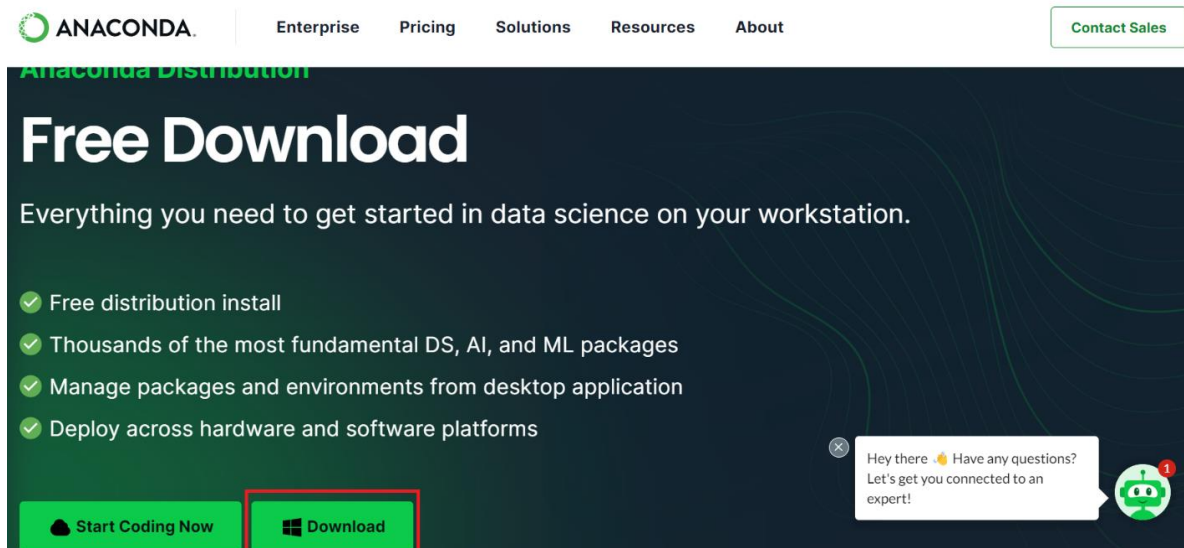**LAB MANUAL 1**

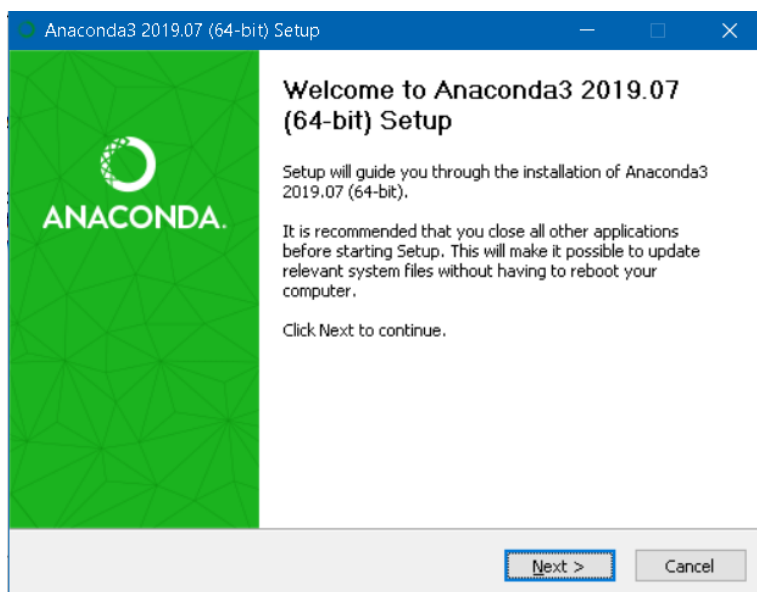# Demonstration of Anaconda Installation

Practical 1- Installing Anaconda on Windows
