

# Kernels

And customizing Jupyter

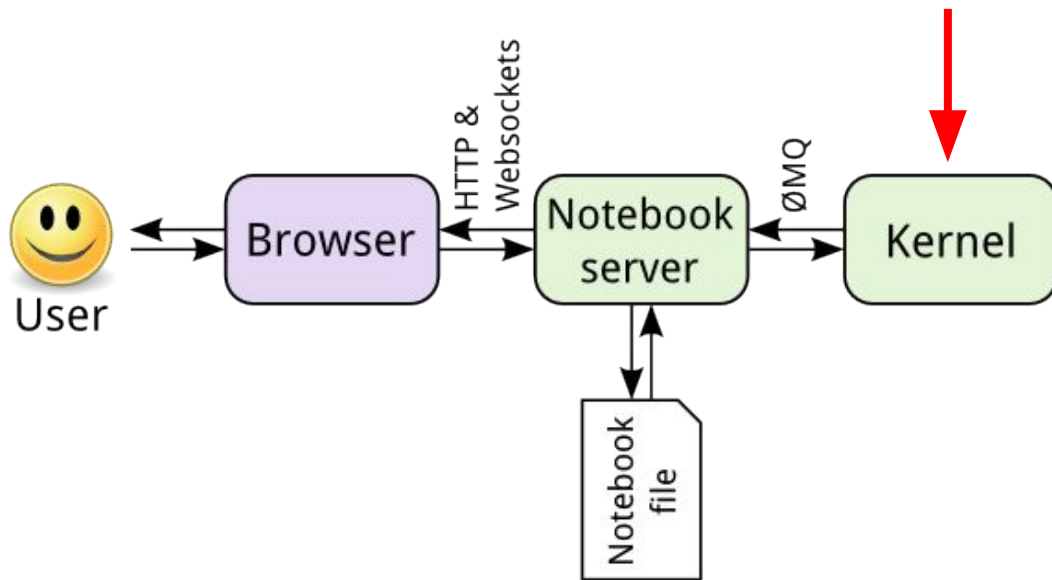
# What is the kernel

A **kernel** provides programming language support in Jupyter. IPython is the default kernel. Additional kernels include R, Julia, and many more.

Fun fact: Jupyter was named after **JULia** **PY**Thon and **R** once the old IPython notebook started supporting other languages

