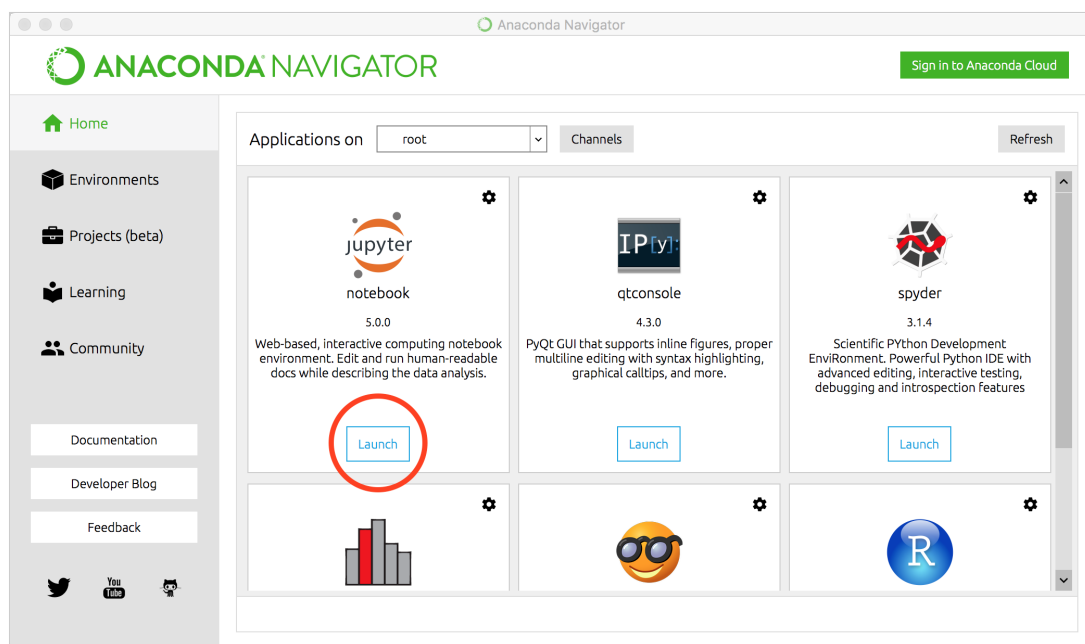


Get started with "Jupyter Notebook"

Hideki Kozima (xkozima@tohoku.ac.jp)

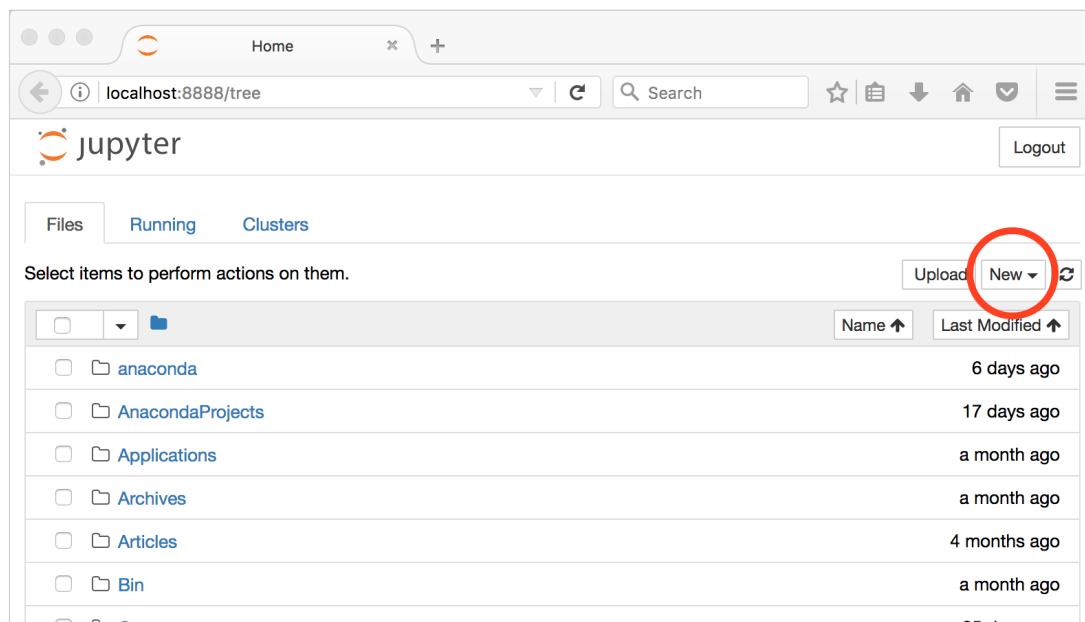
All you need is "Anaconda"

