Introduction to Jupyter Notebook

Ward Fisher

Unidata Python Workshop

July 2015

Overview

- Brief overview of Jupyter Notebook
  - What is it?
  - How do you use it?

Overview

- Brief overview of Jupyter Notebook
  - What is it?
  - How do you use it?
- Examples of basic Jupyter Notebook Usage.

Overview

- Brief overview of Jupyter Notebook
  - What is it?
  - How do you use it?
- Examples of basic Jupyter Notebook Usage.
- Discuss some Advanced Jupyter Notebook Uses.

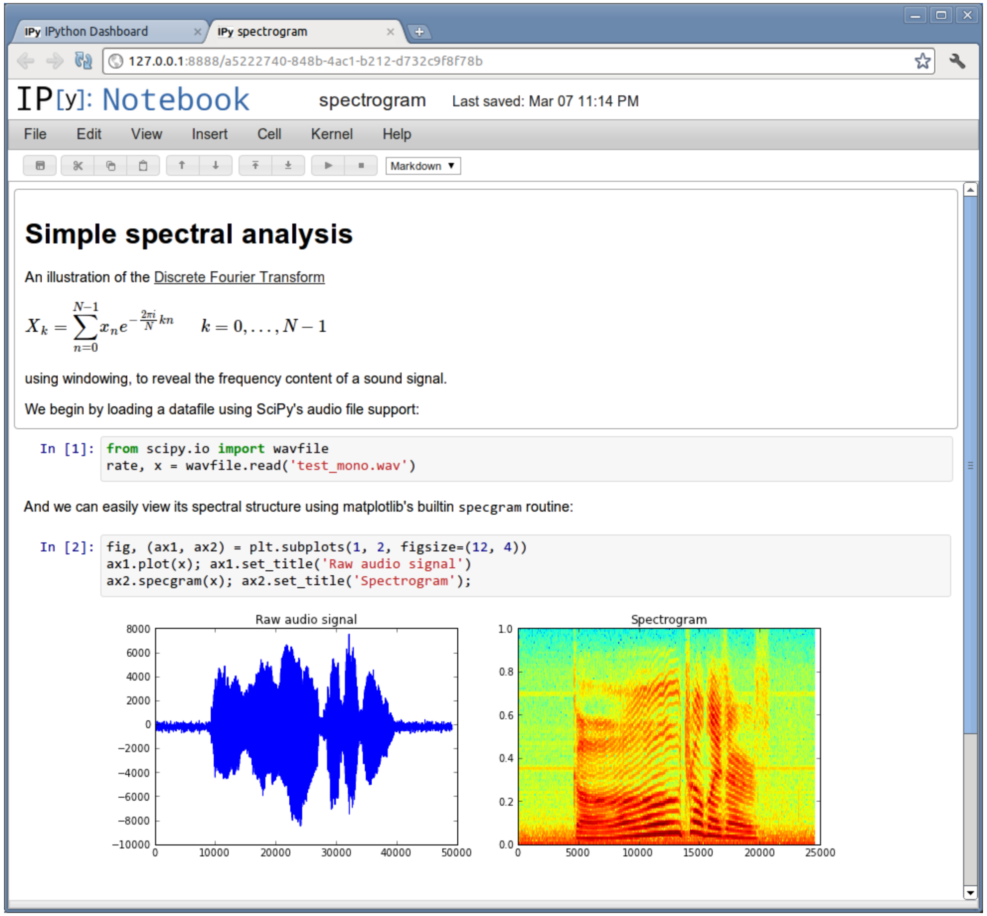
Overview

- *We* will not be going too in-depth with what you can do in regards to using Jupyter Notebooks for actual science.
- You will be seeing that throughout the rest of the workshop.

Remember IPython Notebook?

“The IPython Notebook is a web-based interactive computational environment where you can combine code execution, text, mathematics, plots and rich media into a single document”

http://ipython.org/notebook.html



Simple spectral analysis

An illustration of the Discrete Fourier Transform

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \qquad k = 0, \dots, N-1$$

using windowing, to reveal the frequency content of a sound signal.

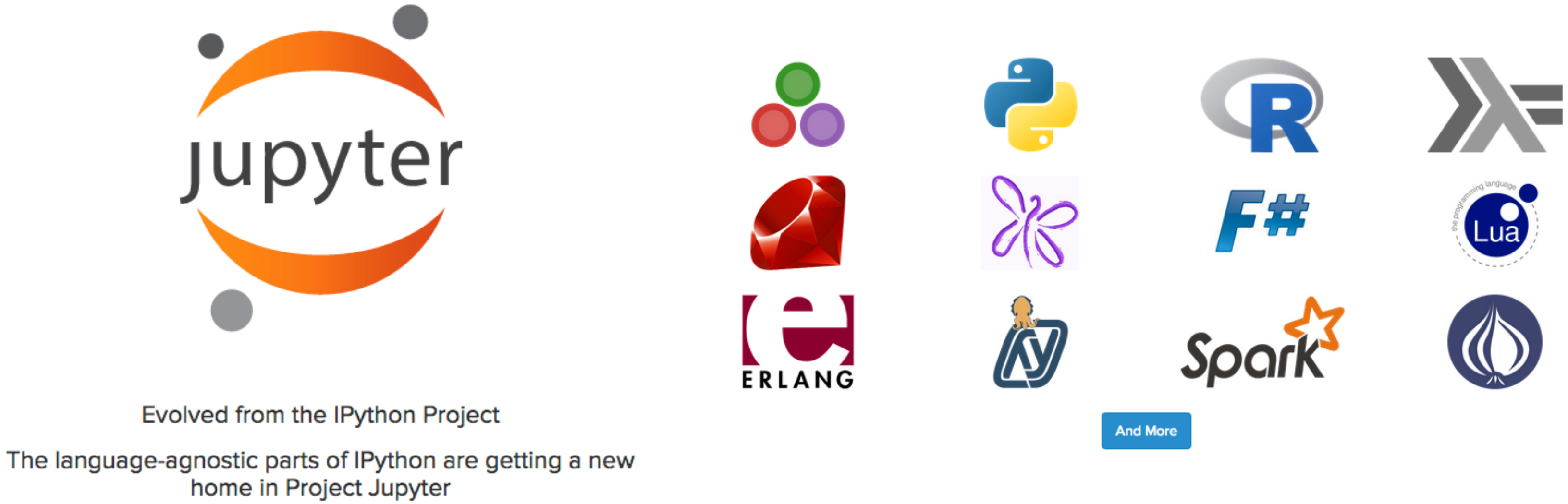
We begin by loading a datafile using SciPy's audio file support:

```
In [1]: from scipy.io import wavfile
        rate, x = wavfile.read('test_mono.wav')
```

And we can easily view its spectral structure using matplotlib's builtin `specgram` routine:

```
In [2]: fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))
        ax1.plot(x); ax1.set_title('Raw audio signal')
        ax2.specgram(x); ax2.set_title('Spectrogram');
```

What is Jupyter Notebook?



Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter

What is Jupyter Notebook?

- Jupyter Notebooks has two components:
  - The Jupyter Notebook server
  - Individual Notebooks (.ipynb)

Evolved from the IPython Project

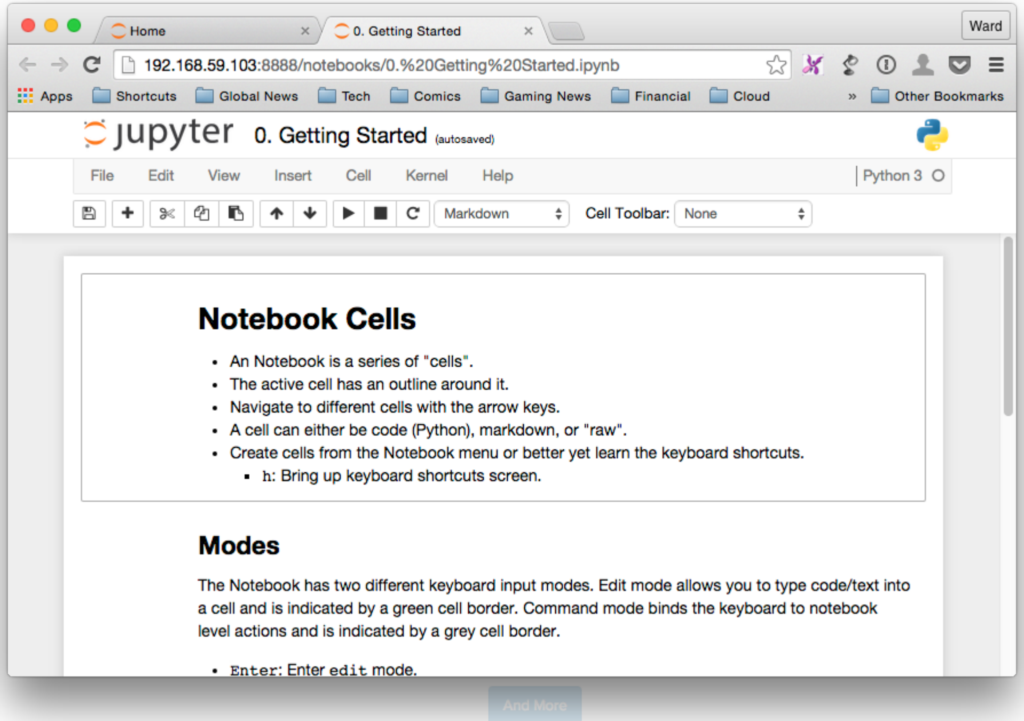
The language-agnostic parts of IPython are getting a new home in Project Jupyter

Jupyter Notebooks

- An Jupyter Notebook is a collection of *cells.*

Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter



Notebook Cells

- An Notebook is a series of "cells".
- The active cell has an outline around it.
- Navigate to different cells with the arrow keys.
- A cell can either be code (Python), markdown, or "raw".
- Create cells from the Notebook menu or better yet learn the keyboard shortcuts.
    - `h`: Bring up keyboard shortcuts screen.

Modes

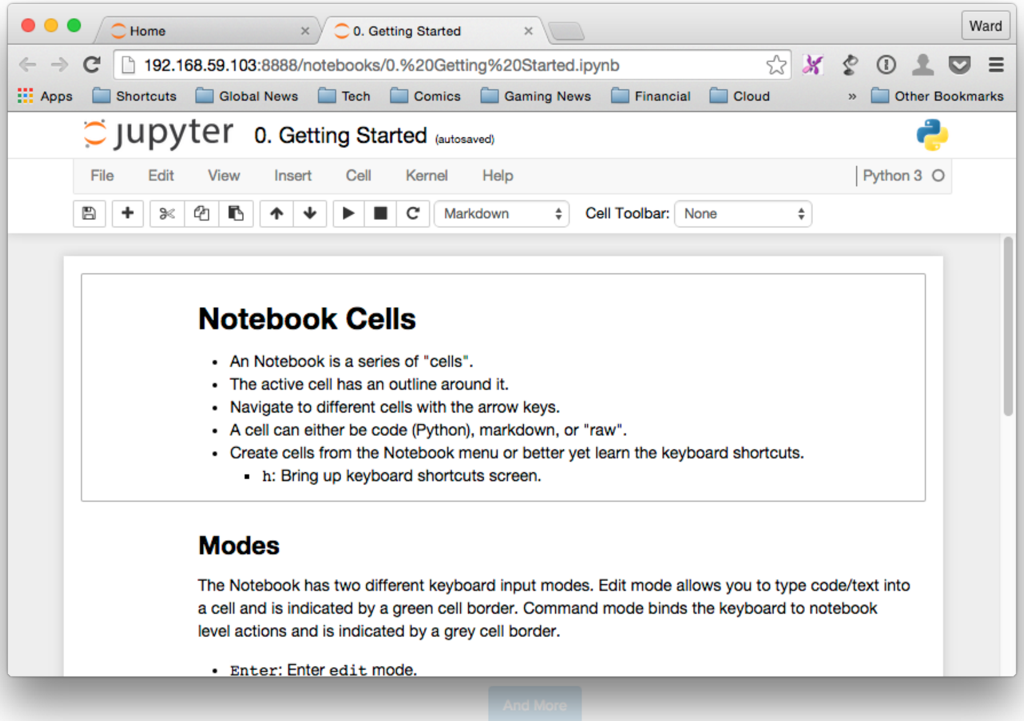
The Notebook has two different keyboard input modes. Edit mode allows you to type code/text into a cell and is indicated by a green cell border. Command mode binds the keyboard to notebook level actions and is indicated by a grey cell border.

- `Enter`: Enter `edit` mode.

And More

Jupyter Notebooks

- An Jupyter Notebook is a collection of *cells*.
  - Markdown



Notebook Cells

- An Notebook is a series of "cells".
- The active cell has an outline around it.
- Navigate to different cells with the arrow keys.
- A cell can either be code (Python), markdown, or "raw".
- Create cells from the Notebook menu or better yet learn the keyboard shortcuts.
  - `h`: Bring up keyboard shortcuts screen.

Modes

The Notebook has two different keyboard input modes. Edit mode allows you to type code/text into a cell and is indicated by a green cell border. Command mode binds the keyboard to notebook level actions and is indicated by a grey cell border.

- `Enter`: Enter `edit` mode.

Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter

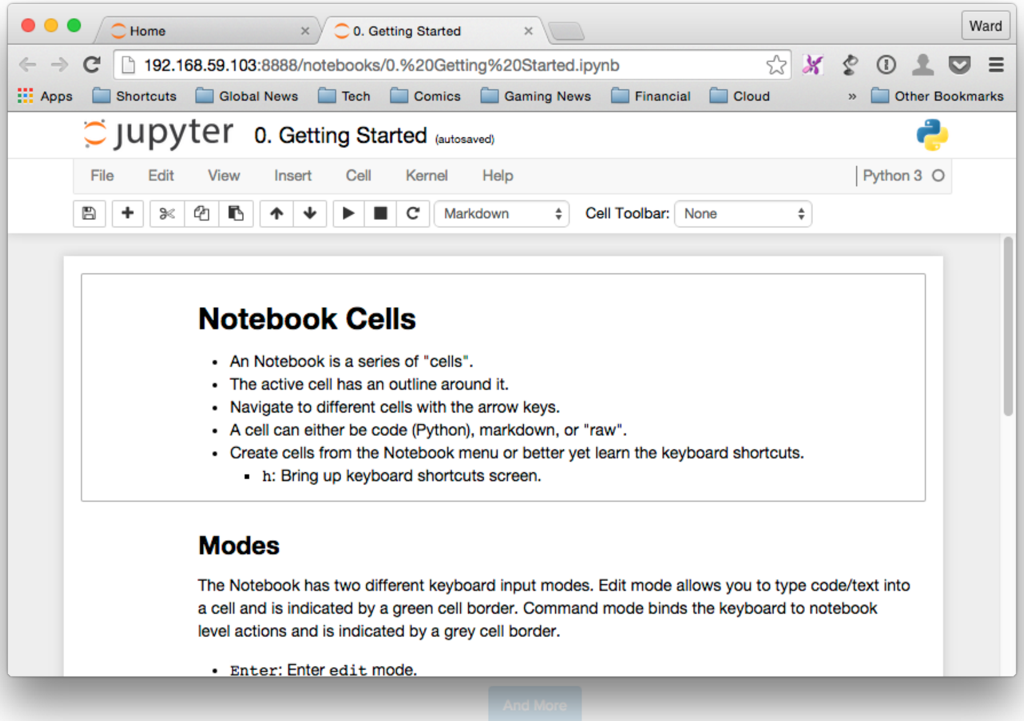
And More

Jupyter Notebooks

- An Jupyter Notebook is a collection of *cells*.
  - Markdown
  - Code

Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter



Notebook Cells

- An Notebook is a series of "cells".
- The active cell has an outline around it.
- Navigate to different cells with the arrow keys.
- A cell can either be code (Python), markdown, or "raw".
- Create cells from the Notebook menu or better yet learn the keyboard shortcuts.
  - `h`: Bring up keyboard shortcuts screen.

Modes

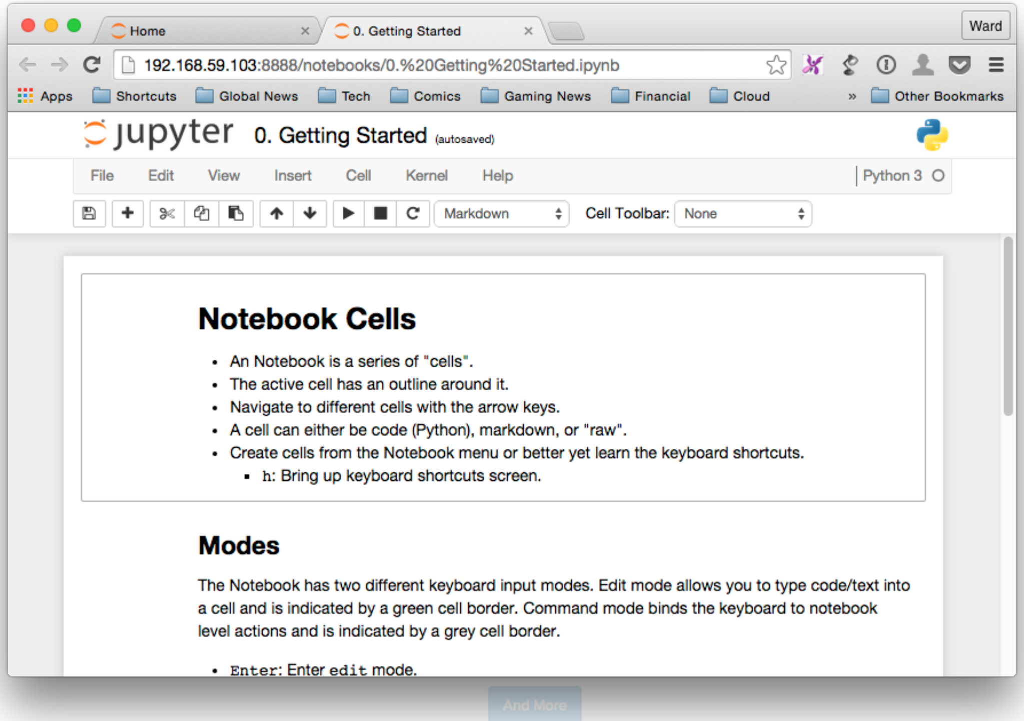
The Notebook has two different keyboard input modes. Edit mode allows you to type code/text into a cell and is indicated by a green cell border. Command mode binds the keyboard to notebook level actions and is indicated by a grey cell border.

- `Enter`: Enter `edit` mode.

And More

Jupyter Notebooks

- An Jupyter Notebook is a collection of *cells*.
  - Markdown
  - Code
  - “Raw” - Raw cells are left ‘as is’ and are not processed.



Notebook Cells

- An Notebook is a series of "cells".
- The active cell has an outline around it.
- Navigate to different cells with the arrow keys.
- A cell can either be code (Python), markdown, or "raw".
- Create cells from the Notebook menu or better yet learn the keyboard shortcuts.
  - `h`: Bring up keyboard shortcuts screen.

Modes

The Notebook has two different keyboard input modes. Edit mode allows you to type code/text into a cell and is indicated by a green cell border. Command mode binds the keyboard to notebook level actions and is indicated by a grey cell border.

