# Starting with Jupyter in Your Computer

Qixun Qu

# What is Jupyter?

- “The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.”
- “Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.”

(Copy from official web page of Jupyter.)

# Install Jupyter

- First, create a new environment names “jupy”.

  conda create --name jupy python=3

- Activate “jupy”

  In Linux and macOS:

  source activate jupy

  In Windows:

  activate jupy

# Install Jupyter

- Input command:

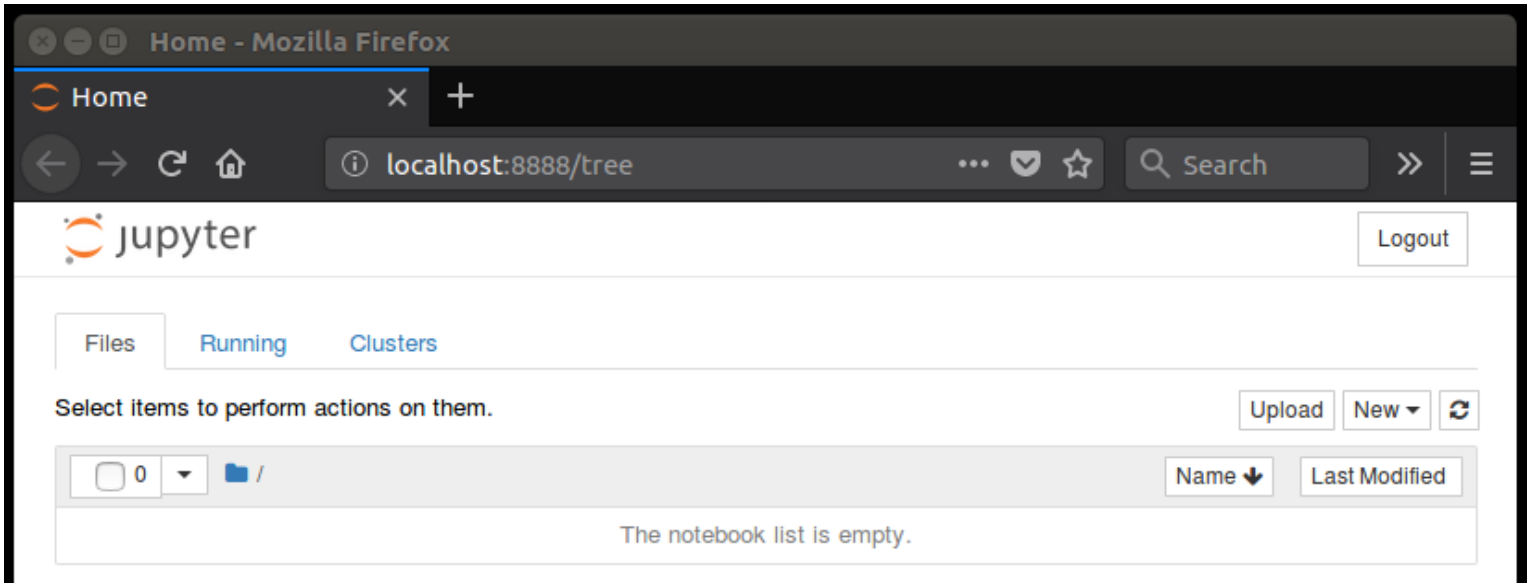
  pip install jupyter
- Verify installation:

  conda list
- You can find jupyter in the list.