Download and install "**Anaconda**" from the official page of Anaconda (<https://www.continuum.io/>). You will get everything to run "Jupyter Notebook". Start "**Anaconda Navigator**" (from "start" menu on Windows, or LaunchPad on macOS), and you will see a collection of Anaconda applications you can choose from.

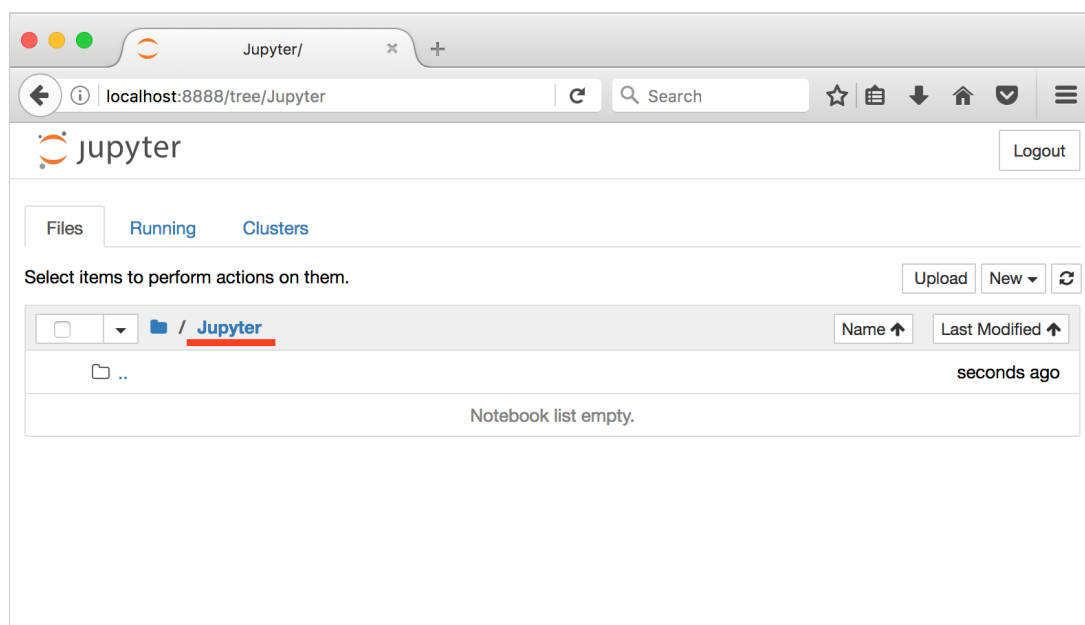


Launch "Jupyter Notebook"

Launch "**jupyter notebook**". It shows up a terminal window, on which the Python engine runs. Shortly it shows up a browser tab on your default Web browser. This browser tab "Home" is the portal page for interaction with your notebooks.