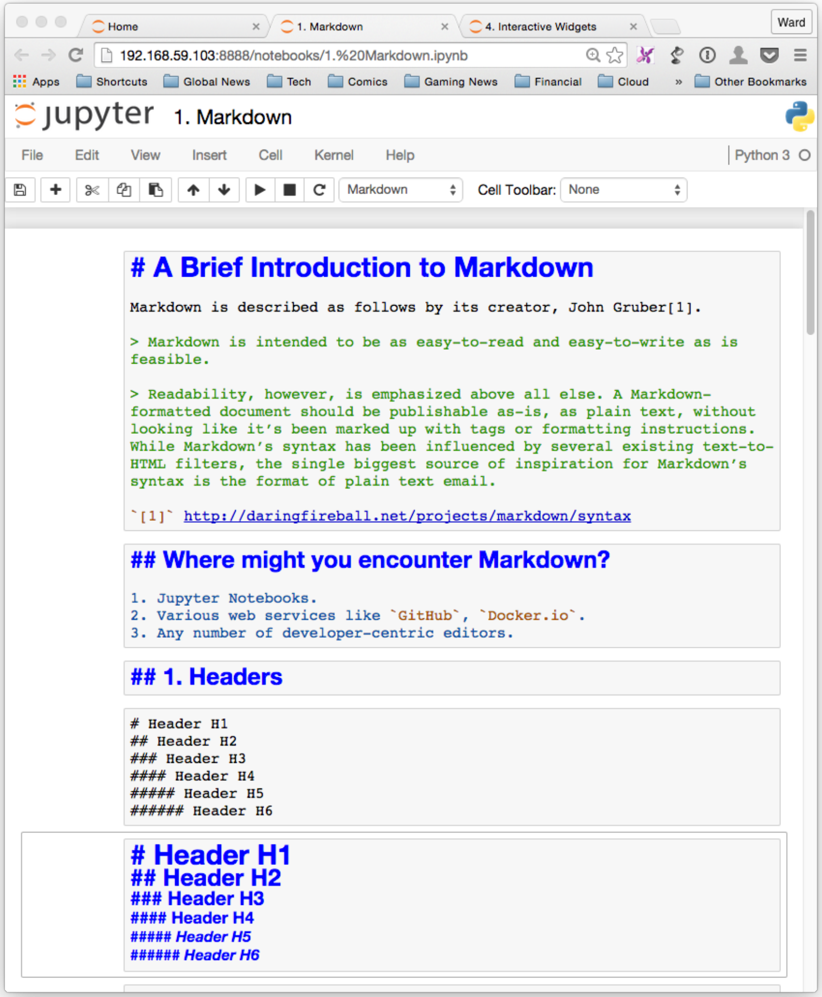
- `Enter`: Enter `edit` mode.

Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter

And More

Markdown Cells



```
# A Brief Introduction to Markdown

Markdown is described as follows by its creator, John Gruber[1].

> Markdown is intended to be as easy-to-read and easy-to-write as is
feasible.

> Readability, however, is emphasized above all else. A Markdown-
formatted document should be publishable as-is, as plain text, without
looking like it's been marked up with tags or formatting instructions.
While Markdown's syntax has been influenced by several existing text-to-
HTML filters, the single biggest source of inspiration for Markdown's
syntax is the format of plain text email.

`[1]` http://daringfireball.net/projects/markdown/syntax
```

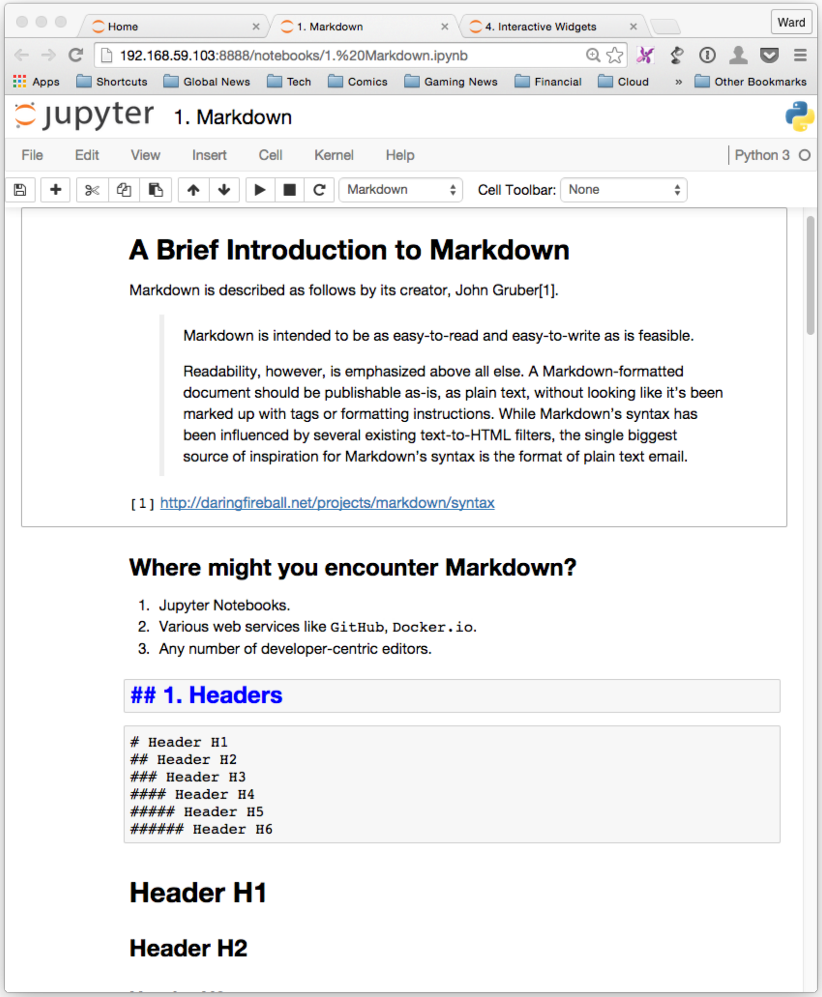
```
## Where might you encounter Markdown?

1. Jupyter Notebooks.
2. Various web services like `GitHub`, `Docker.io`.
3. Any number of developer-centric editors.
```

```
## 1. Headers
```

```
# Header H1
## Header H2
### Header H3
#### Header H4
##### Header H5
###### Header H6
```

```
# Header H1
## Header H2
### Header H3
#### Header H4
##### Header H5
###### Header H6
```



A Brief Introduction to Markdown

Markdown is described as follows by its creator, John Gruber[1].

> Markdown is intended to be as easy-to-read and easy-to-write as is feasible.
>
> Readability, however, is emphasized above all else. A Markdown-formatted document should be publishable as-is, as plain text, without looking like it's been marked up with tags or formatting instructions. While Markdown's syntax has been influenced by several existing text-to-HTML filters, the single biggest source of inspiration for Markdown's syntax is the format of plain text email.

`[1]` http://daringfireball.net/projects/markdown/syntax

Where might you encounter Markdown?

