

Tutorial 0: Environment Setup & Python Basics

What will be covered:

1. Python Installation
2. Anaconda Installation
3. IPython Tutorial

Preparation:

- Download & unzip *Tutorial0_python*
-

1. Python Installation (Local Machine)

Python: Install python3 version, specifically Python 3.5 / 3.7. Python 2 will no longer be supported at the end of 2019.

1.1 Install python (Window) version

- Go to <https://www.python.org/downloads/windows/>
- Select “Latest Python 3 Release - Python 3.7.4” (you can download python 3.5/ 3.7 release)
- Scroll to the bottom, click “Windows x86-64 executable installer”
- Open the downloaded python exe
- Click “Install now” and “Yes”

The screenshot shows a web browser window with two tabs open. The active tab is titled "Python Releases for Windows". The URL in the address bar is <https://www.python.org/downloads/windows/>. The page content is the "Python Releases for Windows" section of the Python website.

The page features a dark blue header with the Python logo and the word "python" in white. Below the header is a navigation menu with links for "About", "Downloads", "Documentation", "Community", "Success Stories", "News", and "Events".

The main content area has a title "Python Releases for Windows" and two sections: "Stable Releases" and "Pre-releases".

Stable Releases:

- [Latest Python 3 Release - Python 3.7.4](#)
- [Latest Python 2 Release - Python 2.7.16](#)
- [Python 3.7.4 - July 8, 2019](#)
- Note that Python 3.7.4 cannot be used on Windows XP or earlier.**
- [Download Windows help file](#)

Pre-releases:

- [Python 3.8.0b3 - July 29, 2019](#)
- [Download Windows help file](#)
- [Download Windows x86-64 embeddable zip file](#)
- [Download Windows x86-64 executable installer](#)

Windows users

- **Changed in 3.7.4** OpenSSL has been updated from 1.1.0 to 1.1.1 and SQLite updated to 3.28.0.
- The binaries for AMD64 will also work on processors that implement the Intel 64 architecture. (Also known as the "x64" architecture, and formerly known as both "EM64T" and "x86-64".)
- There are now "web-based" installers for Windows platforms; the installer will download the needed software components at installation time.
- There are redistributable zip files containing the Windows builds, making it easy to redistribute Python as part of another software package. Please see the documentation regarding [Embedded Distribution](#) for more information.

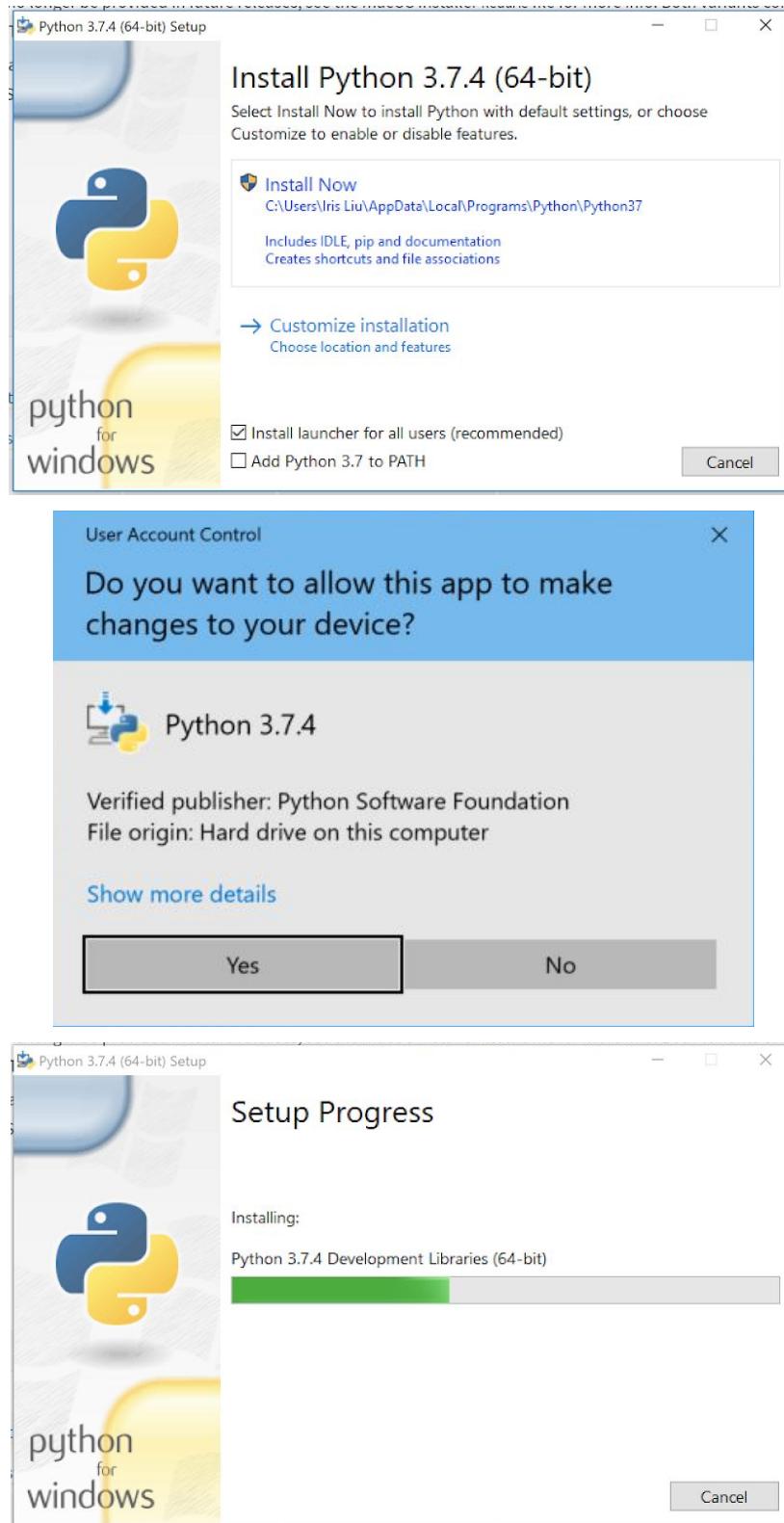
macOS users

- **Changed in 3.7.4** OpenSSL has been updated from 1.1.0 to 1.1.1 and SQLite updated to 3.28.0.
- For Python 3.7.4, we provide two binary installer options for download. The default variant is 64-bit-only and works on macOS 10.9 (Mavericks) and later systems. **Changed in 3.7.4** The 64-bit/32-bit variant that also works on very old versions of macOS, from 10.6 (Snow Leopard) on, is now deprecated and will no longer be provided in future releases; see the macOS installer `ReadMe` file for more info. Both variants come with batteries-included versions of Tcl/Tk 8.6 for users of IDLE and other tkinter-based GUI applications; third-party and system versions of Tcl/Tk are no longer used.
- Both python.org installer variants include private copies of OpenSSL. Please carefully read the `Important Information` displayed during installation for information about SSL/TLS certificate validation and the `Install Certificates.command`.