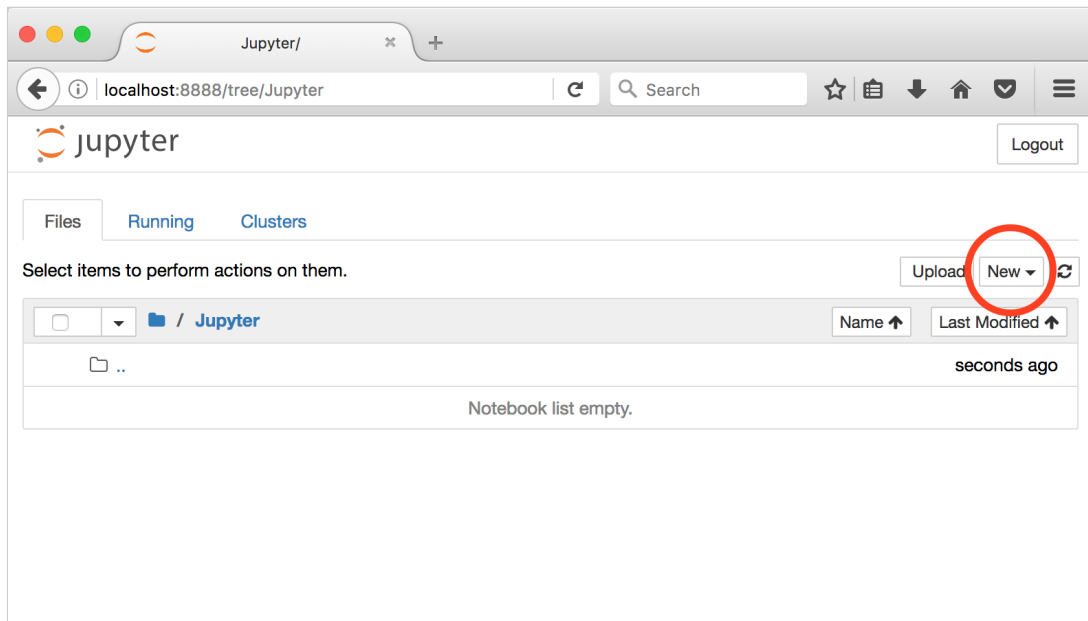


The browser tab shows the contents of your home folder. Let us make a "Jupyter" folder in your home folder, in which you can organize your jupyter notebooks. Click "**New**" on the right side of jupyter tab, and choose "**Folder**" from the menu. You will get a untitled folder. Then, check the left side of the folder name, and the click "**Rename**" button above. Let us give it a new name, say, "Jupyter". Let us move into the folder.

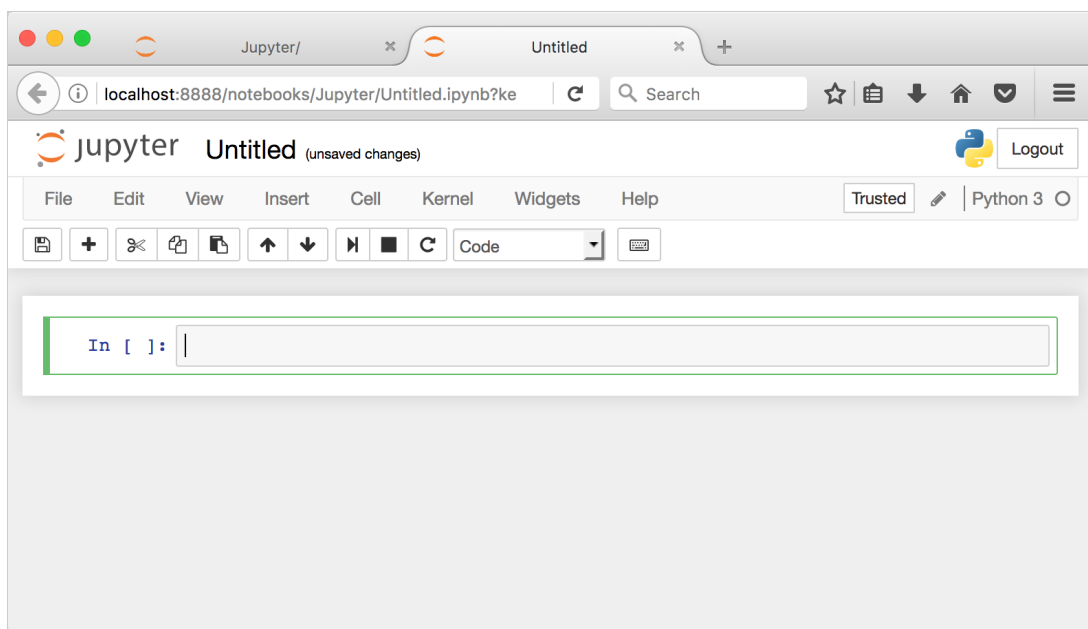


Make your own "notebook"

Now let us make a new jupyter notenook, on which we learn some basics of Python and data science. Click "**New**" on the right side of jupyter tab, and choose "**Python 3**" from the menu.