# Start Jupyter

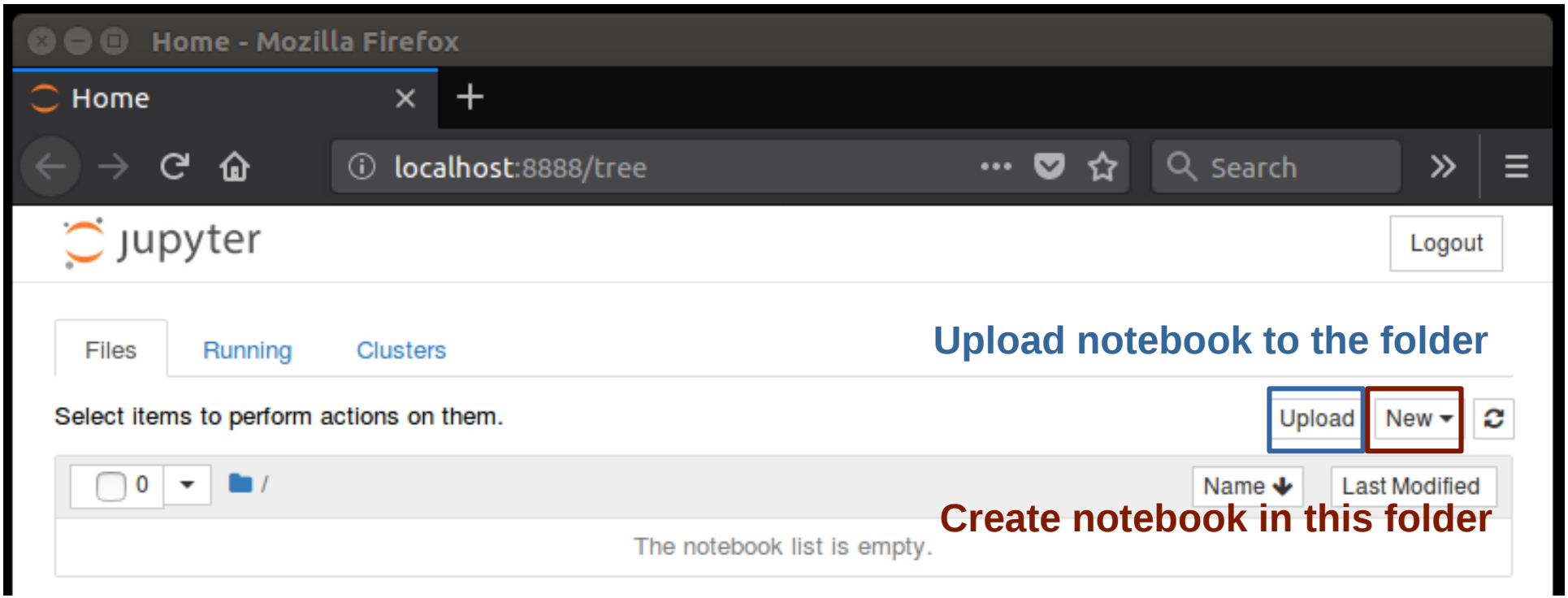
- At first, create a directory names “test_jupy”.
- Change the working directory to “test_jupy”.
- Start Jupyter by the command:

  jupyter notebook
- Browser is opened and it looks like the image below.