[Full Changelog](#)

Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		68111671e5b2db4aef7b9ab01bf0f9be	23017663	SIG
XZ compressed source tarball	Source release		d33e4aae66097051c2eca45ee3604803	17131432	SIG
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7583daff1a442cba8cee08e6	34898416	SIG
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	5dd605c38217a45773bf5e4a936b241f	28082845	SIG
Windows help file	Windows		d63999573a2c06b2ac56cade6b4f7cd2	8131761	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	9b00c8cf6d9ec0b9abe83184a40729a2	7504391	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	a702b4b0ad76debdb3043a583e563400	26680368	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	28cb1c608bbd73ae8e53a3bd351b4bd2	1362904	SIG
Windows x86 embeddable zip file	Windows		9fab3b81f8841879fda94133574139d8	6741626	SIG
Windows x86 executable installer	Windows		33cc602942a54446a3d6451476394789	25663848	SIG
Windows x86 web-based installer	Windows		1b670cfa5d317df82c30983ea371d87c	1324608	SIG



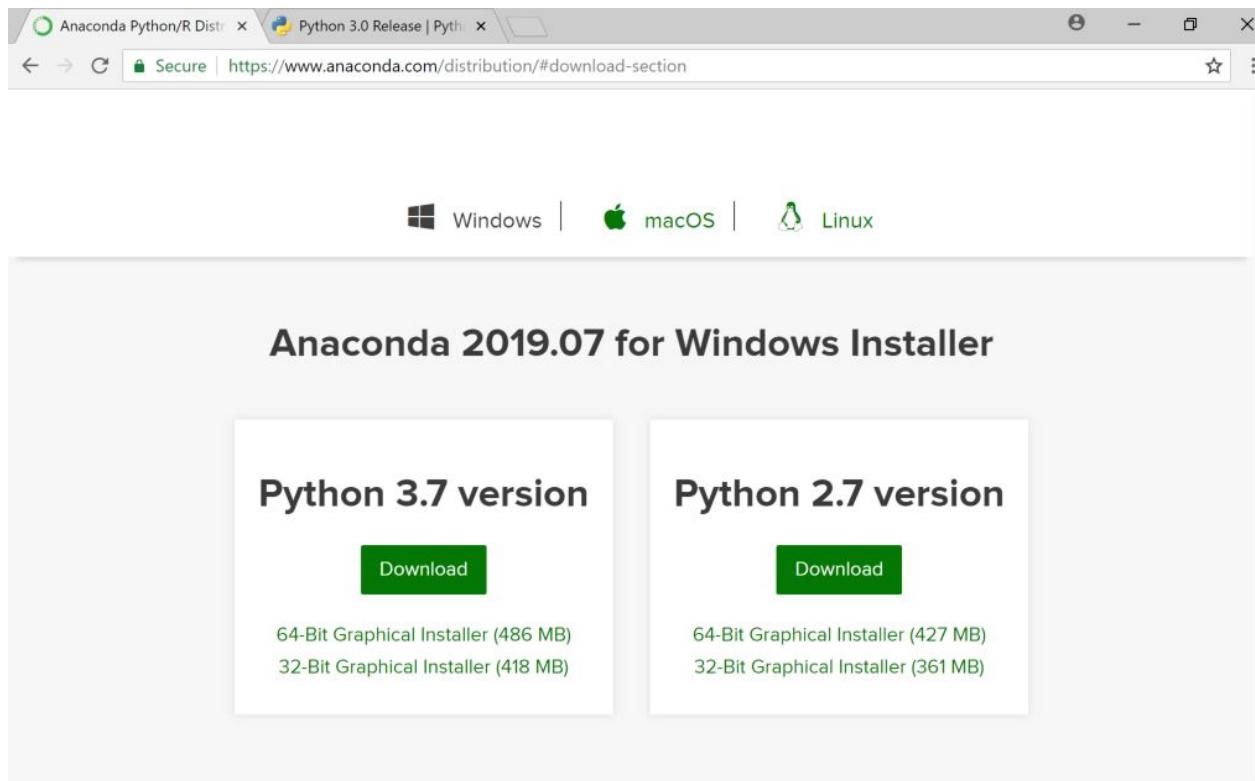
2. Anaconda Installation

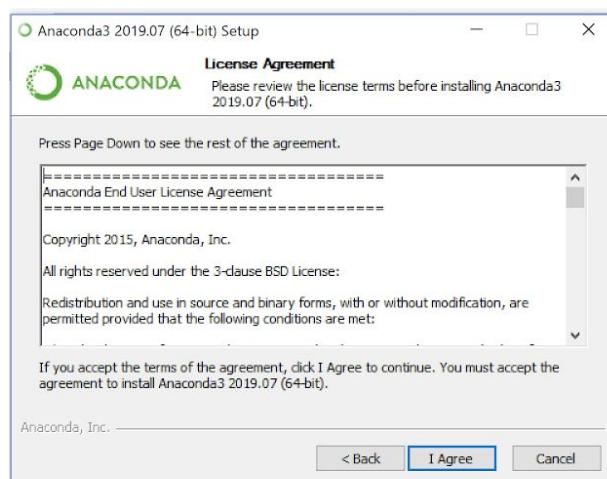
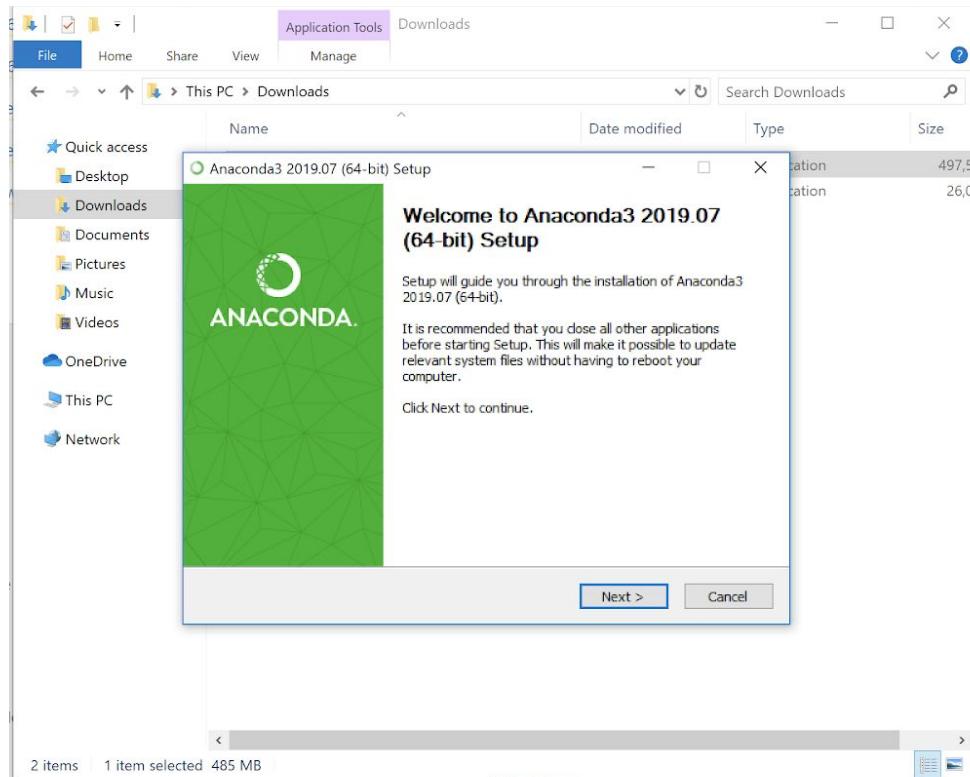
Installing Anaconda: If you decide to work locally, we recommend using the free [Anaconda Python distribution](#), which provides an easy way for you to handle package dependencies. Please be sure to download the Python 3 version, which currently installs Python 3.7.