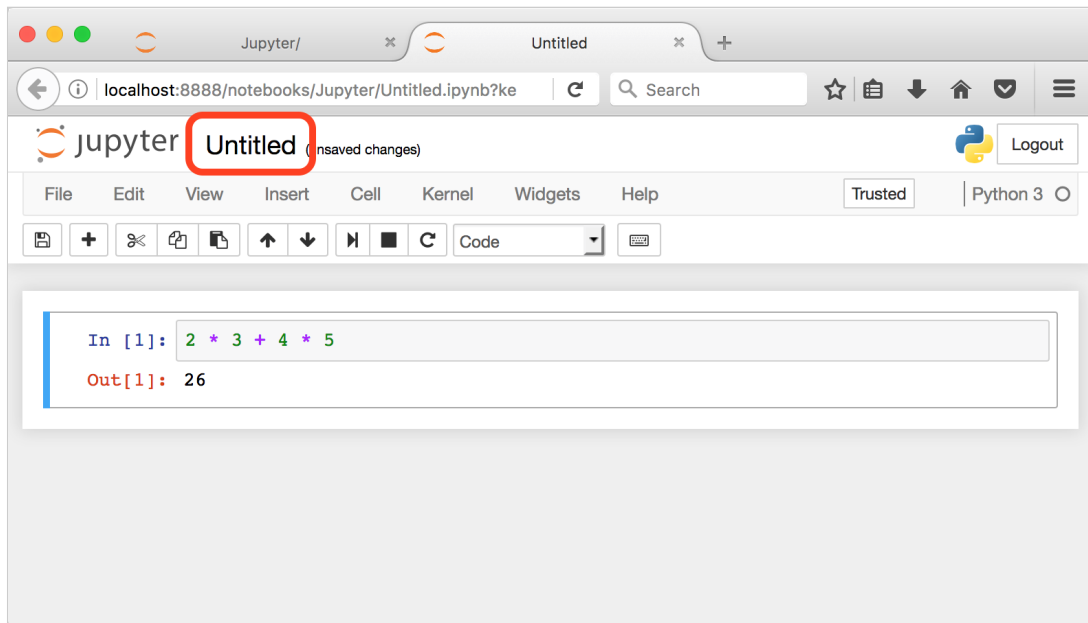


You will get an empty notebook in a new browser tab. This is where you interact with the **Python kernel**, that is an interpreter of the programming language.

