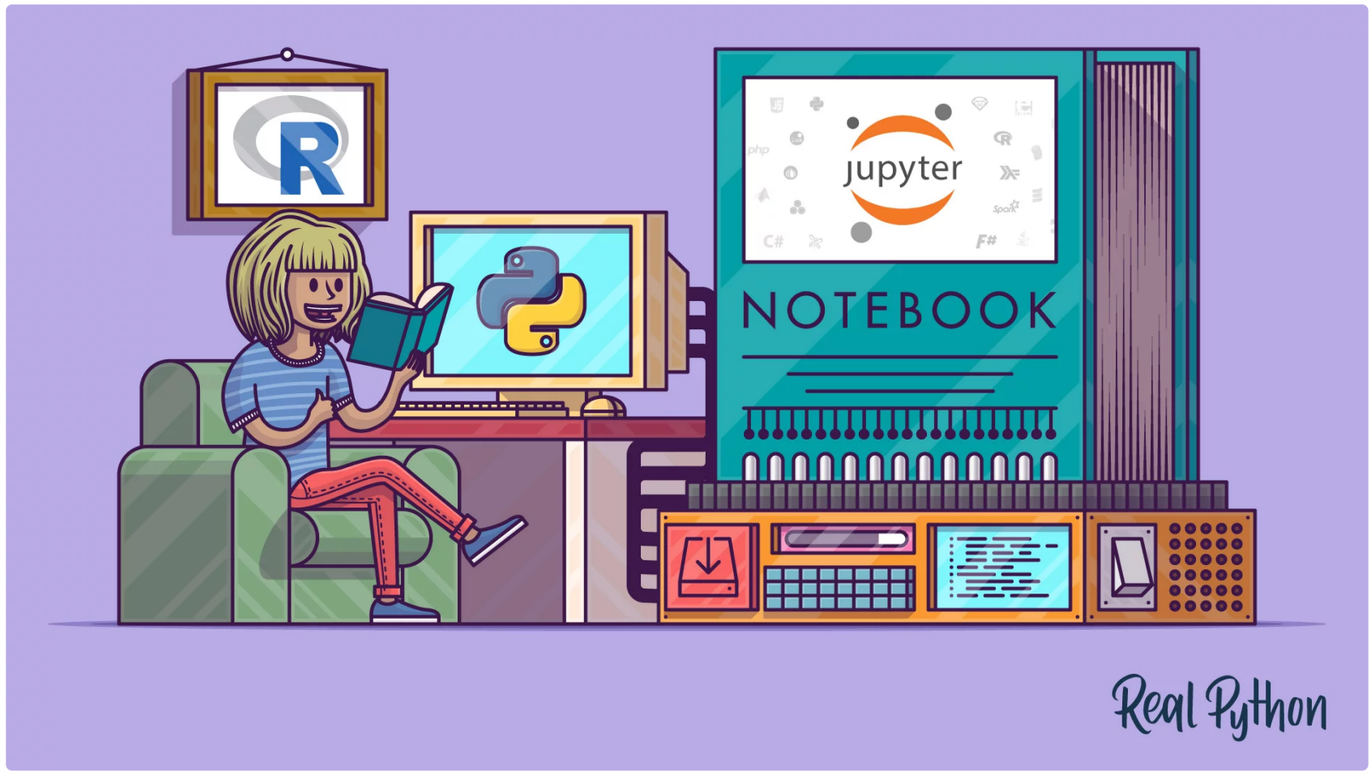


Jupyter Notebook: An Introduction

by Mike Driscoll Jan 28, 2019 5 Comments intermediate tools

View in CMS » Preflight »

Tweet Share Email

Table of Contents

Watch Now This tutorial has a related video course created by the Real Python team. Watch it together with th written tutorial to deepen your understanding: **Using Jupyter Notebooks**

Help

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter.

Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

Getting Up and Running With Jupyter Notebook

The Jupyter Notebook is not included with Python, so if you want to try it out, you will need to install Jupyter.