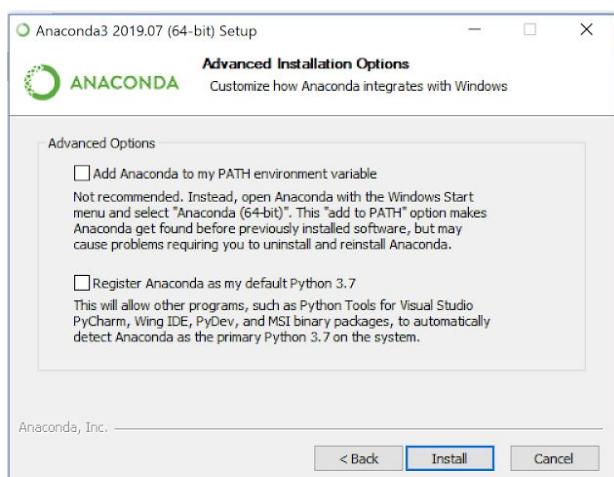
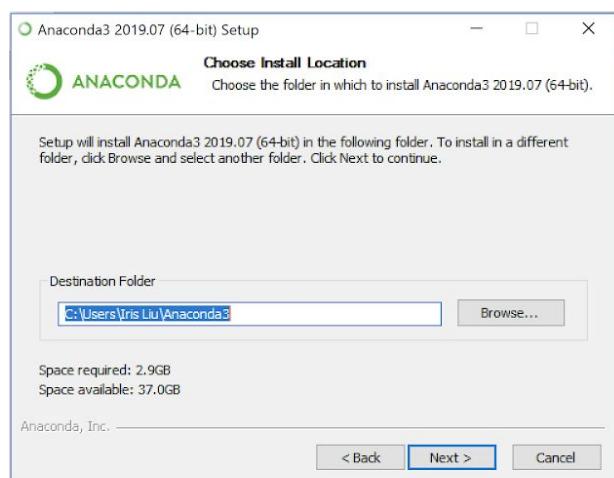
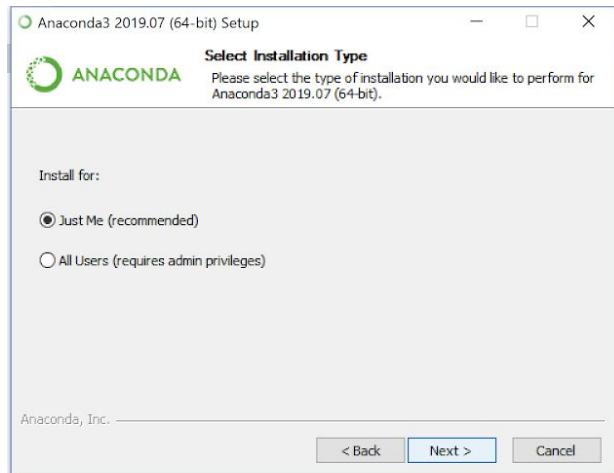
Installation takes around 10 mins. You may want to download this first!

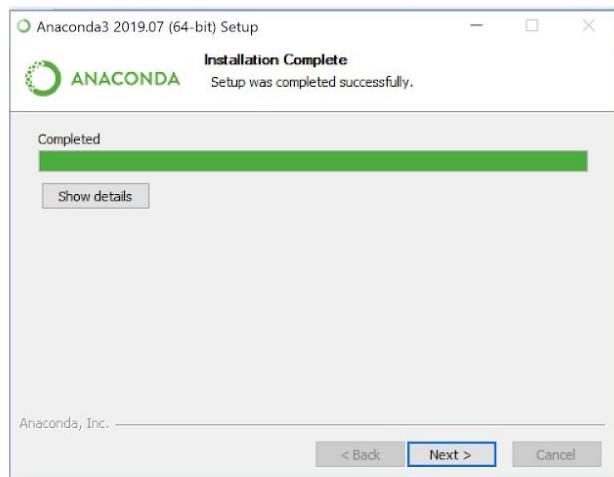
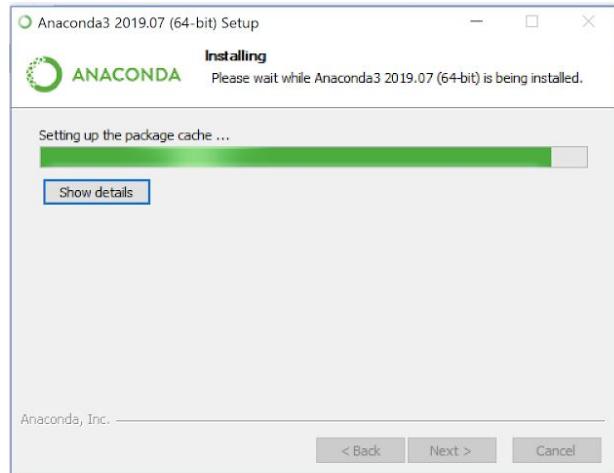
2.1 Install anaconda (Window) version