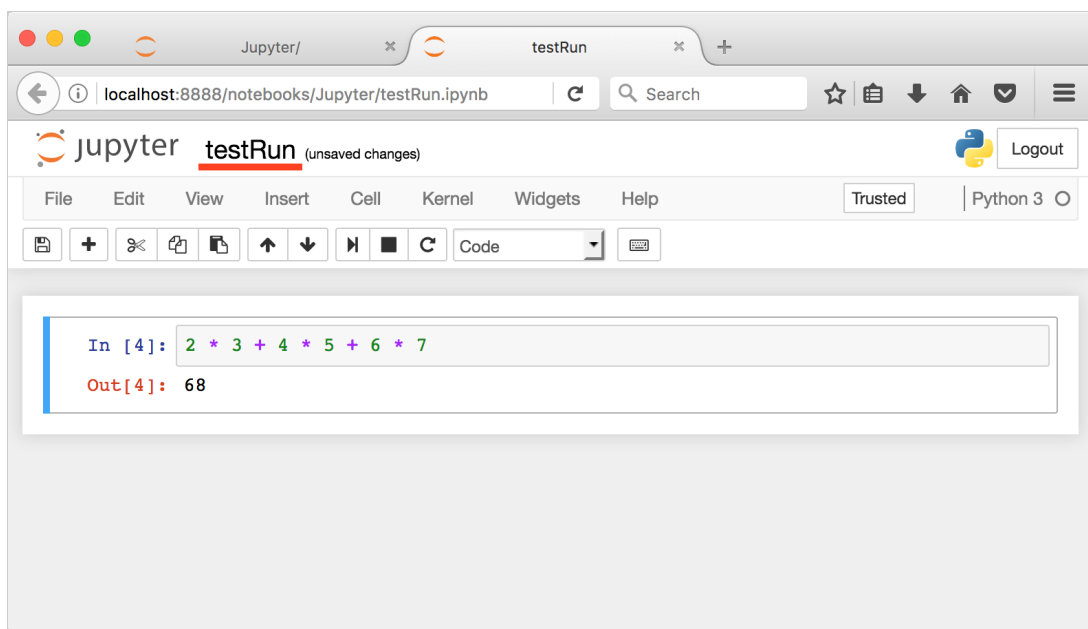


Type and run Python codes

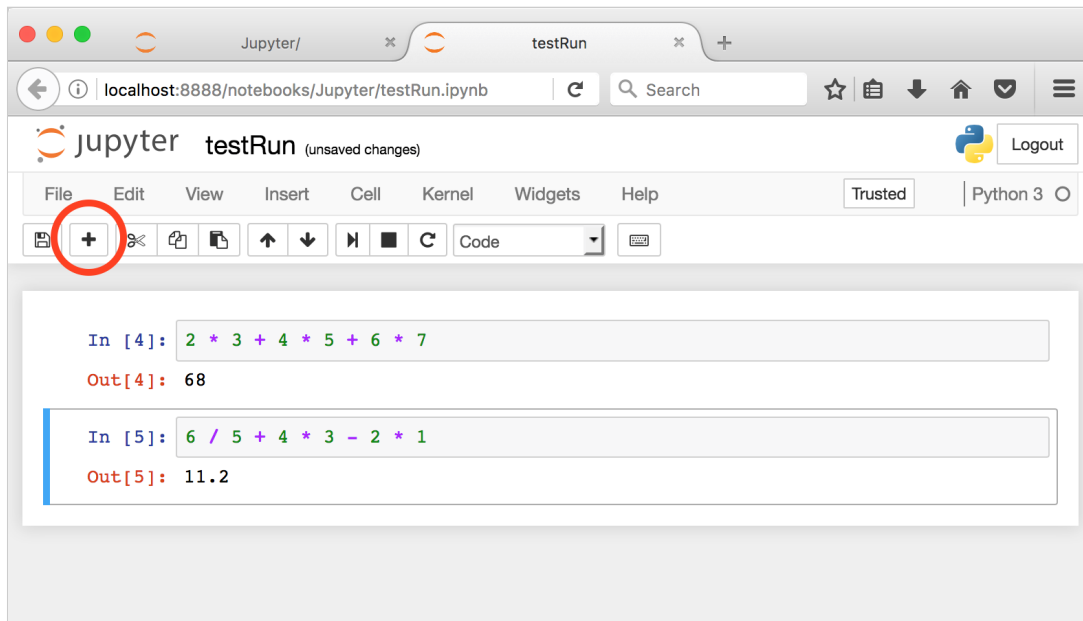
You can type any Python code in the input box. Press "**Control-Enter**" to run the code. You can modify the code and re-run it. Try it out!



You can save the notebook with an appropriate name. Click the current name "**Untitled**" above and **rename** it as, for example, "testRun".



If you want to make a new In-Out box, leaving the previous interaction behind, click "+" icon. You will get a new input box just below the previous one.