There are many distributions of the Python language. This article will focus on just two of them for the purposes of installing Jupyter Notebook. The most popular is CPython, which is the reference version of Python that you can get from their website. It is also assumed that you are using **Python 3**.

Installation

If so, then you can use a handy tool that comes with Python called **pip** to install Jupyter Notebook like this:

Shell

```
$ pip install jupyter
```

The next most popular distribution of Python is Anaconda. Anaconda has its own installer tool called **conda** that you could use for installing a third-party package. However, Anaconda comes with many scientific libraries preinstalled, including the Jupyter Notebook, so you don't actually need to do anything other than install Anaconda itself.

Starting the Jupyter Notebook Server

Now that you have Jupyter installed, let's learn how to use it. To get started, all you need to do is open up your terminal application and go to a folder of your choice. I recommend using something like your Documents folder to start out with and create a subfolder there called *Notebooks* or something else that is easy to remember.

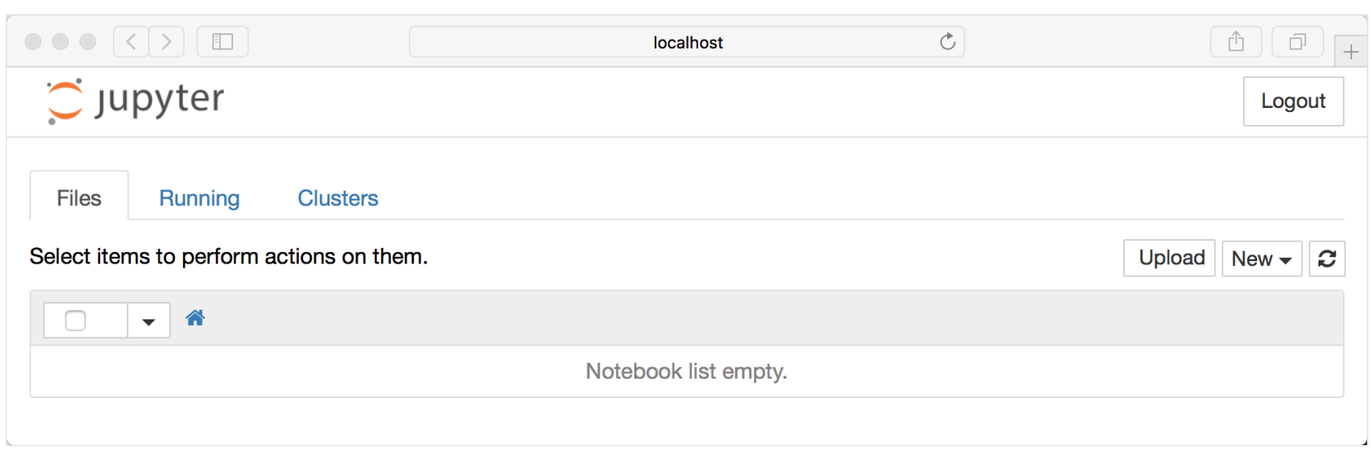
Then just go to that location in your terminal and run the following command:

Shell

```
$ jupyter notebook
```

This will start up Jupyter and your default browser should start (or open a new tab) to the following URL: http://localhost:8888/tree

Your browser should now look something like this:



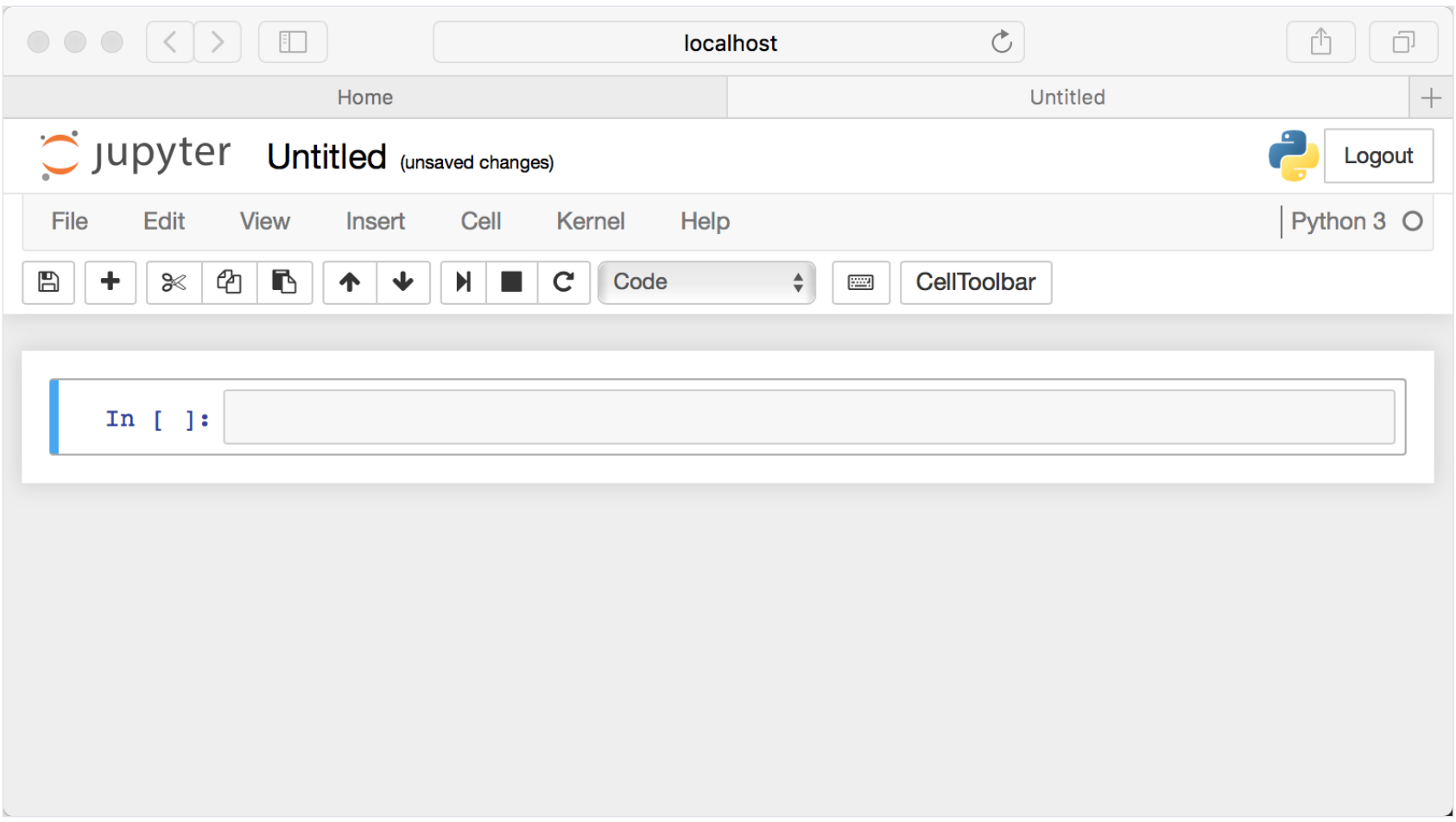
Note that right now you are not actually running a Notebook, but instead you are just running the Notebook server. Let's actually create a Notebook now!

Creating a Notebook

Now that you know how to start a Notebook server, you should probably learn how to create an actually Notebook document.

All you need to do is click on the *New* button (upper right), and it will open up a list of choices. On my machine, I happen to have Python 2 and Python 3 installed, so I can create a Notebook that uses either of these. For simplicity's sake, let's choose Python 3.