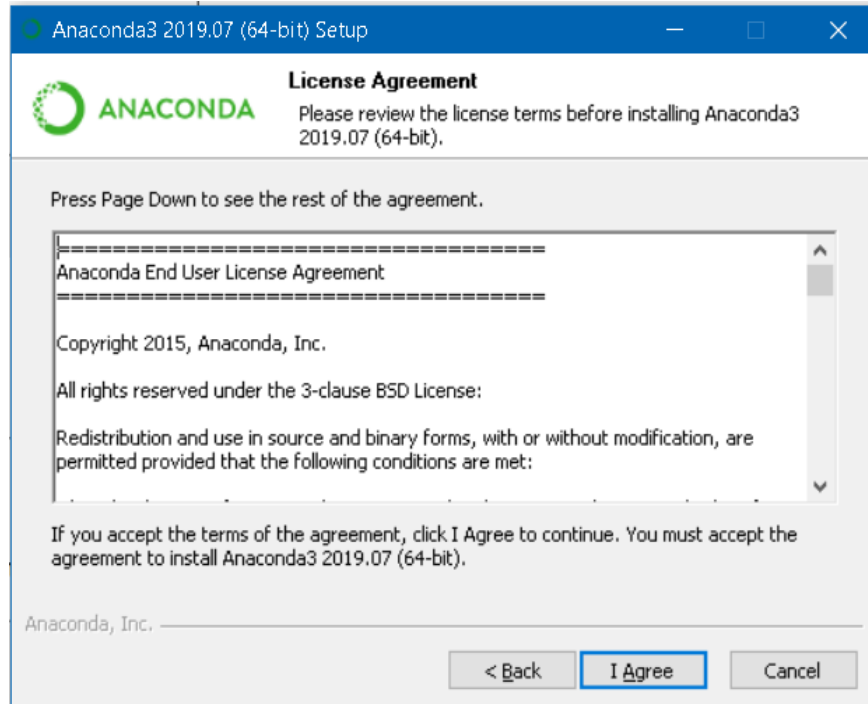
Step 1- At first, visit the following link: https://www.anaconda.com/download and the page will pop up like this, just click on Download.
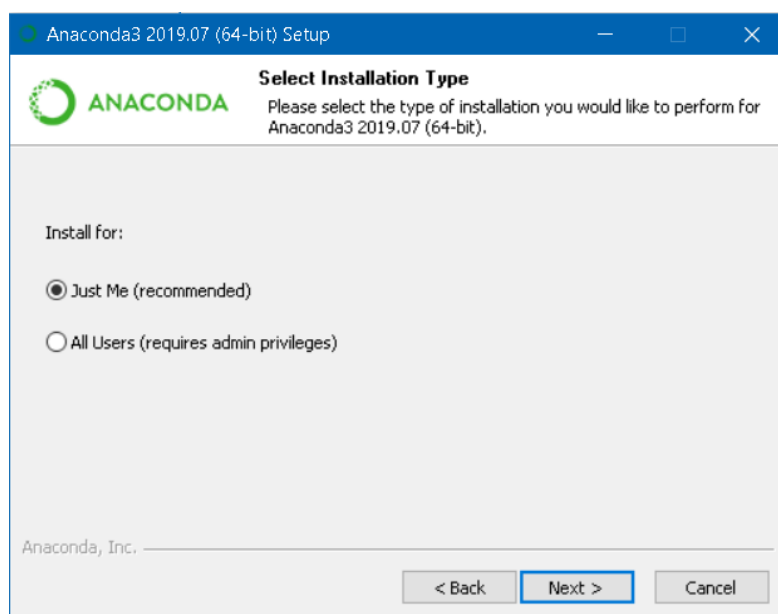