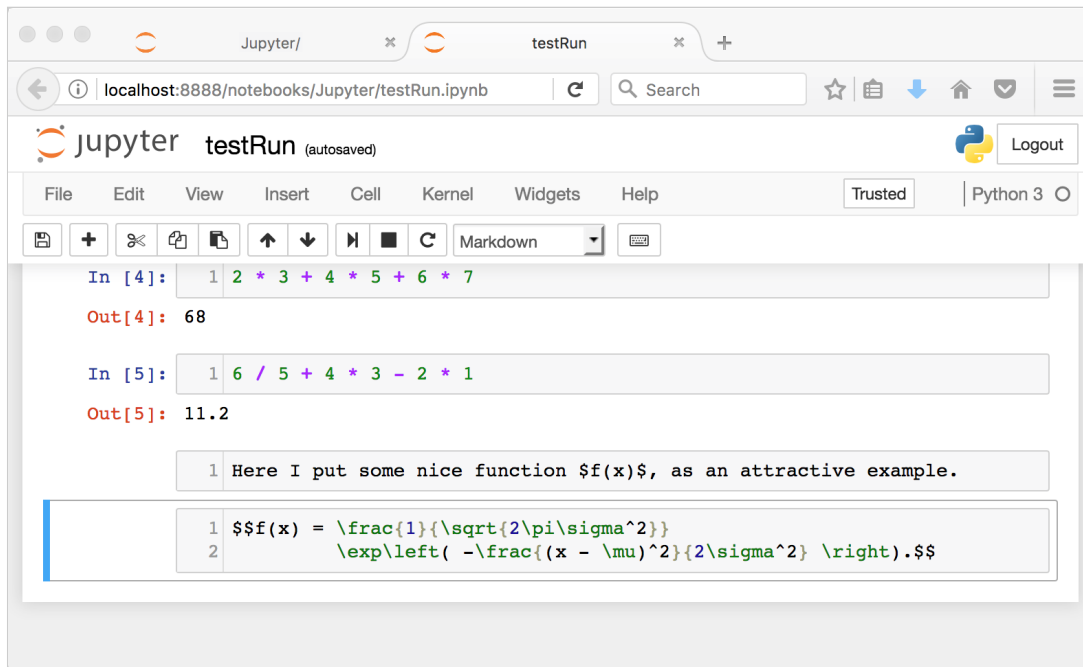


You can go back and forth to any In/Out box on your notebook. Just click one. You can modify it and re-run it. Copy-and-paste also works. Yes, this is your notebook for your learning and research.

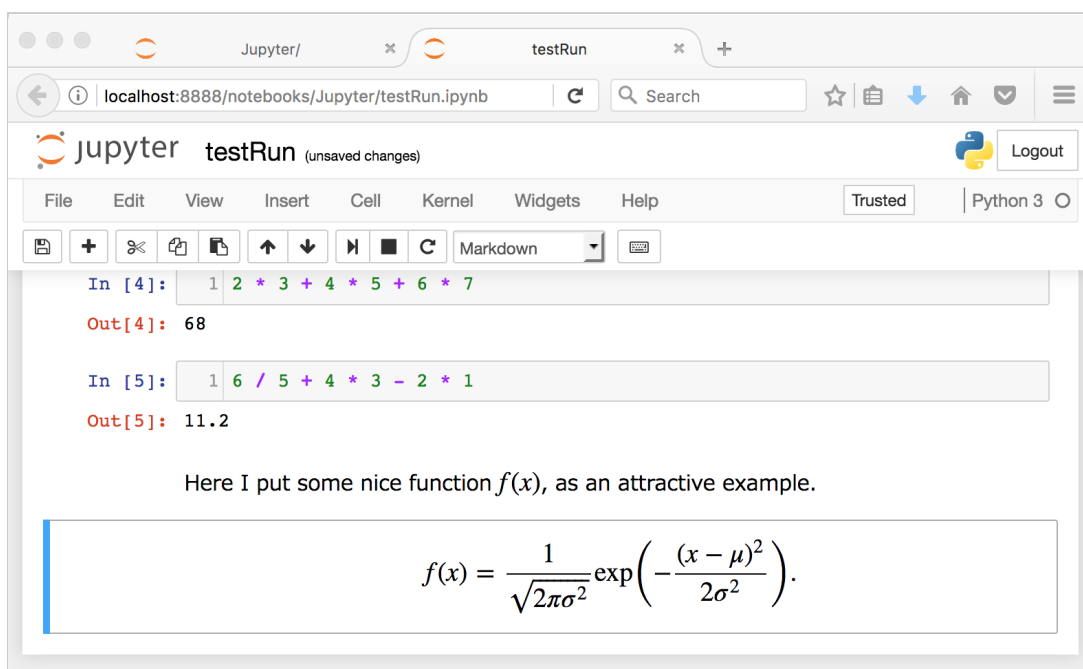
Note that "**division**" in Python 3 always results in **float** (real number). In the above example, the division between integers "6 / 5" gives "1.2" in a float. While, in other major programming languages (including Python 2), such a division between integers gives just an integer quotient.

Add text and math

If you want to leave some comments, say, just below an In/Out box, click the box and click "+" in the menu. You will get an empty "**Code**" box. Change it to "**Markdown**" from the pulldown menu. Now you can type in any **text**, even with **HTML** tags, or **math** expressions in **LaTeX** notation.

