03-JupyterQuickStart

August 11, 2020

1 Jupyter

1.1 Launch Jupyter server

`jupyter notebook`

- Go to notebooks folder
- Open the file 03.JupyterQuickStart.ipynb

1.2 Make a Copy

Before modifying the notebook, make a copy of it. Go to to `File` menu on the top left of the notebook and click on `Make a Copy...`

1.3 Jupyter Notebook

Jupyter notebook, formerly known as the IPython notebook, is a flexible tool that helps you create readable analyses, as you can keep code, images, comments, formulae and plots together.

Jupyter is quite extensible, supports many programming languages and is easily hosted on your computer or on almost any server — you only need to have ssh or http access. Best of all, it's completely free.

The name Jupyter is an indirect acronyum of the three core languages it was designed for: **JU**lia, **PYT**hon, and **R**

1.4 Keyboard Shortcuts

- To access keyboard shortcuts, use the command palette: `Cmd + Shift + P`
- `Esc` will take you into command mode where you can navigate around your notebook with arrow keys.
- While in command mode:
- A to insert a new cell above the current cell, B to insert a new cell below.
- M to change the current cell to Markdown, Y to change it back to code
- D + D (press the key twice) to delete the current cell

1.5 Easy links to documentation

- Shift + Tab will also show you the Docstring

[1]:
```
dict
```

[1]:
```
dict
```

1.6 Magic commands

[2]:
```
%lsmagic
```

[2]:
```
Available line magics:
%alias  %alias_magic  %autoawait  %autocall  %automagic  %autosave  %bookmark
%cat  %cd  %clear  %colors  %conda  %config  %connect_info  %cp  %debug  %dhist
%dirs  %doctest_mode  %ed  %edit  %env  %gui  %hist  %history  %killbgscripts
%ldir  %less  %lf  %lk  %ll  %load  %load_ext  %loadpy  %logoff  %logon
%logstart  %logstate  %logstop  %ls  %lsmagic  %lx  %macro  %magic  %man
%matplotlib  %mkdir  %more  %mv  %notebook  %page  %pastebin  %pdb  %pdef  %pdoc
%pfile  %pinfo  %pinfo2  %pip  %popd  %pprint  %precision  %prun  %psearch
%psource  %pushd  %pwd  %pycat  %pylab  %qtconsole  %quickref  %recall  %rehashx
%reload_ext  %rep  %rerun  %reset  %reset_selective  %rm  %rmdir  %run  %save
%sc  %set_env  %store  %sx  %system  %tb  %time  %timeit  %unalias  %unload_ext
%who  %who_ls  %whos  %xdel  %xmode

Available cell magics:
%%!  %%HTML  %%SVG  %%bash  %%capture  %%debug  %%file  %%html  %%javascript
%%js  %%latex  %%markdown  %%perl  %%prun  %%pypy  %%python  %%python2
%%python3  %%ruby  %%script  %%sh  %%svg  %%sx  %%system  %%time  %%timeit
```

```
%%writefile

Automagic is ON, % prefix IS NOT needed for line magics.
```

[3]:
```
%ls
```

```
01-GitBasics.ipynb                book-017.dat  book-054.dat  book-091.dat
02-Installation.ipynb             book-018.dat  book-055.dat  book-092.dat
03-JupyterQuickStart.ipynb        book-019.dat  book-056.dat  book-093.dat
04-WordCount.ipynb                book-020.dat  book-057.dat  book-094.dat
05-MapReduce.ipynb                book-021.dat  book-058.dat  book-095.dat
06-ParallelComputation.ipynb      book-022.dat  book-059.dat  book-096.dat
07-AsynchronousProcessing.ipynb   book-023.dat  book-060.dat  book-097.dat
08-DaskDelayed.ipynb              book-024.dat  book-061.dat  book-098.dat
09-DaskBag.ipynb                  book-025.dat  book-062.dat  book-099.dat
10-PandasSeries.ipynb             book-026.dat  book-063.dat
data/
11-PandaDataframes.ipynb          book-027.dat  book-064.dat  fibonacci.py
12-PySpark.ipynb                  book-028.dat  book-065.dat  images/
13-UnixCommands.ipynb             book-029.dat  book-066.dat  log
14-Hadoop.ipynb                   book-030.dat  book-067.dat
mapper.py*
15-HadoopFileFormats.ipynb        book-031.dat  book-068.dat  pmap.py
16-DaskDataframes.ipynb           book-032.dat  book-069.dat
reducer.py*
17-SparkDataFrames.ipynb          book-033.dat  book-070.dat  sample00.txt
18-NYCTaxiCabTripDask.ipynb       book-034.dat  book-071.dat  sample01.txt
19-NYCTaxiCabTripSpark.ipynb      book-035.dat  book-072.dat  sample02.txt
Makefile                          book-036.dat  book-073.dat  sample03.txt
book-000.dat                      book-037.dat  book-074.dat  sample04.txt
book-001.dat                      book-038.dat  book-075.dat  sample05.txt
book-002.dat                      book-039.dat  book-076.dat  sample06.txt
book-003.dat                      book-040.dat  book-077.dat  sample07.txt
book-004.dat                      book-041.dat  book-078.dat  sample08.txt
book-005.dat                      book-042.dat  book-079.dat  sample09.txt
book-006.dat                      book-043.dat  book-080.dat  sample10.txt
book-007.dat                      book-044.dat  book-081.dat  sample11.txt
book-008.dat                      book-045.dat  book-082.dat  sample12.txt
book-009.dat                      book-046.dat  book-083.dat  sample13.txt
book-010.dat                      book-047.dat  book-084.dat  sample14.txt
book-011.dat                      book-048.dat  book-085.dat  sample15.txt
book-012.dat                      book-049.dat  book-086.dat  sample16.txt
book-013.dat                      book-050.dat  book-087.dat  sample17.txt
book-014.dat                      book-051.dat  book-088.dat  sample18.txt
book-015.dat                      book-052.dat  book-089.dat  sample19.txt
book-016.dat                      book-053.dat  book-090.dat  test_stdin.py
```

[4]:
```
%%file sample.txt

write the cell content to the file sample.txt.
The file is created when you run this cell.
```

```
Writing sample.txt
```

[5]:
```
%cat sample.txt
```

```
write the cell content to the file sample.txt.
The file is created when you run this cell.
```

[6]:
```
%%file fibonacci.py

f1, f2 = 1, 1
for n in range(10):
    print(f1, end=',')
    f1, f2 = f2, f1+f2
```

```
Overwriting fibonacci.py
```

[7]:
```
%run fibonacci.py
```

```
1,1,2,3,5,8,13,21,34,55,
```

[8]:
```
# %load fibonacci.py

f1, f2 = 1, 1
for n in range(10):
    print(f1, end=',')
    f1, f2 = f2, f1+f2
```

```
1,1,2,3,5,8,13,21,34,55,
```

[9]:
```
%%time
f1, f2 = 1, 1
for n in range(10):
    print(f1, end=',')
    f1, f2 = f2, f1+f2
print()
```

```
1,1,2,3,5,8,13,21,34,55,
CPU times: user 773 µs, sys: 161 µs, total: 934 µs
Wall time: 769 µs
```

1.7 Installing Python Packages from a Jupyter Notebook

1.7.1 Install a conda package in the current Jupyter kernel

Example with package `numpy` from *conda-forge*

```
%conda install -c conda-forge lorem
```