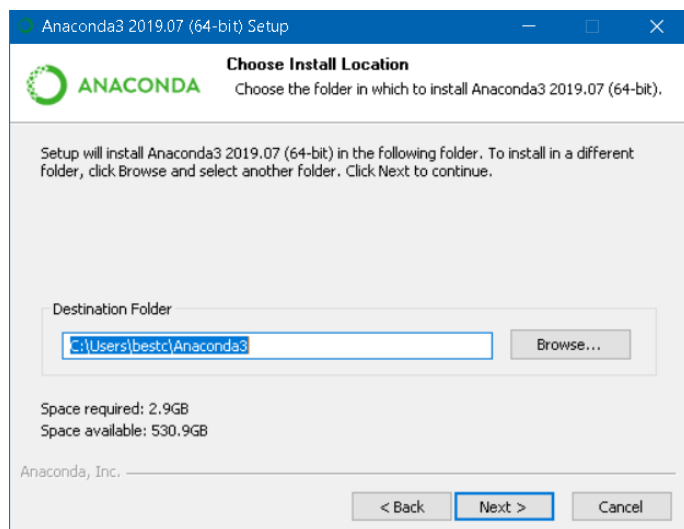**Step 2-** After downloading the file, run the file. The file will open, Click **Next**

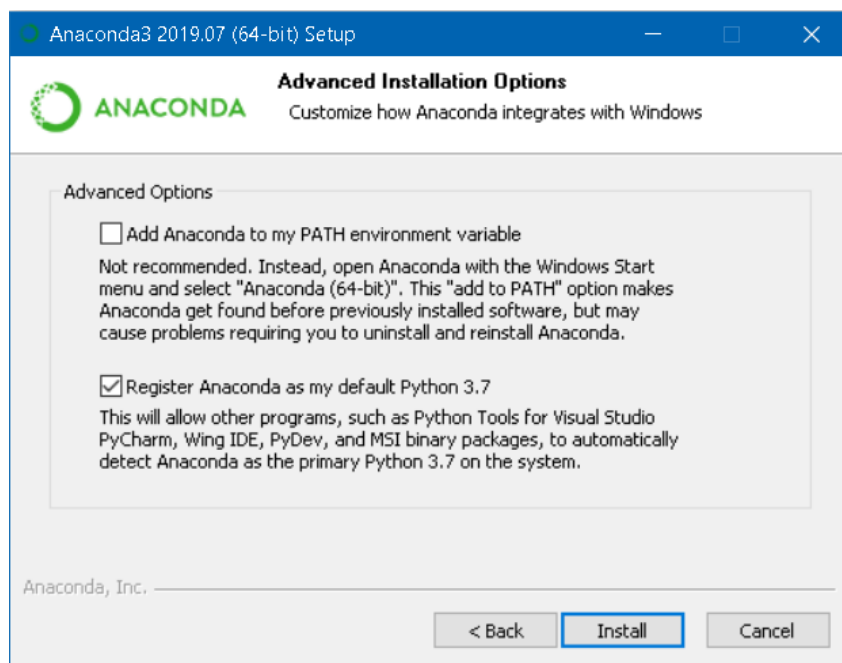**Step 3-** And click **I Agree** to the license.


Step 4- Choose Just Me and click Next.

Step 5- Choose the installation location by clicking **Browse or** leave it as it is (default location) and continue to click **Next**.

Step 6- Here, it is highly recommended to choose the second one "**Register Anaconda as my default Python 3.7**" and click **Install.**

Step 7- Once the installation is done, open the **Anaconda Prompt** from Windows start menu bar**.**
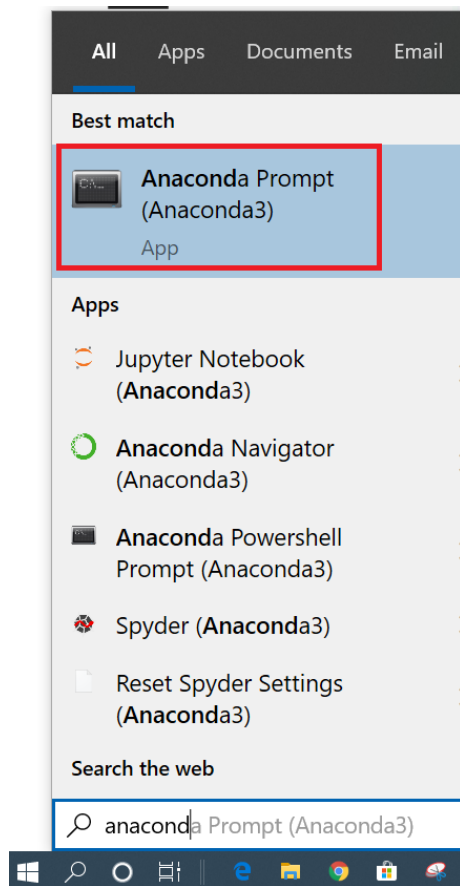
Step 8- Anaconda Prompt is shell similar to Windows Command Prompt (Windows Terminal) powered by Anaconda distribution. To check whether we have successfully installed Anaconda or not, type **python** command in the shell.