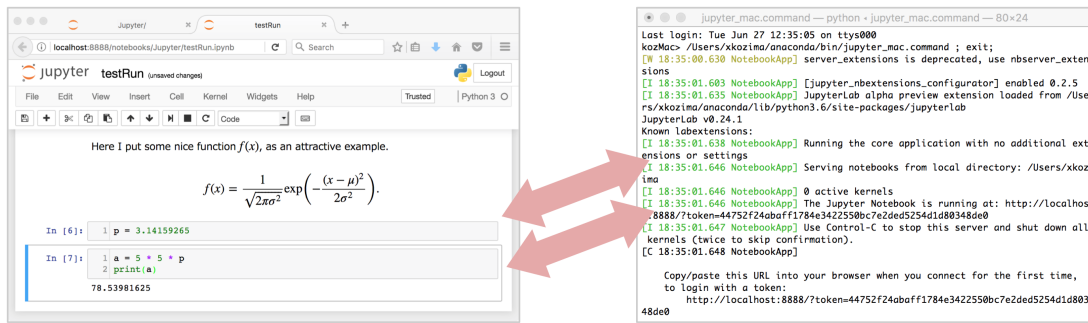


The "Markdown" texts above will turn into attractive text and mathematical expressions as shown below. If you want to know the details of LaTeX notation, refer to [WikiBooks on LaTeX/Mathematics](https://en.wikibooks.org/wiki/LaTeX/Mathematics) (<https://en.wikibooks.org/wiki/LaTeX/Mathematics>).