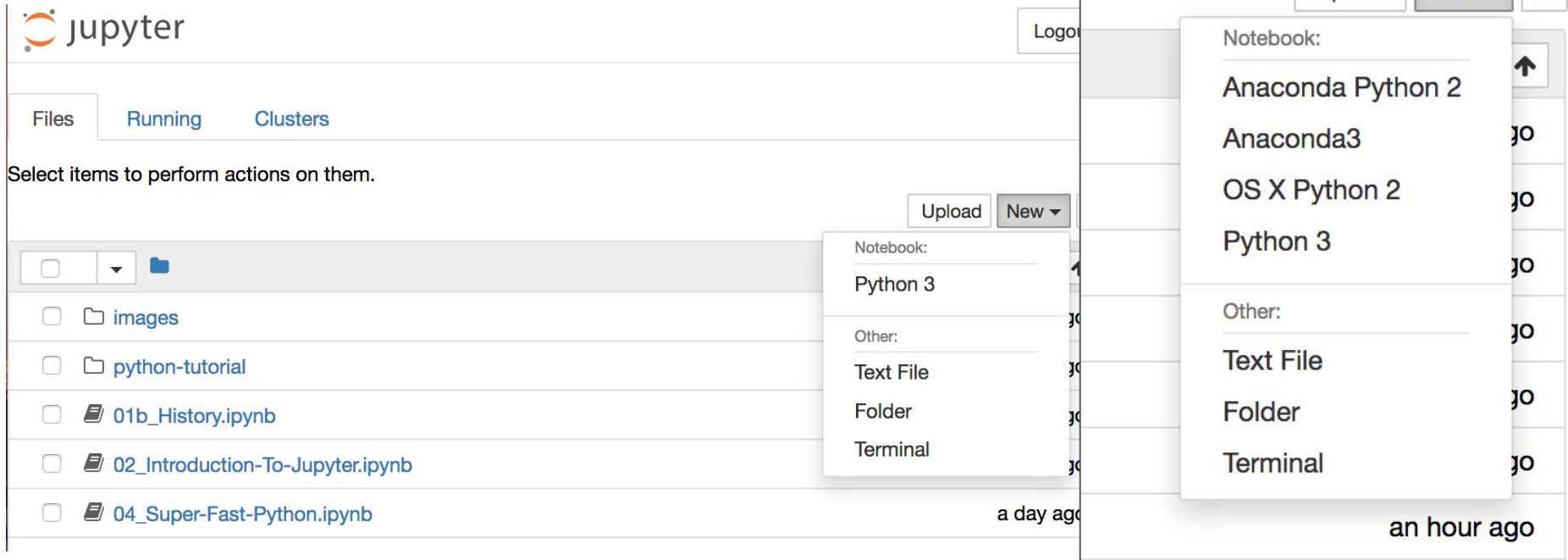


# Under the Covers



# Selecting a Kernel

In the Jupyter File Browser ...



The screenshot displays the Jupyter File Browser interface. At the top left is the Jupyter logo. Below it are tabs for 'Files', 'Running', and 'Clusters'. A message states 'Select items to perform actions on them.' Below this is a list of files and folders: 'images', 'python-tutorial', '01b\_History.ipynb', '02\_Introduction-To-Jupyter.ipynb', and '04\_Super-Fast-Python.ipynb'. On the right side, there are 'Upload' and 'New' buttons. The 'New' button is open, showing a dropdown menu. The menu is divided into two sections: 'Notebook:' and 'Other:'. The 'Notebook:' section lists 'Anaconda Python 2', 'Anaconda3', 'OS X Python 2', and 'Python 3'. The 'Other:' section lists 'Text File', 'Folder', and 'Terminal'. The 'New' button is also open in the top right corner of the interface, showing the same dropdown menu.

Jupyter

Files Running Clusters

Select items to perform actions on them.

Upload New

Notebook:

- Anaconda Python 2
- Anaconda3
- OS X Python 2
- Python 3

Other:

- Text File
- Folder
- Terminal

Upload New

Notebook:

- Anaconda Python 2
- Anaconda3
- OS X Python 2
- Python 3

Other:

- Text File
- Folder
- Terminal

images

python-tutorial

01b\_History.ipynb

02\_Introduction-To-Jupyter.ipynb

04\_Super-Fast-Python.ipynb

a day ago

an hour ago

# Adding a New Kernel

Instructions will vary from each language but you basically install a package and write a `kernel.json` file

# Python 2

[https://ipython.readthedocs.io/en/latest/install/kernel\\_install.html](https://ipython.readthedocs.io/en/latest/install/kernel_install.html)

```
> conda create -n ipykernel_py2 python=2 ipykernel  
> source activate ipykernel_py2 # Windows: remove 'source'  
> python -m ipykernel install --user
```