Practical 1- Installing Anaconda on Linux

Step 1- At first, visit the following link: https://www.anaconda.com/download and select the 64-

bit x86 download link as shown below.

Step 2: Verify integrity of the files(Optional step)

shasum -a 256 /PATH/TO/INSTALLER-FILENAME

Step 3: Install anaconda on your system by navigating to the folder where the Anaconda.sh file was downloaded and opening the folder in terminal and typing the following command

Step 4: Press enter to continue

Step 5: Scroll all the way down to accept the license agreement after carefully reading it by typing yes

Step 6: Next**,** press enter to confirm the default location for Anaconda installation.

Wait for the installation to finish.

Step 7: The terminal will as whether we would like to update our shell profile on start. Type yes to add conda to our shell profile.

```
mainak@mainak: ~/Downloads

done
installation finished.
Do you wish to update your shell profile to automatically initialize conda?
This will activate conda on startup and change the command prompt when activated.
If you'd prefer that conda's base environment not be activated on startup,
   run the following command when conda is activated:

conda config --set auto_activate_base false

You can undo this by running `conda init --reverse $SHELL`? [yes|no]
[no] >>>
```

Step 8: Close the terminal and open a new terminal (ctrl+alt+t) to check if anaconda was installed properly.

```
:~$ conda --version
conda 24.1.2
:~$ 
```

Step 9: Open Anaconda Navigator by typing the following command in the terminal.