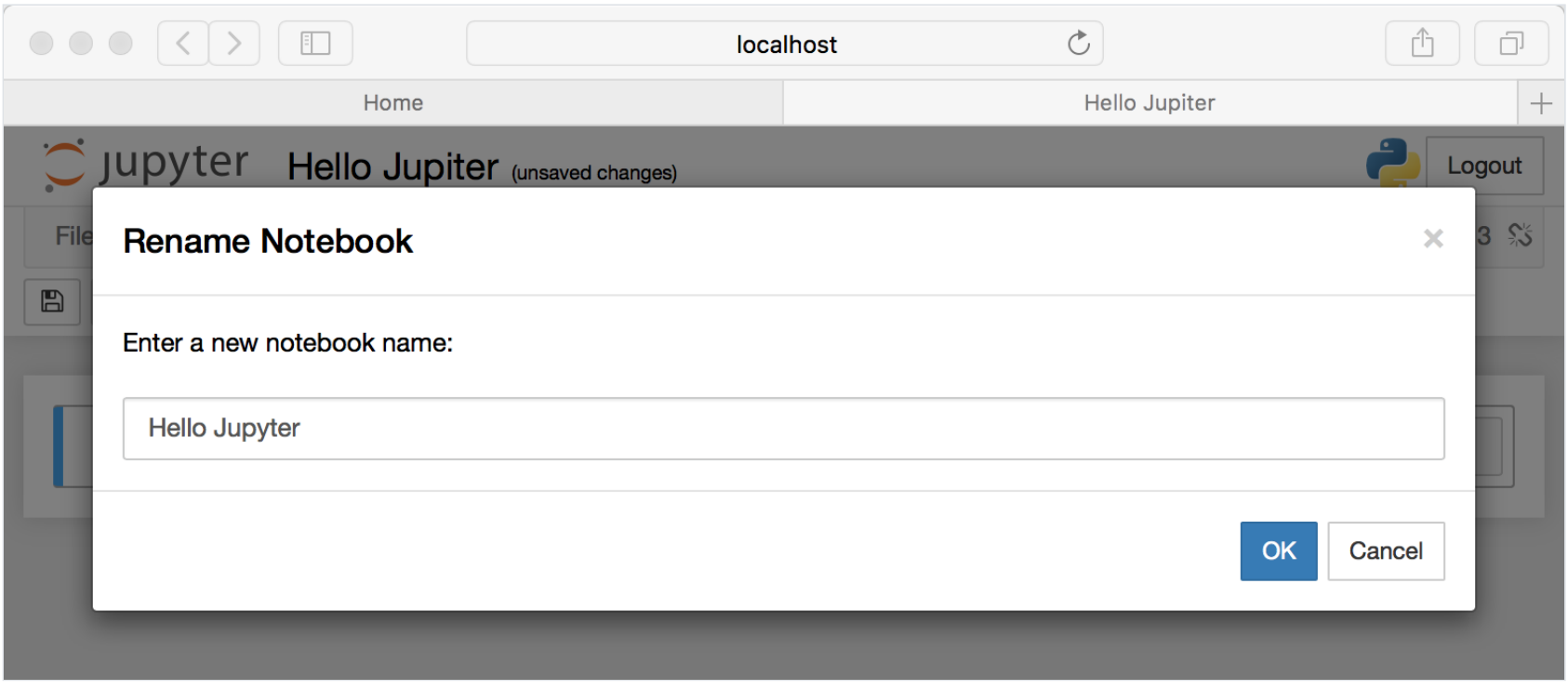
Your web page should now look like this:



Naming

You will notice that at the top of the page is the word *Untitled*. This is the title for the page and the name of your Notebook. Since that isn't a very descriptive name, let's change it!

Just move your mouse over the word *Untitled* and click on the text. You should now see an in-browser dialog titled *Rename Notebook*. Let's rename this one to *Hello Jupyter*:



Running Cells

A Notebook's cell defaults to using code whenever you first create one, and that cell uses the kernel that you chose when you started your Notebook.

In this case, you started yours with Python 3 as your kernel, so that means you can write Python code in your code cells. Since your initial Notebook has only one empty cell in it, the Notebook can't really do anything.