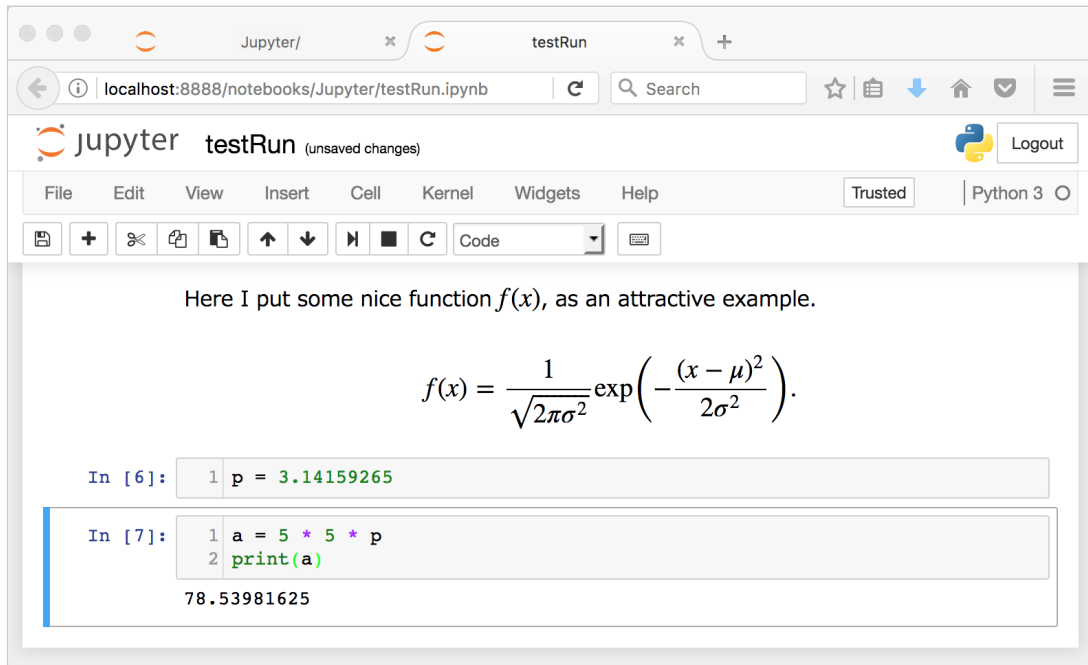


Under the hood

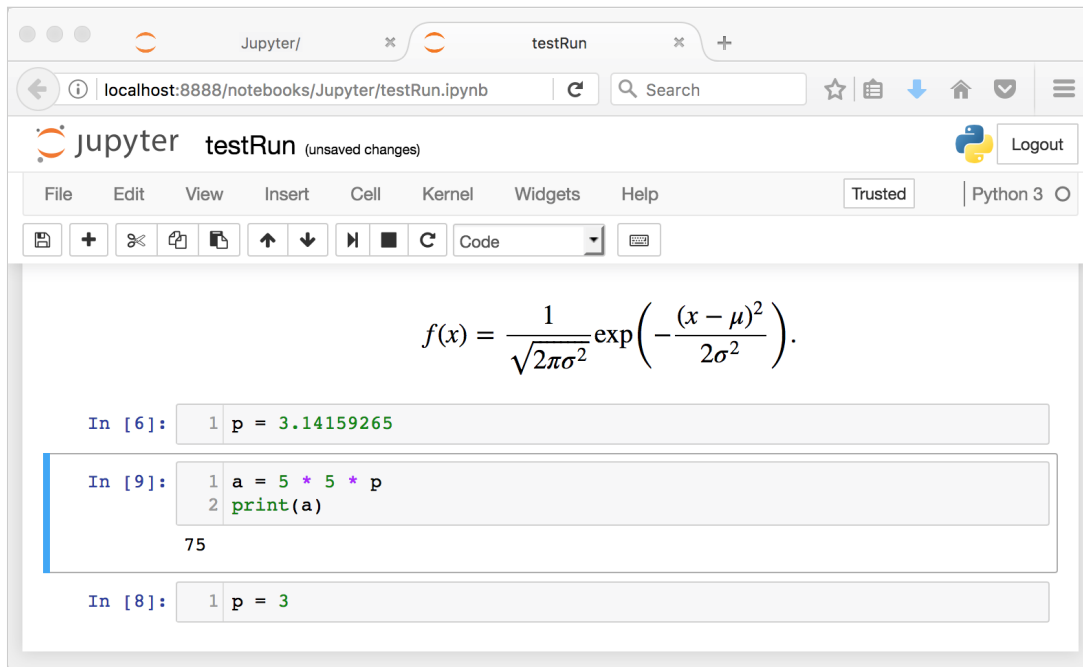
You may want to make a number of In/Out boxes on your notebook. Those boxes are connected to a **single Python kernel**, or a Python interpreter, under the engine hood. When you execute some code in a box, the resulting effect (for example, new values assigned to variables) will be transferred to another box you are interacting with.




Take a look at the following example. In the last box, the value of "p" is carried over from the assignment in the preceding box. Those boxes are connected to the same kernel.



Next example is a bit more interesting. After we re-assign "p" as 3 in the last box, we select the preceding box and re-execute the code in it. The result changed to 75 (in integer). What matters is **the order of execution**, not the order of placement in the notebook.



Get stuck?

If you think you get stuck, or you think your Python kernel does not respond properly, try restarting the kernel by clicking  icon.

Still cannot solve the problem? As the last resort, close all the browser tabs for Jupyter Notebooks, close the terminal where the Python kernel is running, and then start over from Anaconda Navigator. I hope you can come back to where you were.

Closing and re-opening your notebook

To close (shut down) your notebook, first **save** the current contents of the notebook by pressing "**Command-S**" or "**Control-S**", or clicking "save" icon, to preserve any progress you have made on your notebook. Then, **close** the browser tab, by pressing "**Command-W**" or "**Control-W**", or clicking the closing button on the browser tab.

If you want to stop the Python kernel as well, **close the terminal** behind that runs the kernel. You should close all the notebooks first, then kill the kernel.

To re-open your notebook, if the Python kernel is not running, **start "Anaconda Navigator"**, from which **launch "jupyter notebook"**. If the kernel is running but the "Home" tab is not present on your browser, open "**localhost:8888/tree**" in a new tab. If you see the "**Home**" tab, just click the notebook file to open.

Python for data science

Now, everything is ready to go. Let us start learning Python for data science. We are using "jupyter notebooks" for working on the following list of learning contents.

0. Get started with "Jupyter Notebook" (this material)
1. Learning Python in 90 minutes: [pyTutorial1.zip](http://www.ei.tohoku.ac.jp/xkozima/lab/deep/pyTutorial1.zip)
(<http://www.ei.tohoku.ac.jp/xkozima/lab/deep/pyTutorial1.zip>)
2. Introduction to data science with Python: pyTutorial2.zip
3. Introduction to deep learning: pyTutorial3.zip

To proceed to "**1. Learning Python in 90 minutes**", download "**pyTutorial1.zip**" from the link above and unzip it. You will get a folder "pyTutorial1". Place the folder in your "Jupyter" folder.

From the "Home" tab on your browser, open the file "**pyTutorial1.ipynb**" in the folder. You will have a notebook opened in your browser. That is your notebook as well as the textbook for learning Python for data science. From now on, please follow the instruction in the notebook.

(cc) Koziken, MMXVII