anaconda-navigator

```
:~$ anaconda-navigator
Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland
 to run on Wayland anyway.
MESA-LOADER: failed to open zink: /usr/lib/dri/zink_dri.so: cannot open shared o
bject file: No such file or directory (search paths /usr/lib/x86_64-linux-gnu/dr
i:\$${ORIGIN}/dri:/usr/lib/dri, suffix _dri)
MESA-LOADER: failed to open swrast: /usr/lib/dri/swrast_dri.so: cannot open shar
ed object file: No such file or directory (search paths /usr/lib/x86_64-linux-gn
u/dri:\$${ORIGIN}/dri:/usr/lib/dri, suffix _dri)
```
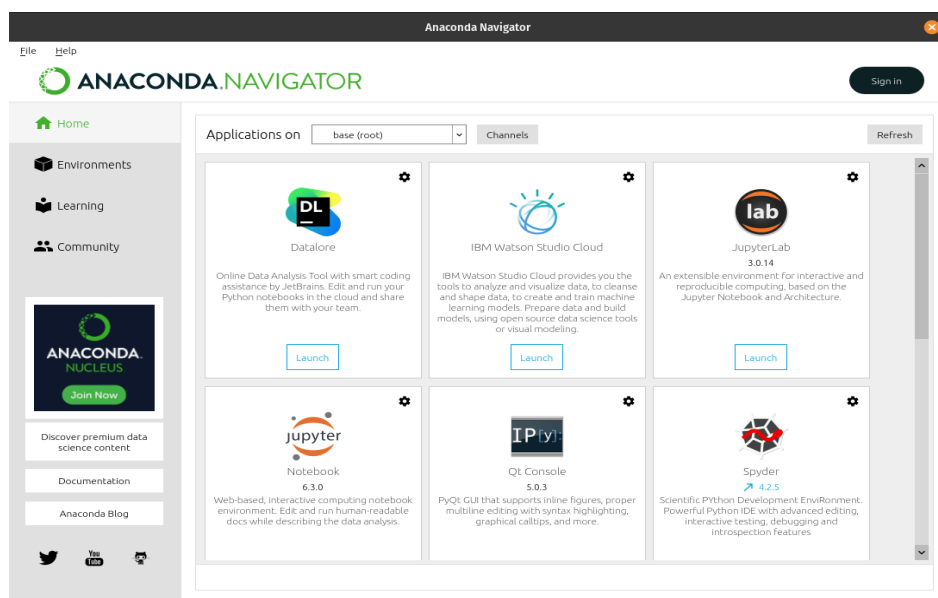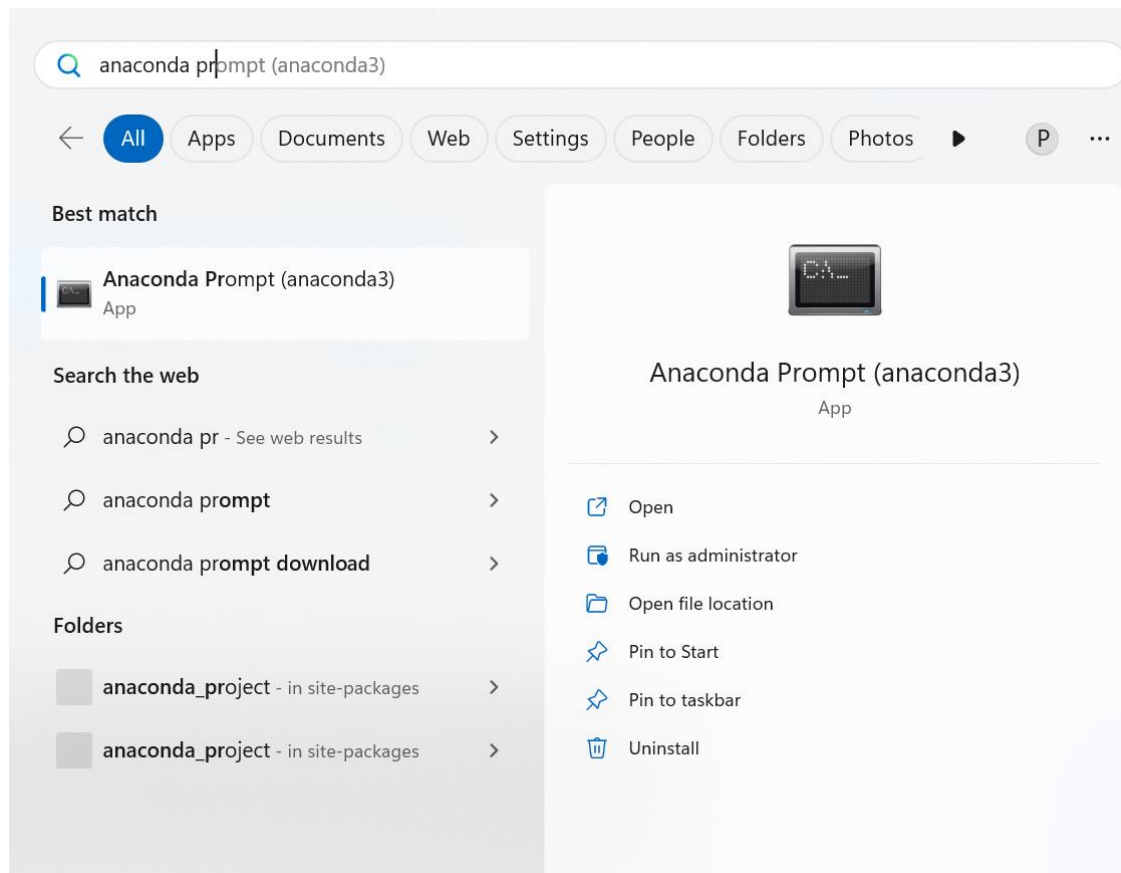
Practical 2– Getting Familiar with Jupyter Notebook

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.