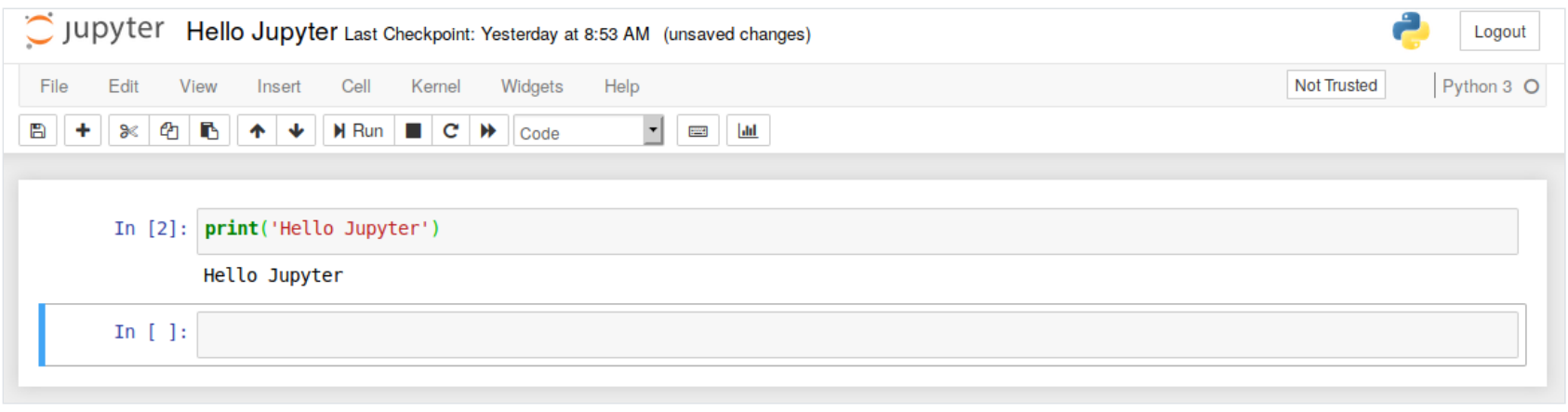
Thus, to verify that everything is working as it should, you can add some Python code to the cell and try running its contents.

Let's try adding the following code to that cell:

```
Python
print('Hello Jupyter!')
```

Running a cell means that you will execute the cell's contents. To execute a cell, you can just select the cell and click the *Run* button that is in the row of buttons along the top. It's towards the middle. If you prefer using your keyboard, you can just press ⇧Shift + Enter ↩.

When I ran the code above, the output looked like this:



If you have multiple cells in your Notebook, and you run the cells in order, you can share your variables and imports across cells. This makes it easy to separate out your code into logical chunks without needing to reimport libraries or recreate variables or functions in every cell.

When you run a cell, you will notice that there are some square braces next to the word *In* to the left of the cell. The square braces will auto fill with a number that indicates the order that you ran the cells. For example, if you open a fresh Notebook and run the first cell at the top of the Notebook, the square braces will fill with the number *1*.

The Menus

The Jupyter Notebook has several menus that you can use to interact with your Notebook. The menu runs along the top of the Notebook just like menus do in other applications. Here is a list of the current menus:

- *File*
- *Edit*
- *View*
- *Insert*
- *Cell*
- *Kernel*
- *Widgets*
- *Help*

Let's go over the menus one by one. This article won't go into detail for every single option in every menu, but it will focus on the items that are unique to the Notebook application.

The first menu is the File menu. In it, you can create a new Notebook or open a preexisting one. This is also where you would go to rename a Notebook. I think the most interesting menu item is the *Save and Checkpoint* option. This allows you to create checkpoints that you can roll back to if you need to.

Next is the *Edit* menu. Here you can cut, copy, and paste cells. This is also where you would go if you wanted to delete, split, or merge a cell. You can reorder cells here too.