1. Jupyter Notebooks.
2. Various web services like `GitHub`, `Docker.io`.
3. Any number of developer-centric editors.

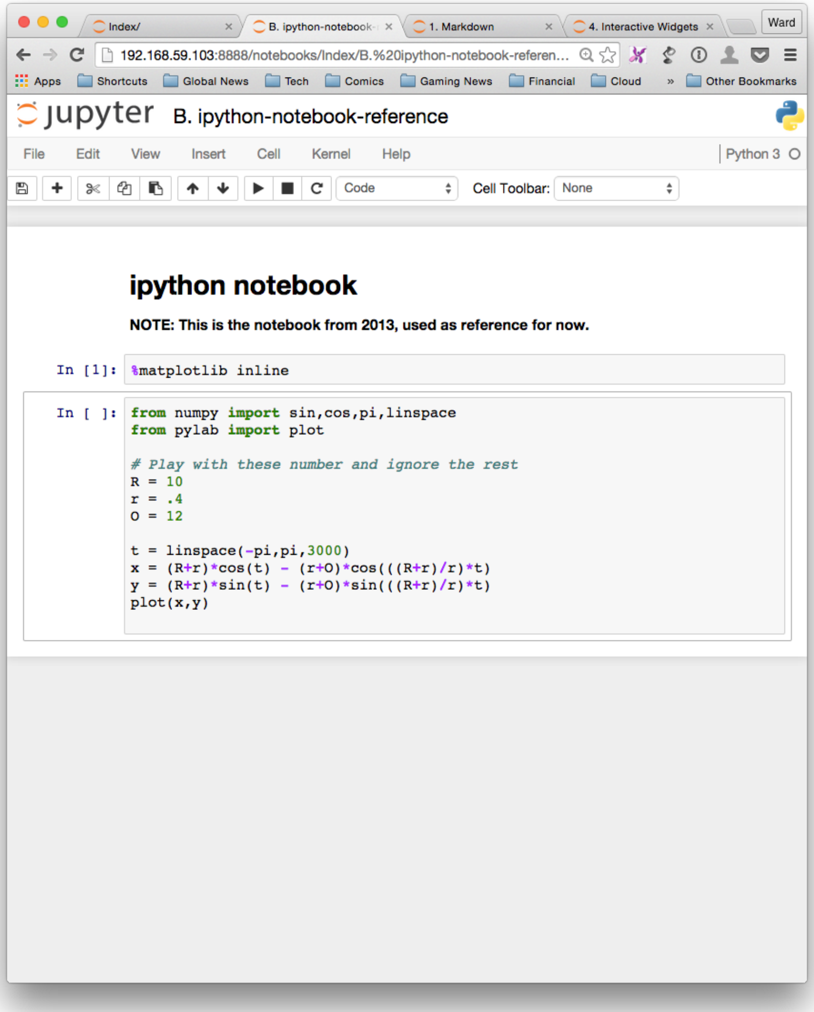
```
## 1. Headers
```

```
# Header H1
## Header H2
### Header H3
#### Header H4
##### Header H5
###### Header H6
```

Header H1

Header H2

Python Cells



B. ipython-notebook-reference

ipython notebook

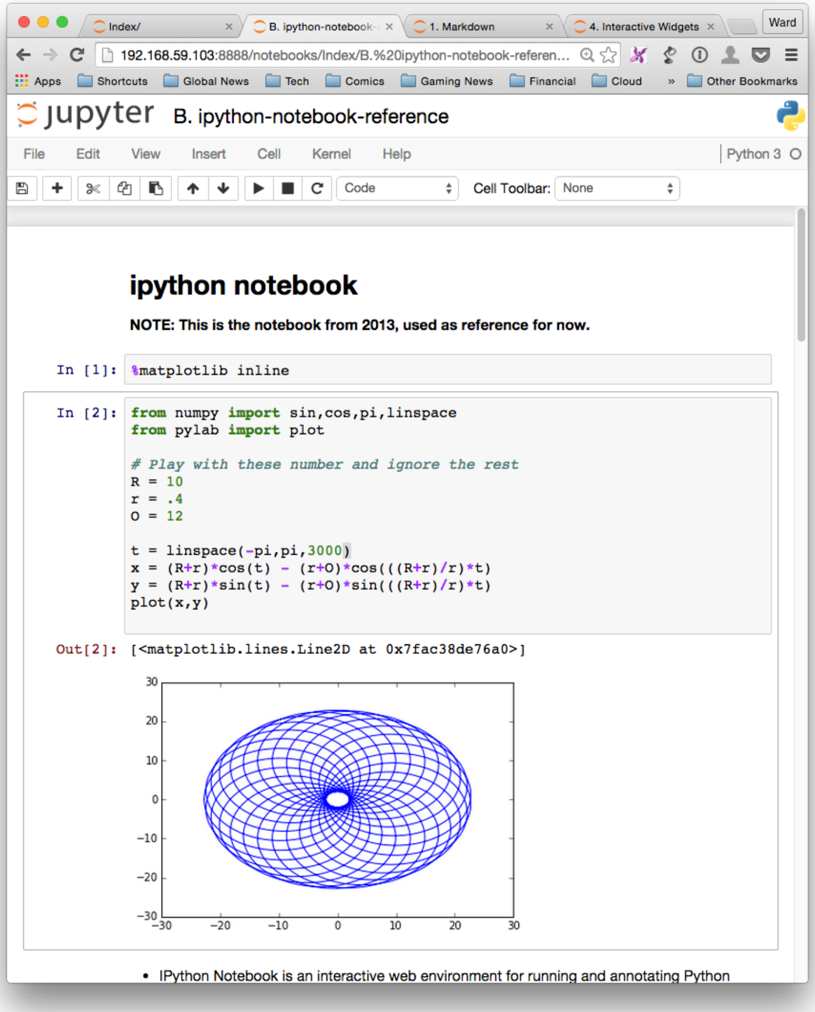
NOTE: This is the notebook from 2013, used as reference for now.

```
In [1]: %matplotlib inline
```

```
In [ ]: from numpy import sin,cos,pi,linspace
        from pylab import plot

        # Play with these number and ignore the rest
        R = 10
        r = .4
        O = 12

        t = linspace(-pi,pi,3000)
        x = (R+r)*cos(t) - (r+O)*cos(((R+r)/r)*t)
        y = (R+r)*sin(t) - (r+O)*sin(((R+r)/r)*t)
        plot(x,y)
```



B. ipython-notebook-reference

ipython notebook

NOTE: This is the notebook from 2013, used as reference for now.

```
In [1]: %matplotlib inline
```

```
In [2]: from numpy import sin,cos,pi,linspace
        from pylab import plot

        # Play with these number and ignore the rest
        R = 10
        r = .4
        O = 12

        t = linspace(-pi,pi,3000)
        x = (R+r)*cos(t) - (r+O)*cos(((R+r)/r)*t)
        y = (R+r)*sin(t) - (r+O)*sin(((R+r)/r)*t)
        plot(x,y)

Out[2]: [<matplotlib.lines.Line2D at 0x7fac38de76a0>]
```

- IPython Notebook is an interactive web environment for running and annotating Python