Now restart jupyter and open up notebook 10b\_Kernels-Py27.ipynb

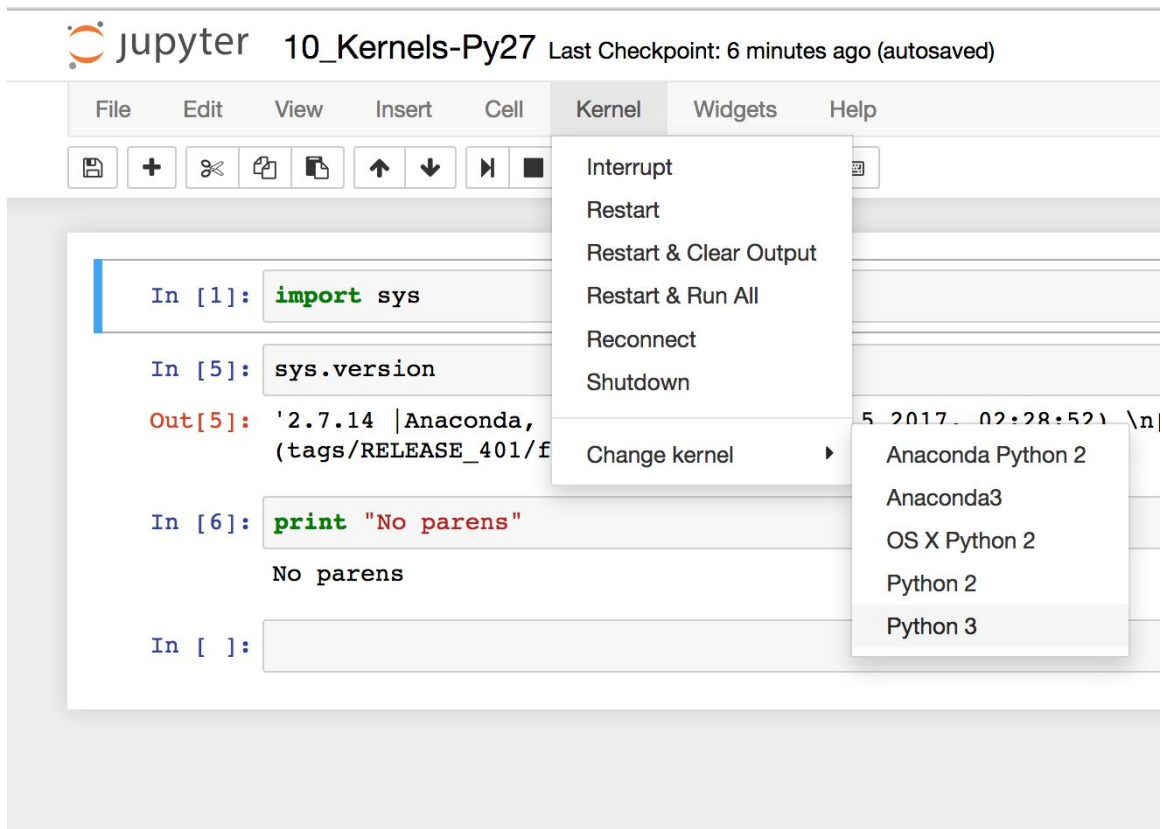
# R

<https://www.anaconda.com/developer-blog/jupyter-and-conda-r/>

```
> conda install -c rdonnellyr r-essentials
```

Try the 10c\_R\_demo.ipynb notebook

# Switching Kernels



The screenshot shows the JupyterLab interface for a notebook named '10\_Kernels-Py27'. The top bar indicates the last checkpoint was 6 minutes ago (autosaved). The 'Kernel' menu is open, showing options: Interrupt, Restart, Restart & Clear Output, Restart & Run All, Reconnect, Shutdown, and Change kernel. The 'Change kernel' option is selected, opening a sub-menu with the following options: Anaconda Python 2, Anaconda3, OS X Python 2, Python 2, and Python 3. The notebook content shows the following code cells:

```
In [1]: import sys
```

```
In [5]: sys.version
```

```
Out[5]: '2.7.14 |Anaconda,
(tags/RELEASE_401/f
```

```
In [6]: print "No parens"
```

```
No parens
```

```
In [ ]:
```

Re-run the notebook  
with a Python 3  
kernel



# Where do Kernels Live?

Look for a `kernel.json` file in

## UNIX

`~/Library/Jupyter/kernels` (Mac OS)

`~/.local/share/jupyter/kernels` (Linux)

`{sys.prefix}/share/jupyter/kernels`

`/usr/share/jupyter/kernels`

`/usr/local/share/jupyter/kernels`

## WINDOWS

`%APPDATA%\jupyter\kernels`

`%PROGRAMDATA%\jupyter\kernels`

# Customizing the Kernel Spec (kernel.json)

```
{
  "argv": [
    "/Users/shreyas/anaconda2/bin/python",
    "-m",
    "IPython.kernel",
    "-f",
    "{connection_file}"
  ],
  "env": {
    "PATH": "/Users/shreyas/anaconda2/bin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin",
    "PYTHONPATH": "/my/python/libs",
    "LD_LIBRARY_PATH": "/usr/local/lib"
  },
  "display_name": "Custom Python",
  "language": "python"
}
```

# Jupyter Settings

For command line options see

```
> jupyter notebook --help
```

Options can also be set by creating a file named `jupyter_notebook_config.py` in your Jupyter folder (`$HOME/.jupyter`).

To create a `jupyter_notebook_config.py` file, with all the defaults commented out, you can use the following command line:

```
> jupyter notebook --generate-config
```



# Let's change the Jupyter Default Directory

Edit `.jupyter/jupyter_notebook_config.py`

Change

```
#c.NotebookApp.notebook_dir = ''
```

to

```
c.NotebookApp.notebook_dir = '/'
```

Restart the notebook and notice where you are in the file browser!