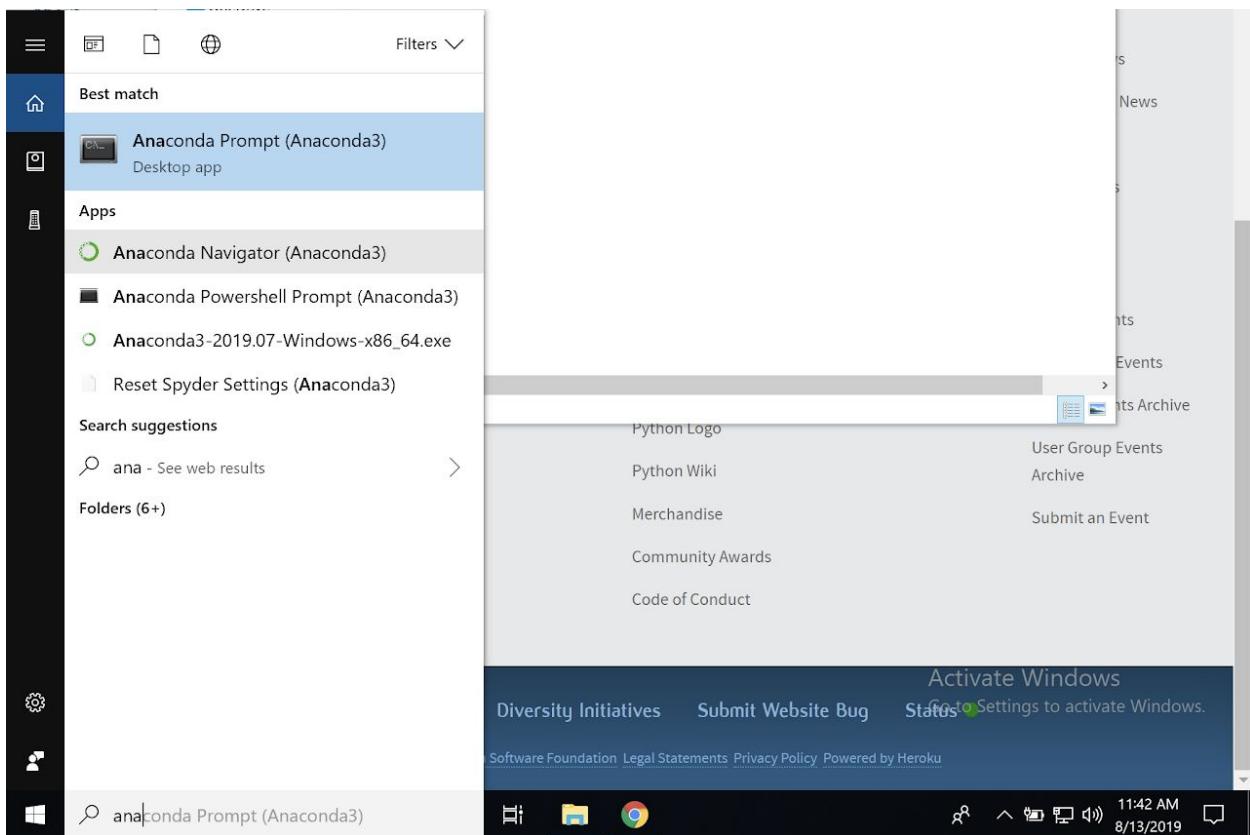
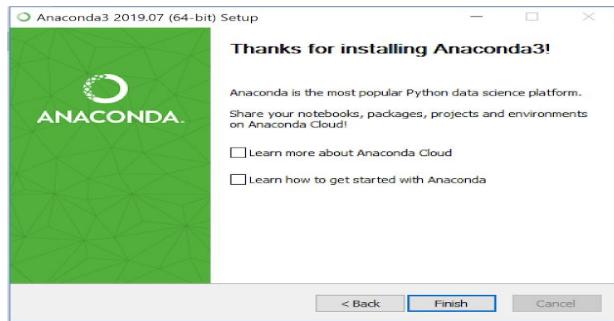
- Go to <https://www.anaconda.com/distribution/>
- Select “Windows” if you are downloading on your window machine. Click “Download” for “Python 3.7 version”.
- Go to “Downloads” and double click to open the anaconda application
- Once Anaconda is successfully installed, you should be able to search it in window Home









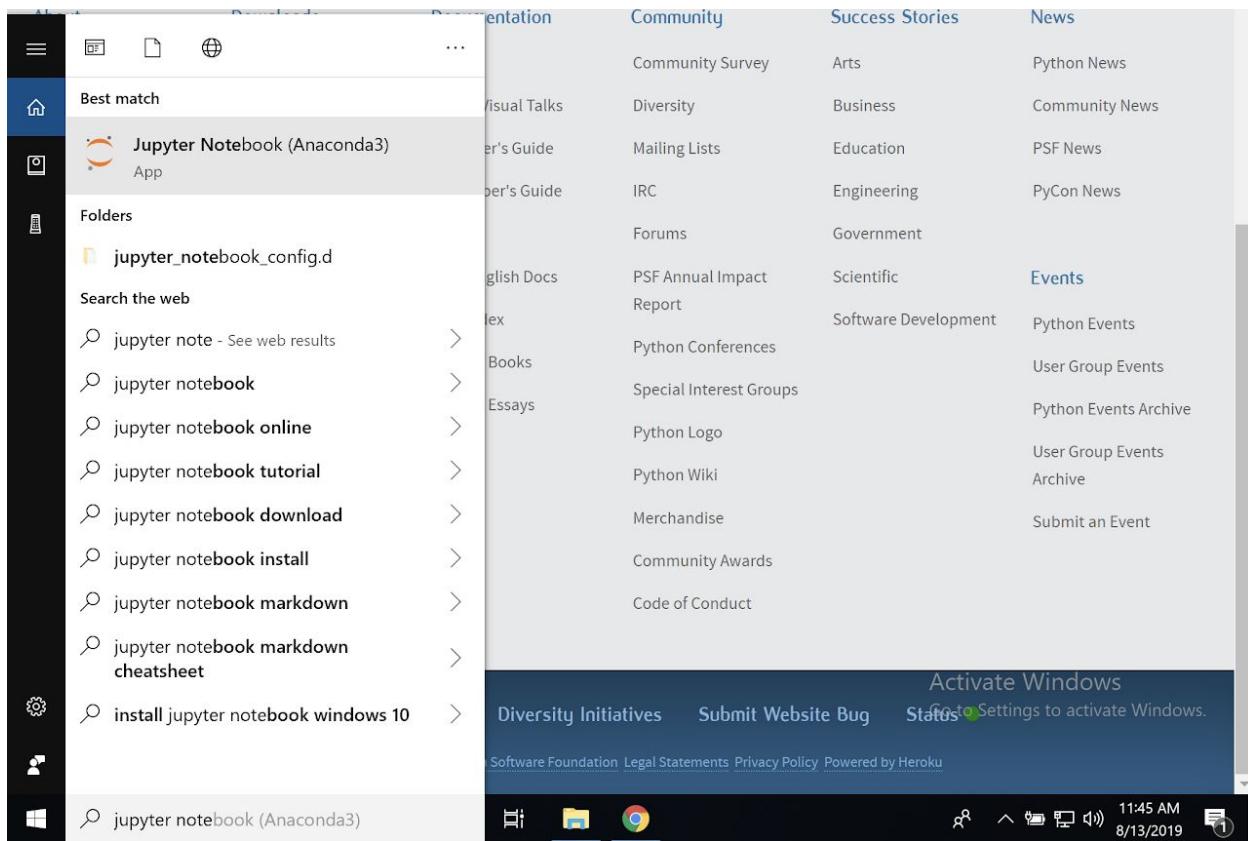


3. IPython Tutorial

In this class, we will use Python notebooks (more recently known as [Jupyter notebooks](#)) for the programming assignments. An IPython notebook lets you write and execute Python code in your web browser. IPython notebooks make it very easy to tinker with code and execute it in bits and pieces; for this reason IPython notebooks are widely used in scientific computing.