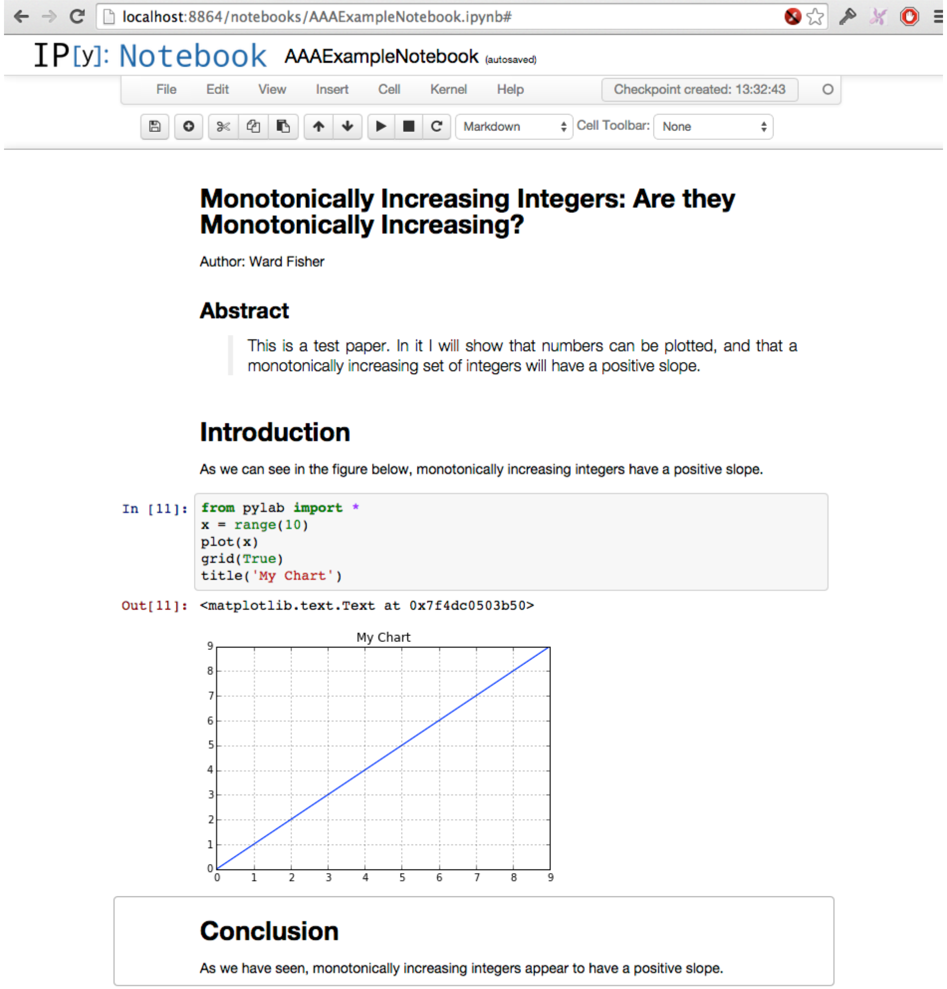
What does this get you?

- A sharable document with embedded, reproducible experimental data analysis.

jupyter

Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter


localhost:8864/notebooks/AAAExampleNotebook.ipynb#
IP[y]: Notebook
AAAExampleNotebook (autosaved)
File
Edit
View
Insert
Cell
Kernel
Help
Checkpoint created: 13:32:43
Markdown
Cell Toolbar:
None
My Chart
0
1
2
3
4
5
6
7
8
9

Monotonically Increasing Integers: Are they Monotonically Increasing?

Author: Ward Fisher

Abstract

> This is a test paper. In it I will show that numbers can be plotted, and that a monotonically increasing set of integers will have a positive slope.

Introduction

As we can see in the figure below, monotonically increasing integers have a positive slope.

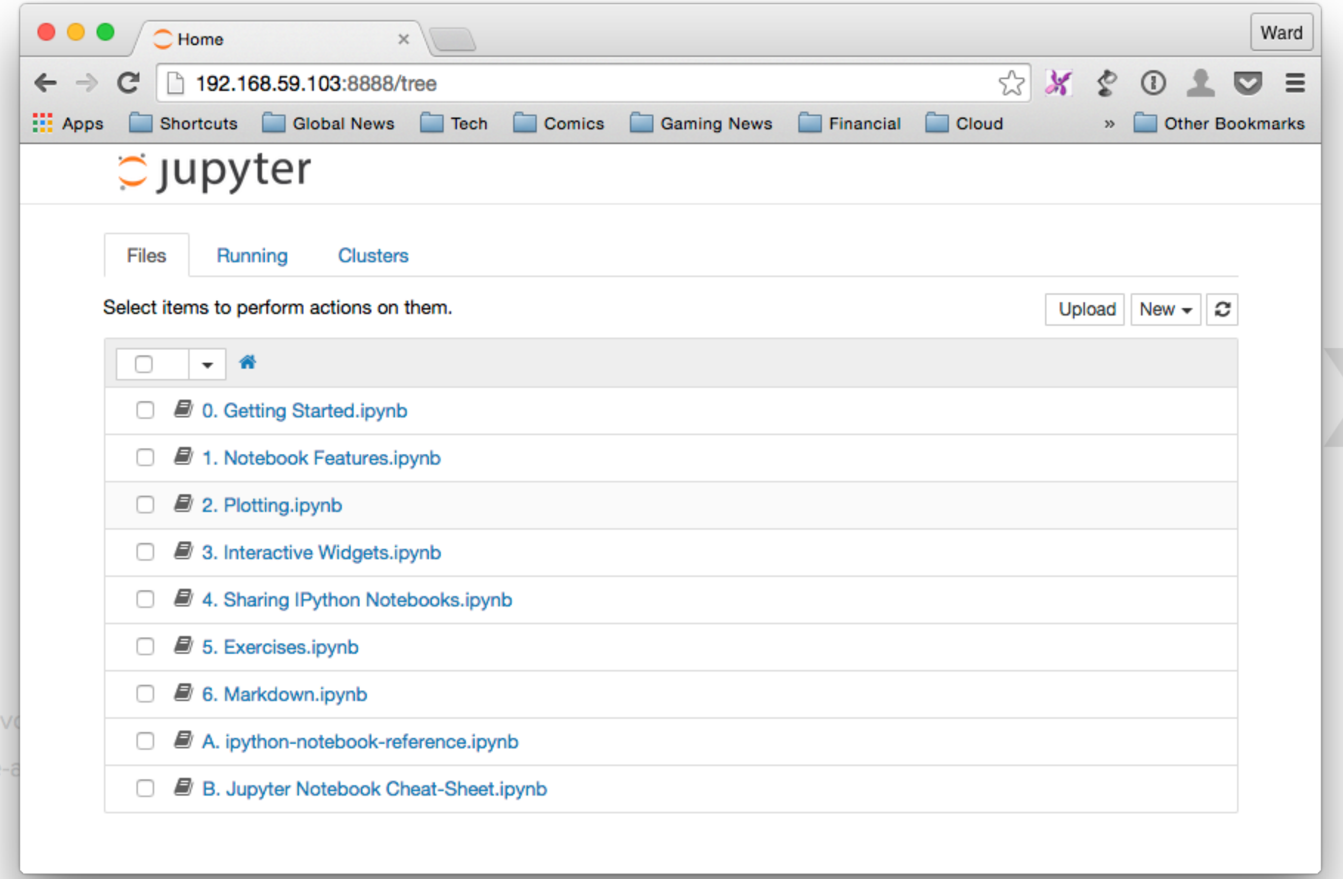
```
In [11]: from pylab import *
         x = range(10)
         plot(x)
         grid(True)
         title('My Chart')

Out[11]: <matplotlib.text.Text at 0x7f4dc0503b50>
```

Conclusion

As we have seen, monotonically increasing integers appear to have a positive slope.

