mltutorial/notebooks/IPython-Tutorial/2%20-%20Markdown%20%26%20LATEX.ipynb\)](#)
  - [Markdown cheat sheet \(https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet\)](#)
  - [Equations, categorized \(http://www.equationsheet.com/\)](#)

# Introspection

- TAB completion
  - Data properties
- ?
  - Function help
- ??
  - Code inspection

[Sample notebook \(Sample.ipynb\).](#)

# Checkpoints

- Jupyter will save a snapshot ("checkpoint") each time you save your notebook
- Jupyter will auto-save your notebook as you change it
  - You can discard the auto-saved changes by reverting back to a checkpoint