- Go to Window Home, type “Jupyter Notebook”

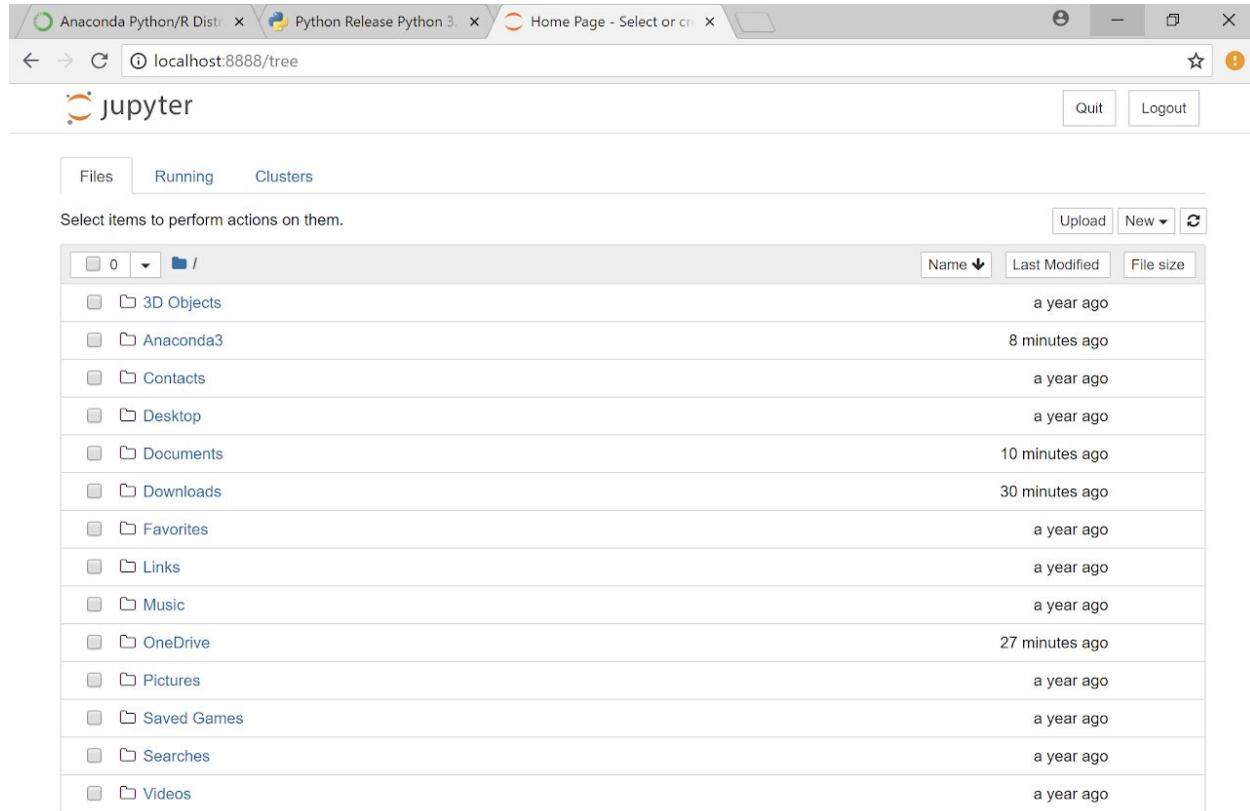
- Click “Jupyter Notebook (Anaconda 3)” (Jupyter Notebook comes in the Anaconda package)
- The jupyter server is launched automatically and you will see the black console



```
Jupyter Notebook (Anaconda3)
[I 11:46:18.279 NotebookApp] Writing notebook server cookie secret to C:\Users\Iris Liu\AppData\Roaming\jupyter\runtime\notebook_cookie_secret
[I 11:46:19.468 NotebookApp] JupyterLab extension loaded from C:\Users\Iris Liu\Anaconda3\lib\site-packages\jupyterlab
[I 11:46:19.469 NotebookApp] JupyterLab application directory is C:\Users\Iris Liu\Anaconda3\share\jupyter\lab
[I 11:46:19.472 NotebookApp] Serving notebooks from local directory: C:\Users\Iris Liu
[I 11:46:19.472 NotebookApp] The Jupyter Notebook is running at:
[I 11:46:19.472 NotebookApp] http://localhost:8888/?token=8a845324cee60485a69f1fa15a09129887ac76e683bc6db9
[I 11:46:19.472 NotebookApp] or http://127.0.0.1:8888/?token=8a845324cee60485a69f1fa15a09129887ac76e683bc6db9
[I 11:46:19.472 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 11:46:19.589 NotebookApp]

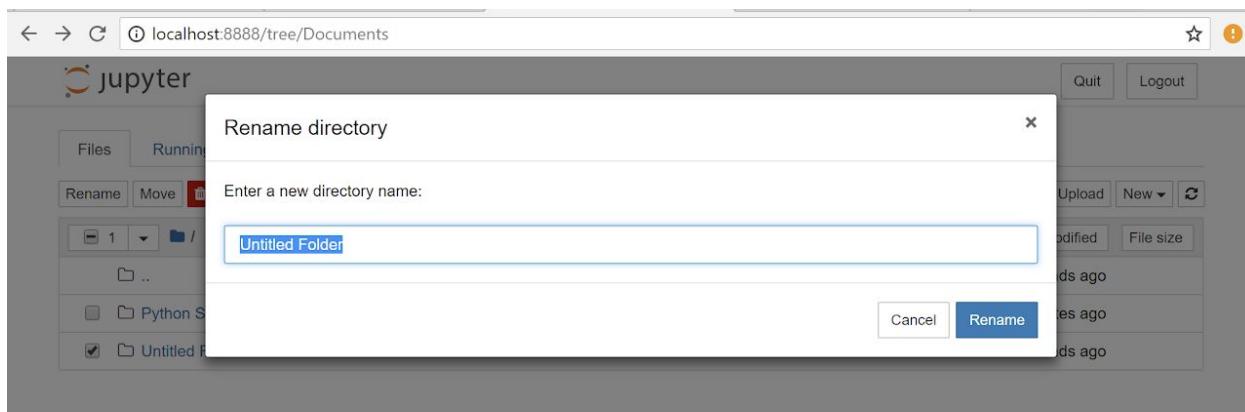
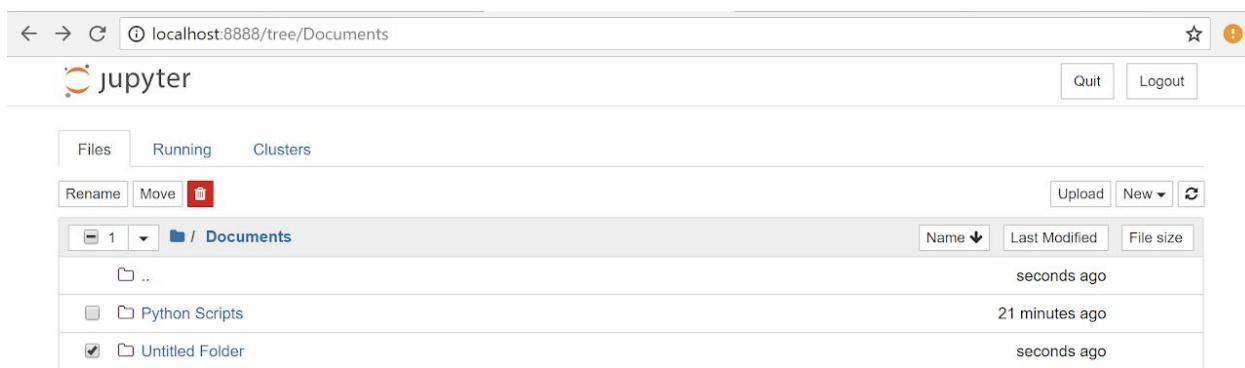
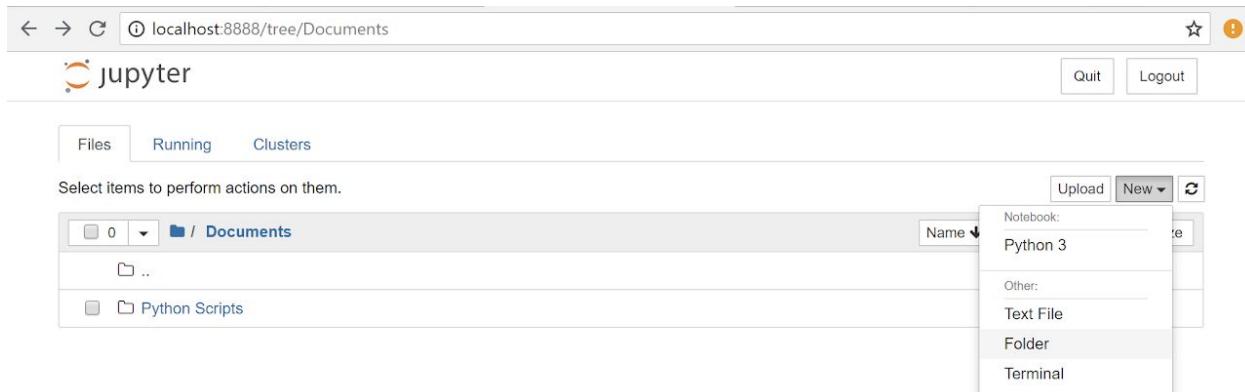
To access the notebook, open this file in a browser:
  file:///C:/Users/Iris%20Liu/AppData/Roaming/jupyter/runtime/nbserver-6160-open.html
Or copy and paste one of these URLs:
  http://localhost:8888/?token=8a845324cee60485a69f1fa15a09129887ac76e683bc6db9
  or http://127.0.0.1:8888/?token=8a845324cee60485a69f1fa15a09129887ac76e683bc6db9
[E 11:46:34.337 NotebookApp] Could not open static file ''
[W 11:46:35.664 NotebookApp] 404 GET /static/components/react/react-dom.production.min.js (::1) 134.99ms referer=http://localhost:8888/tree?token=8a845324cee60485a69f1fa15a09129887ac76e683bc6db9
```

