Jupyter Notebooks

This is a quick introduction to Jupyter notebooks.

Jupyter notebooks are a way to combine executable code, code outputs, and text into one connected file.

The official documentation from project Jupyter is available **here (https://jupyter-notebook.readthedocs.io/en/stable/)** and they also have some example notebooks **here (https://github.com/jupyter/notebook/tree/master/docs/source/examples/Notebook)** .

Menu Options & Shortcuts

To get a quick tour of the Jupyter user-interface, click on the 'Help' menu, then click 'User Interface Tour'.

There are also a large number of useful keyboard shortcuts. Click on the 'Help' menu, and then 'Keyboard Shortcuts' to see a list.

Cells

The main organizational structure of the notebook are 'cells'.

Cells are an independent 'unit'. When you click into a cell, you can 'run' it by clicking Shift + Enter, or by pressing the play button above.

Cells come in differen types for writing different things - mainly, text or code.

Markdown Cells

Cells, can be markdown (text), like this one.

Code Cells

In [2]:
```
# Cell can also be code.
a = 1
b = 2
```

In [3]:
```
# Cells can also have output, that gets printed out below the cell.
c = a + b
print(c)
```

```
3
```

In [ ]:
```
# If you execute a cell with just a variable name in it, it will also get printed
c
```

Running Cells

- The numbers in the square brackets to the left of a cell show which cells have been run, and in what order.
    - An asterisk (*) means that the cell is currently running
    - You do not need to run cells in order! This is useful for flexibly testing and developing code.

Accessing Documentation

Jupyter has useful shortcuts. Add a single '?' after a function or class get a window with the documentation, or a double '??' to pull up the source code.

In [4]:
```
# For example, execute this cell to see the documentation for the 'abs'
abs?
```

Autocomplete

Jupyter also has **tab complete (https://en.wikipedia.org/wiki/Command-line_completion)** capacities, which can autocomplete what you are typing, and/or be used to explore what code is available.

In [ ]:
```
# Move your cursor to the end of the line, press tab, and a drop menu will appear
 showing all possible completions
ra
```

In [ ]:
```
# If there is only one option, tab-complete will auto-complete what you are typing
ran
```

Web Browser

Jupyter notebooks display in a web browser. They are not hosted on the web, everything is happening locally.

If you click on the url in the browser, you will notice it says 'localhost'. This means it is connected to something locally, on your computer.

That local connection is to the 'kernel'.

Kernels

The 'kernel' is the thing that executes your code. It is what connects the notebook (as you see it) with the part of your computer that runs code.

Your kernel also stores your namespace - all the variables and code that you have declared (executed).

It can be useful to clear and re-launch the kernel. You can do this from the 'kernel' drop down menu, at the top, optionally also clearing all ouputs. Note that this will erase any variables that are stored in memory.