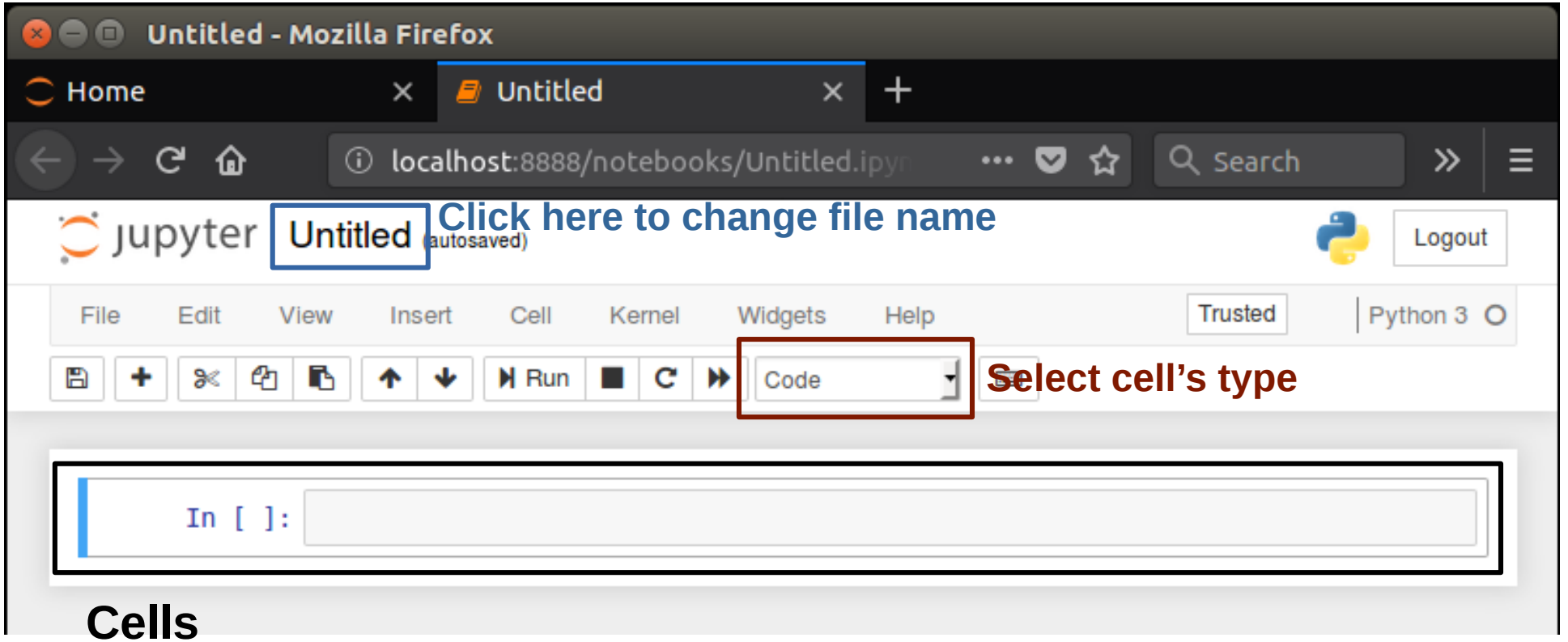
# Create Notebook

- Click “New”, select “Python 3”.
- Browser opens a new tab.



# Notebook

- Change the file's name to "test".
- Change cell's type.



# Cell Types and Hotkeys

- Select one cell, and apply different hotkeys.

In [ ]:
```
# Cell of Code  Hotkey: Esc + y
```

```
**Cell of Markdown**  Hotkey: Esc + m
```

```
Cell of Raw NBConvert (plain text)  Hotkey: Esc + r
```

```
# Cell of Header 1  Hotkey: Esc + 1
```

```
## Cell of Header 2  Hotkey: Esc + 2
```

```
### Cell of Header 3  Hotkey: Esc + 3
```

```
#### Cell of Header 4  Hotkey: Esc + 4
```

```
##### Cell of Header 5  Hotkey: Esc + 5
```

```
###### Cell of Header 6  Hotkey: Esc + 6
```

# Other Hotkeys

- Save notebook: ESC + s
- Delete selected cell：　ESC+ d + d
- Insert cell above: Esc + a
- Insert cell below: Esc + b
- Run selected cell: Ctrl + Enter
- See all hotkeys in:

  Help → Keyboard Shortcuts

# Manage Notebook

- Close “test” tab, back to “Home” page.
- The notebook you created appears in the list.
- Manage notebook after selecting the checkbox.

# Run A Sample

- Create another notebook names “fibo”.
- In code cell, write a function to generate a Fibonacci sequence in given length.
- Click Ctrl + Enter to run cell.
- No output since it’s a function.

In [1]:
```
def fibo(n):
    fibonacci = []
    a, b = 0, 1
    for i in range(n):
        fibonacci.append(a)
        a, b = b, a + b
    return fibonacci
```

# Run A Sample

- Insert a code cell below (Esc + b).
- In this cell, we call the function “fibo”, and print the output.
- Run cell by clicking Ctrl + Enter.

```
In [1]: def fibo(n):
            fibonacci = []
            a, b = 0, 1
            for i in range(n):
                fibonacci.append(a)
                a, b = b, a + b
            return fibonacci
```

```
In [2]: print(fibo(10))

        [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

**Output Region**

# Run A Sample

- Add descriptions.

```
This is a function to generate Fibbonacci sequence.
It has one parameter names "n" which indicates the
the length of the sequence.
```

In [1]:
```
def fibo(n):
    fibonacci = []
    a, b = 0, 1
    for i in range(n):
        fibonacci.append(a)
        a, b = b, a + b
    return fibonacci
```

```
Call the function "fibo" with n equals to 10,
and print the output.
```

In [2]:
```
print(fibo(10))
```

```
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

# What’s More

- In code cell, you can import libraries or implement a class. Each code cell seems like a python script. Remember, run it before call it.
- Plots can also be shown in Output region.

# Stop Jupyter

- In command line, press Ctrl + c.

  Input “y” and press Enter in 5 secends.

  Then you quit the jupyter note book.
- Or press Ctrl + c + c.

# More Reading

- Jupyter Documentation
- A gallery of interesting Jupyter Notebooks