Jupyter Notebook Server



Select items to perform actions on them.

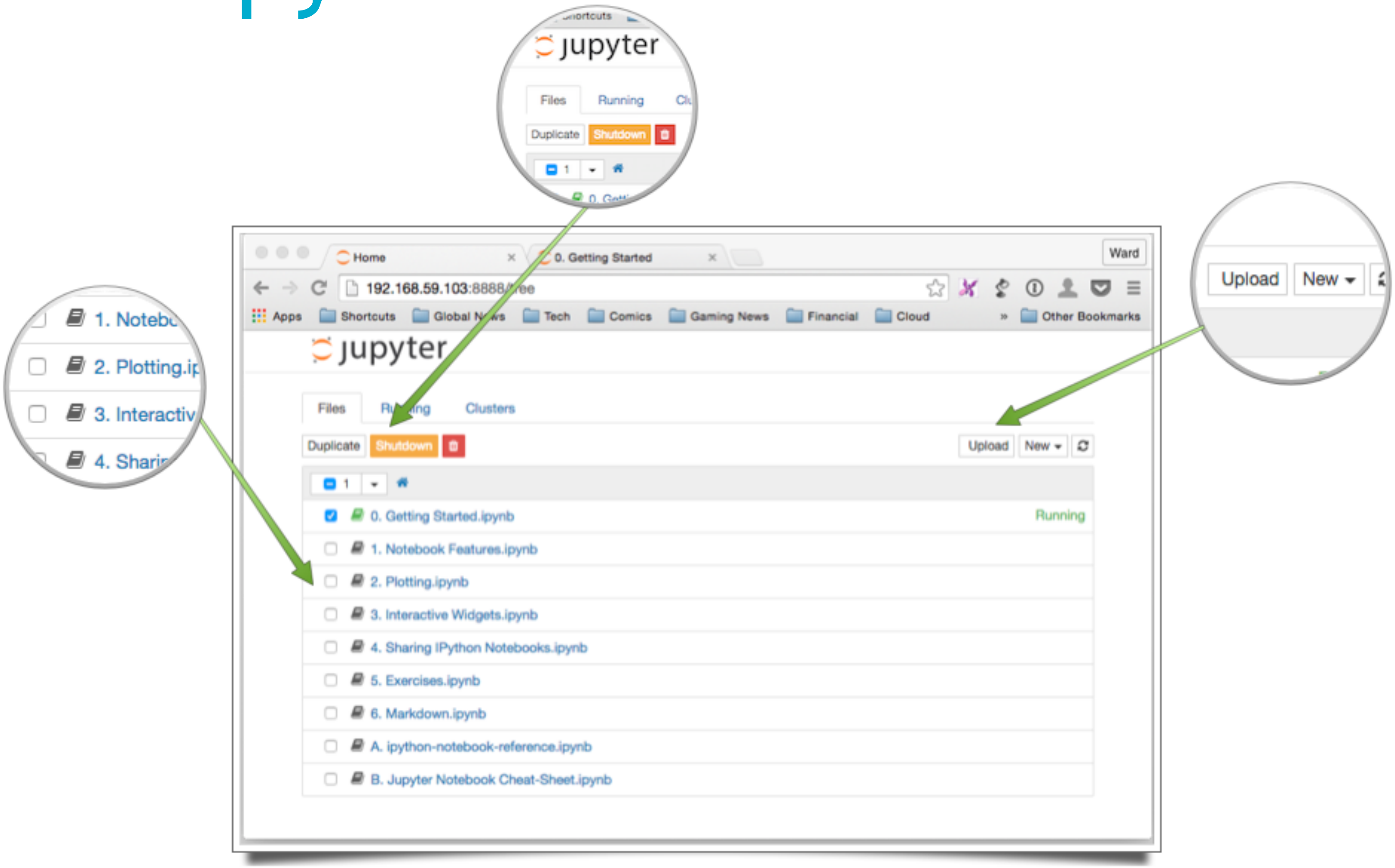
- □ 0. Getting Started.ipynb
- □ 1. Notebook Features.ipynb
- □ 2. Plotting.ipynb
- □ 3. Interactive Widgets.ipynb
- □ 4. Sharing IPython Notebooks.ipynb
- □ 5. Exercises.ipynb
- □ 6. Markdown.ipynb
- □ A. ipython-notebook-reference.ipynb
- □ B. Jupyter Notebook Cheat-Sheet.ipynb

Ev

The language-a

- The Jupyter Notebook Server acts as a dashboard for a collection of individual notebooks.

Jupyter Notebook Server



☑ 0. Getting Started.ipynb Running
□ 1. Notebook Features.ipynb
□ 2. Plotting.ipynb
□ 3. Interactive Widgets.ipynb
□ 4. Sharing IPython Notebooks.ipynb
□ 5. Exercises.ipynb
□ 6. Markdown.ipynb
□ A. ipython-notebook-reference.ipynb
□ B. Jupyter Notebook Cheat-Sheet.ipynb

Installing Jupyter Notebook

- The easiest way to install Jupyter notebook is with a package manager like "Conda"
  - Maintained by Continuum Analytics
  - http://continuum.io/downloads

Installing Jupyter Notebook

- Once Anaconda is installed, you can use the ‘conda’ command to install jupyter notebook (and other packages).

```
$ conda install ipython ipython-notebook
```