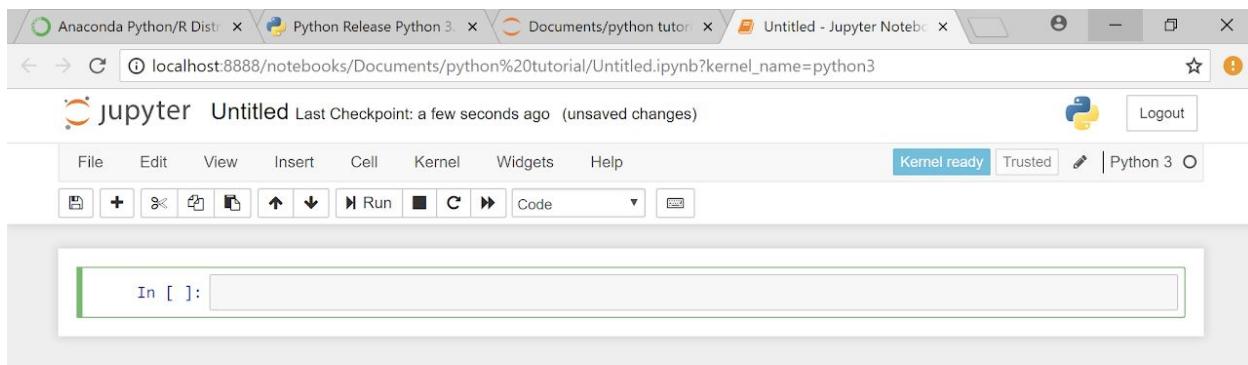
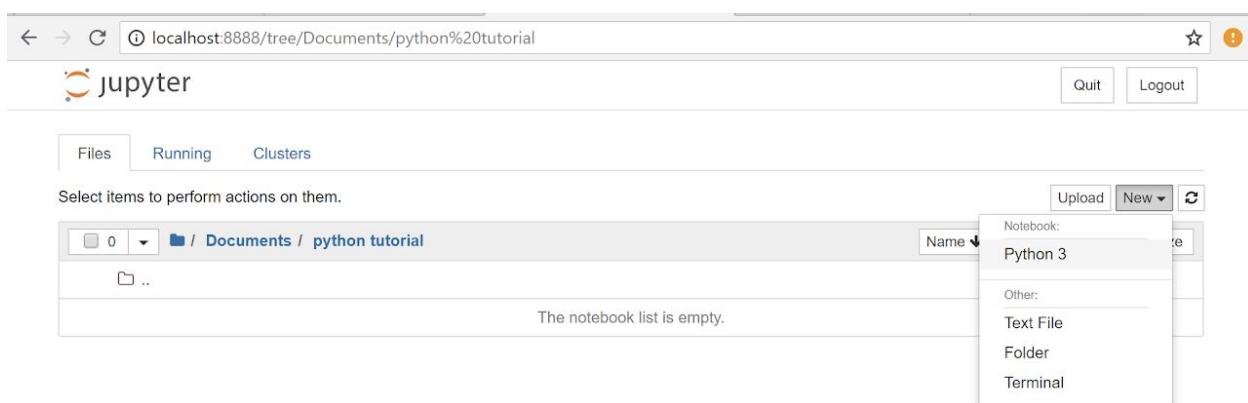
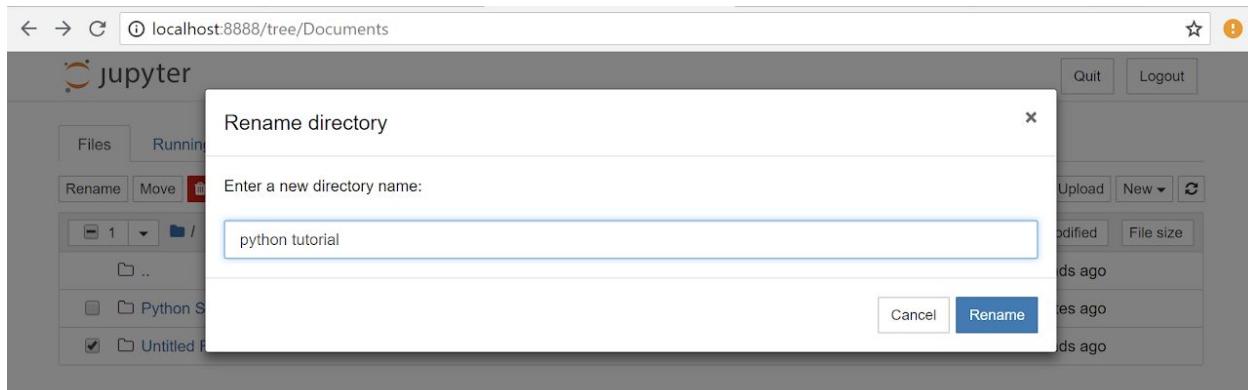
Once your notebook server is running, point your web browser at <http://localhost:8888> to start using your notebooks. If everything worked correctly, you should see a screen like this, showing all available IPython notebooks in the current directory:

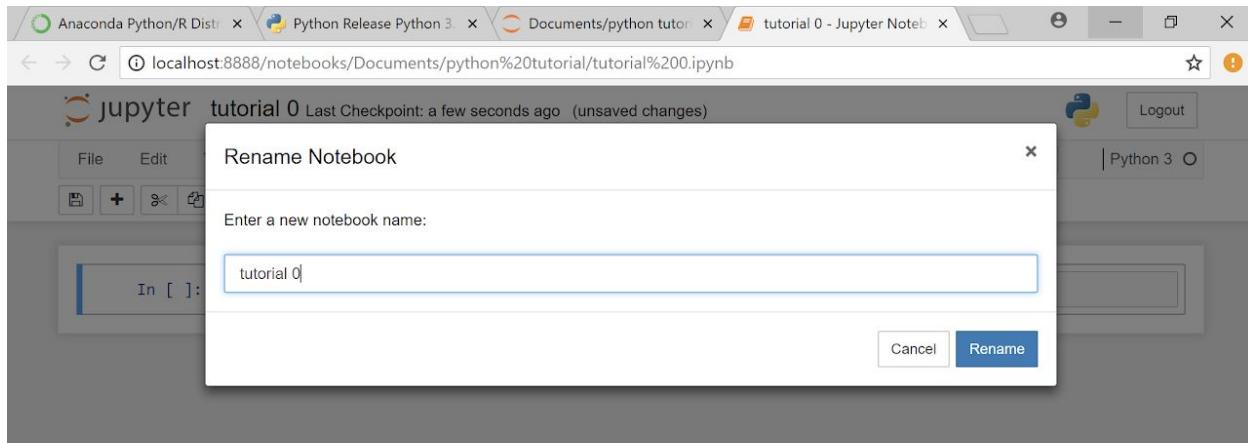


The landing page shows the file directory. Now we are going to open our tutorial folder and create a jupyter notebook file.

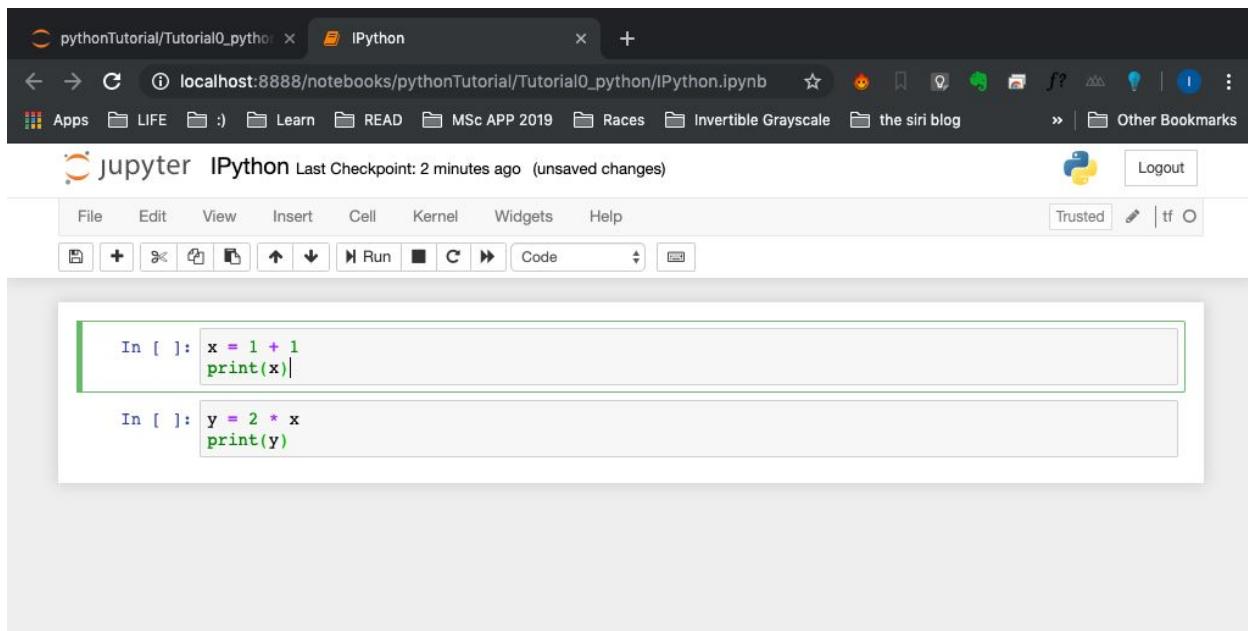
- Click “Documents”
- Click “New” > “Folder” on the right to open a new folder
- Check the box and click “rename” to rename the folder, such as “python tutorial”
- Enter “python tutorial”
- Click “New” > “Python 3” to create a new jupyter notebook document. A new tab will be launched
- Click “Untitled” on the top of the notebook to rename it to “tutorial 0”



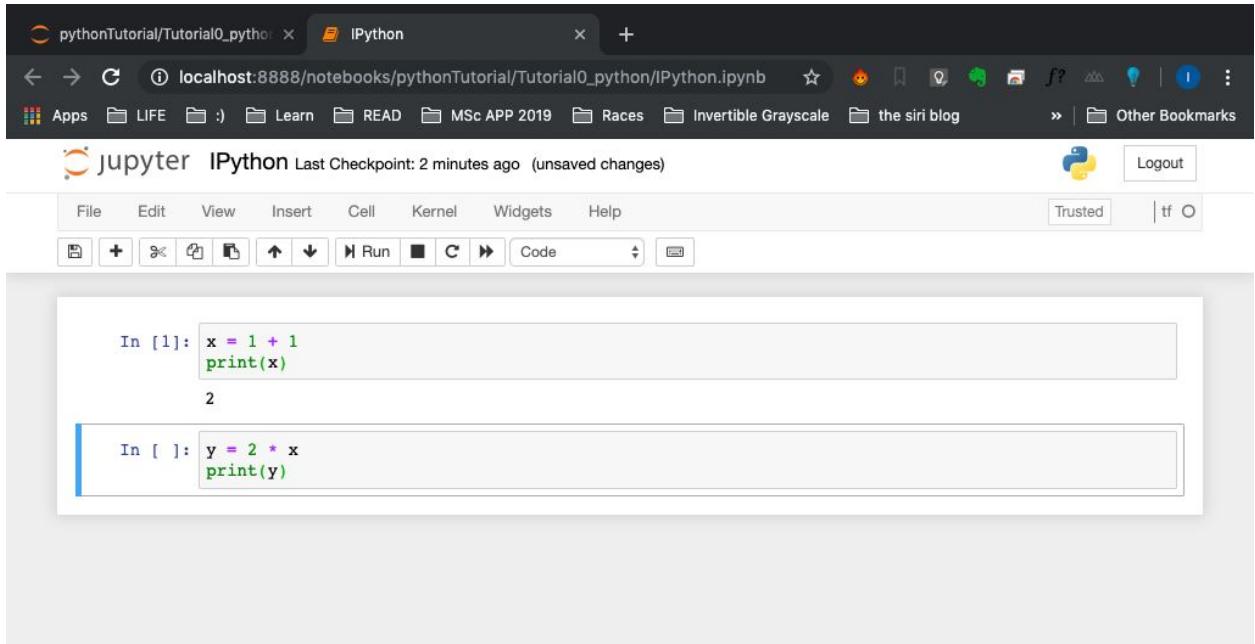




If you click through to a notebook file, you will see a screen like this:



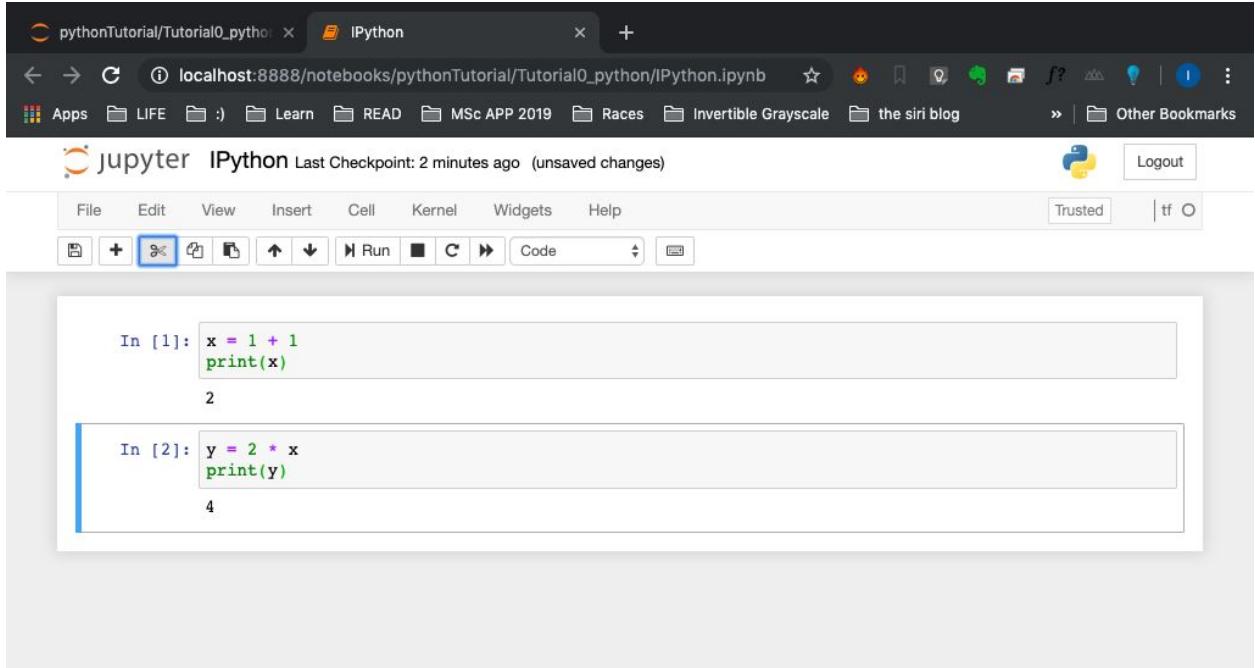
An IPython notebook is made up of a number of **cells**. Each cell can contain Python code. You can execute a cell by clicking on it and pressing Shift-Enter. When you do so, the code in the cell will run, and the output of the cell will be displayed beneath the cell. For example, after running the first cell the notebook looks like this:



```
In [1]: x = 1 + 1
        print(x)
        2

In [2]: y = 2 * x
        print(y)
```

Global variables are shared between cells. Executing the second cell thus gives the following result:



```
In [1]: x = 1 + 1
        print(x)
        2

In [2]: y = 2 * x
        print(y)
        4
```

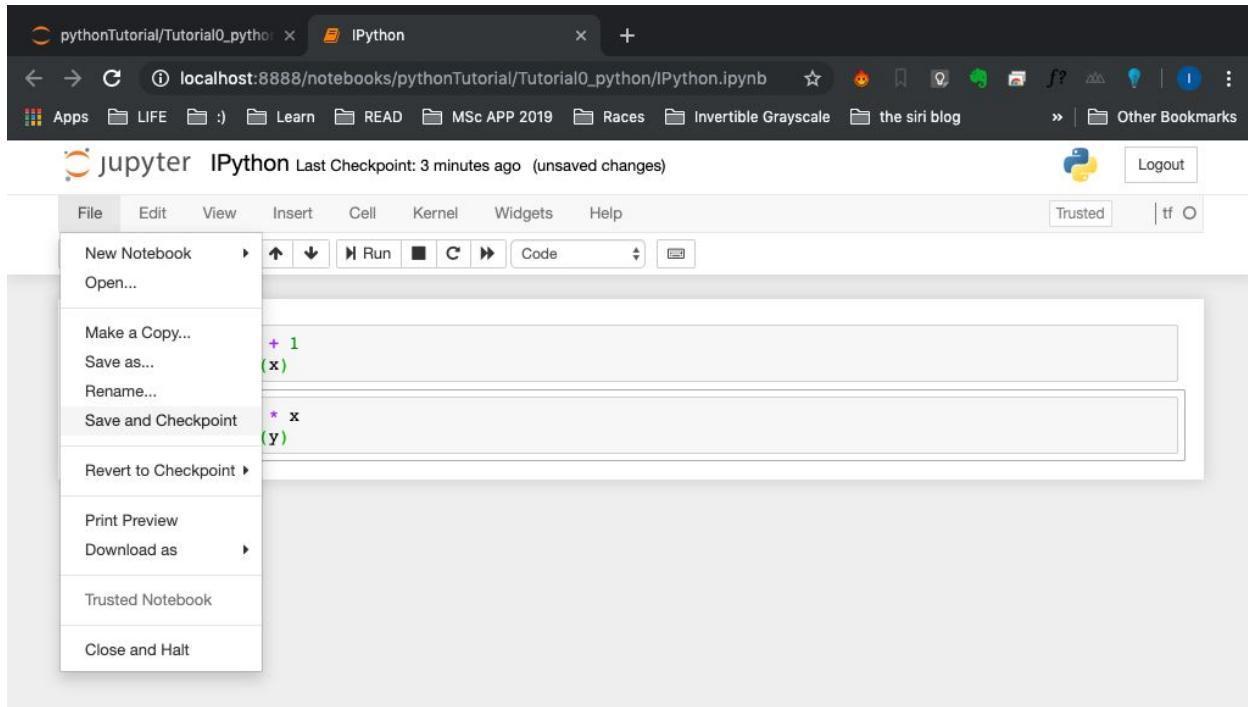
By convention, IPython notebooks are expected to be run from top to bottom. Failing to execute some cells or executing cells out of order can result in errors:

The screenshot shows a Jupyter IPython notebook interface. At the top, there's a toolbar with various icons. Below it is a header bar with the title "pythonTutorial/Tutorial0_python" and "IPython". The main area contains two code cells. The first cell, labeled "In []:", contains the code "x = 1 + 1" and "print(x)". The second cell, labeled "In [1]:", contains "y = 2 * x" and "print(y)". A blue vertical bar highlights the second cell. A red error message follows:

```
NameError                                 Traceback (most recent call last)
<ipython-input-1-b30c63918cd8> in <module>
----> 1 y = 2 * x
      2 print(y)

NameError: name 'x' is not defined
```

After you have modified an IPython notebook for one of the assignments by modifying or executing some of its cells, remember to **save your changes!**



This has only been a brief introduction to IPython notebooks, but it should be enough to get you up and running on the assignments for this course.

4. Python Basics

Open the notebook “*Python Basics With Numpy.ipynb*” from the downloaded folder in the designated location. The notebook will provide a step to step guidance about the basics of python as well as the numpy library.