Launching Jupyter Notebook

- Jupyter Notebook is launched via the command line.

```
$ ipython notebook
```

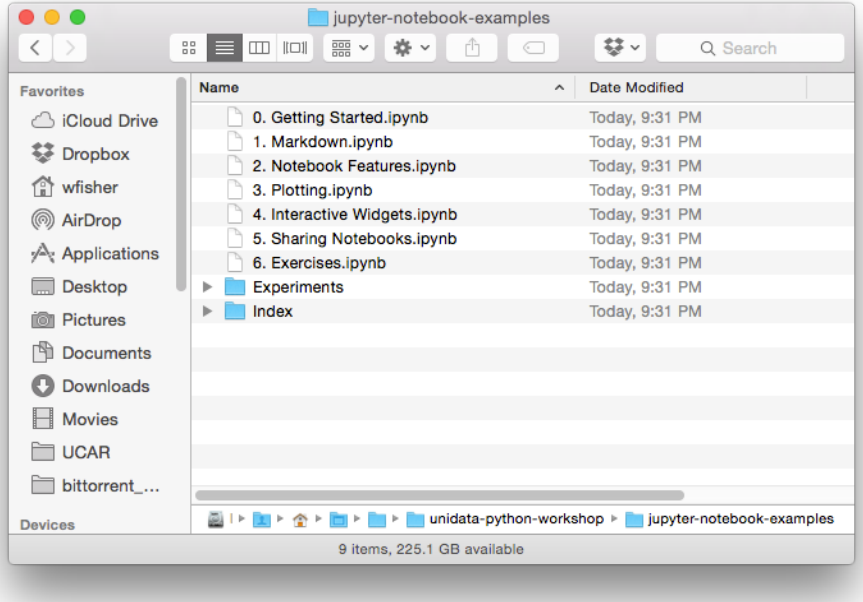
Launching Jupyter Notebook

- Jupyter Notebook is launched via the command line.

```
$ ipython notebook [options]
```

- There are a number of command line options for advanced usage.

Running Jupyter Notebook Server

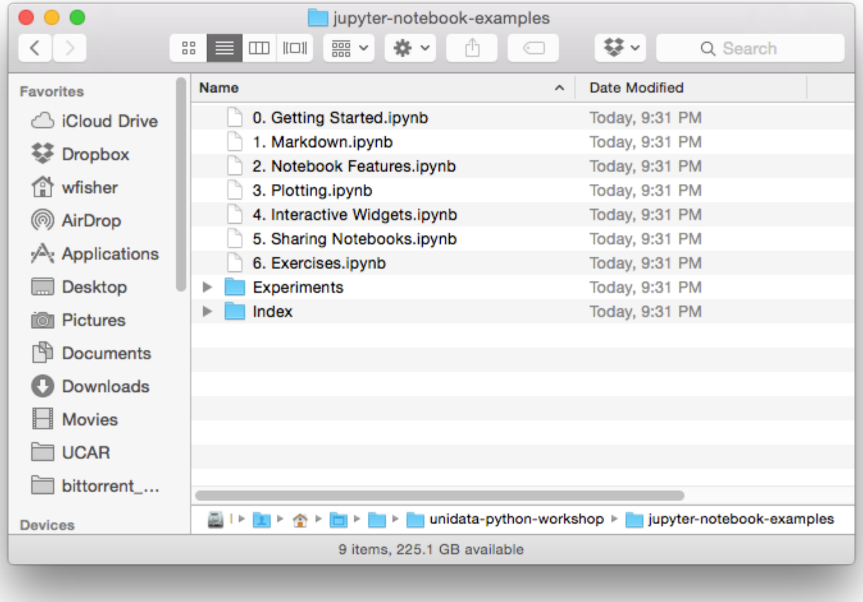


Name	Date Modified
0. Getting Started.ipynb	Today, 9:31 PM
1. Markdown.ipynb	Today, 9:31 PM
2. Notebook Features.ipynb	Today, 9:31 PM
3. Plotting.ipynb	Today, 9:31 PM
4. Interactive Widgets.ipynb	Today, 9:31 PM
5. Sharing Notebooks.ipynb	Today, 9:31 PM
6. Exercises.ipynb	Today, 9:31 PM
Experiments	Today, 9:31 PM
Index	Today, 9:31 PM

9 items, 225.1 GB available

- Notebooks are arranged in a directory.
- You launch Jupyter Notebook from the root of this directory structure.

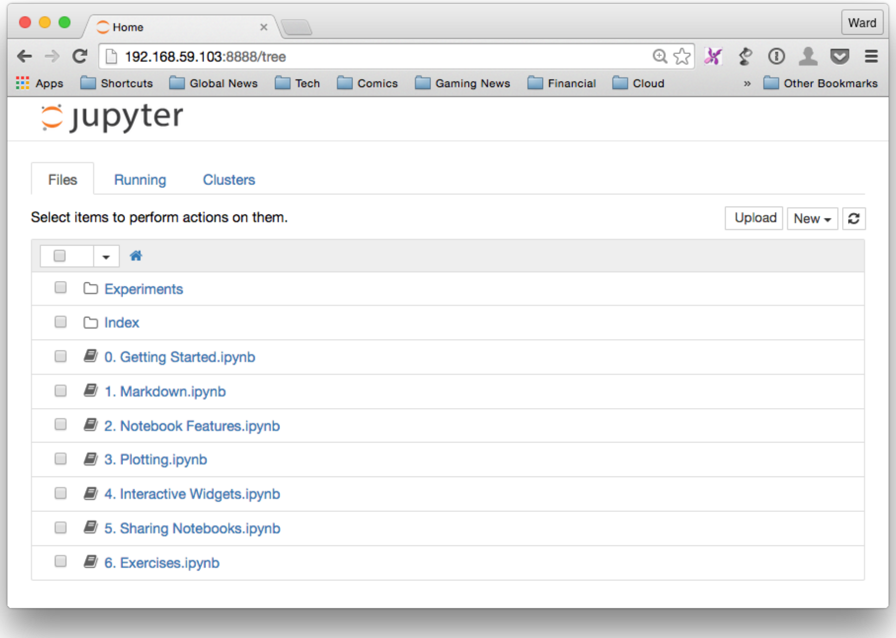
Running Jupyter Notebook Server



Name	Date Modified
0. Getting Started.ipynb	Today, 9:31 PM
1. Markdown.ipynb	Today, 9:31 PM
2. Notebook Features.ipynb	Today, 9:31 PM
3. Plotting.ipynb	Today, 9:31 PM
4. Interactive Widgets.ipynb	Today, 9:31 PM
5. Sharing Notebooks.ipynb	Today, 9:31 PM
6. Exercises.ipynb	Today, 9:31 PM
▶ Experiments	Today, 9:31 PM
▶ Index	Today, 9:31 PM

unidata-python-workshop ▸ jupyter-notebook-examples

9 items, 225.1 GB available



Select items to perform actions on them.

- □ Experiments
- □ Index
- □ 0. Getting Started.ipynb
- □ 1. Markdown.ipynb
- □ 2. Notebook Features.ipynb
- □ 3. Plotting.ipynb
- □ 4. Interactive Widgets.ipynb
- □ 5. Sharing Notebooks.ipynb
- □ 6. Exercises.ipynb

Working in Jupyter Notebook

Switching to the Browser.