If you already installed Anaconda in your machine, then it's very easy to use Jupyter notebook

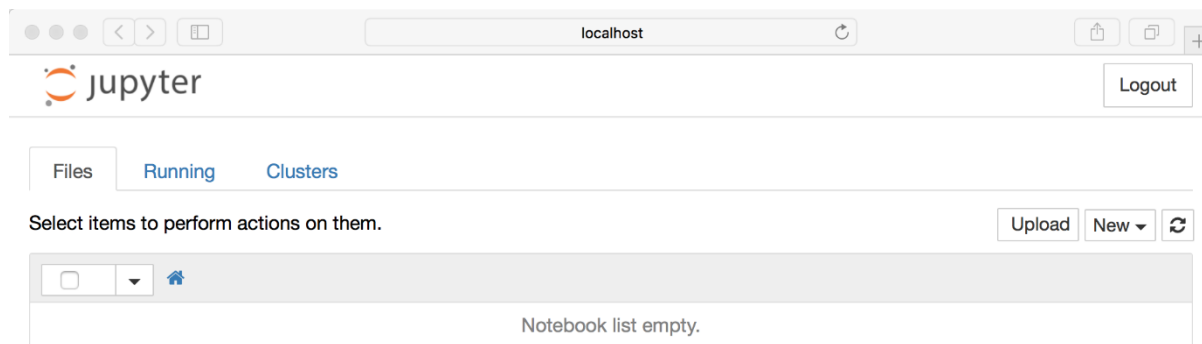**Step 1**- Press window key and Just type anaconda prompt and open.

Step 2- Just Run command Jupyter notebook and hit enter

Jupyter notebook will open in 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:

Step 3- To creating a notebook, Click on New and choose Python3

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

All you need to do is click on the *new* button (upper right), and it will open up a list of choices. Here choose python2 or Python 3, so we can create a Notebook that uses either of these. For simplicity's sake, let's choose Python 3.
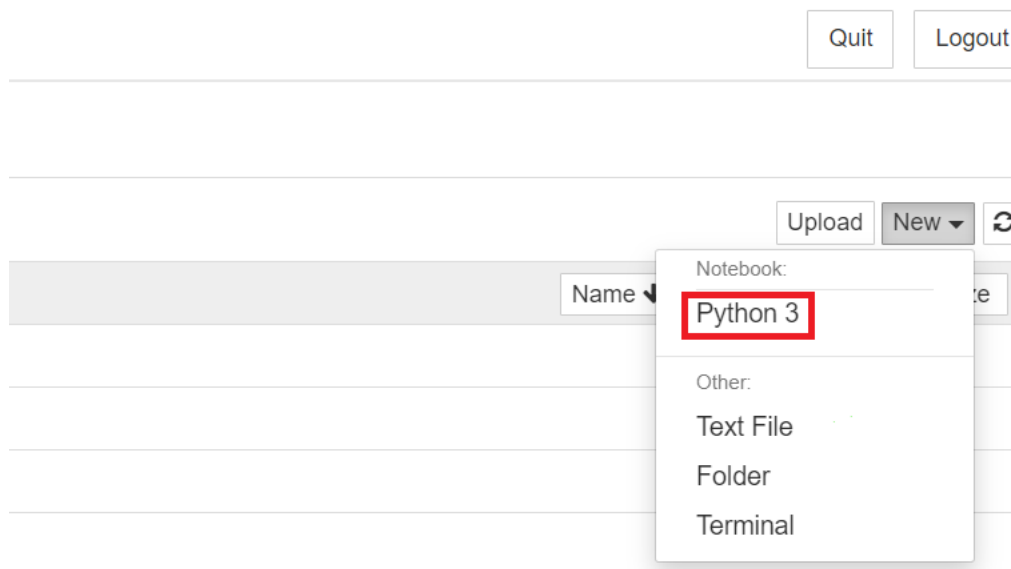
Your web page should now look like this:
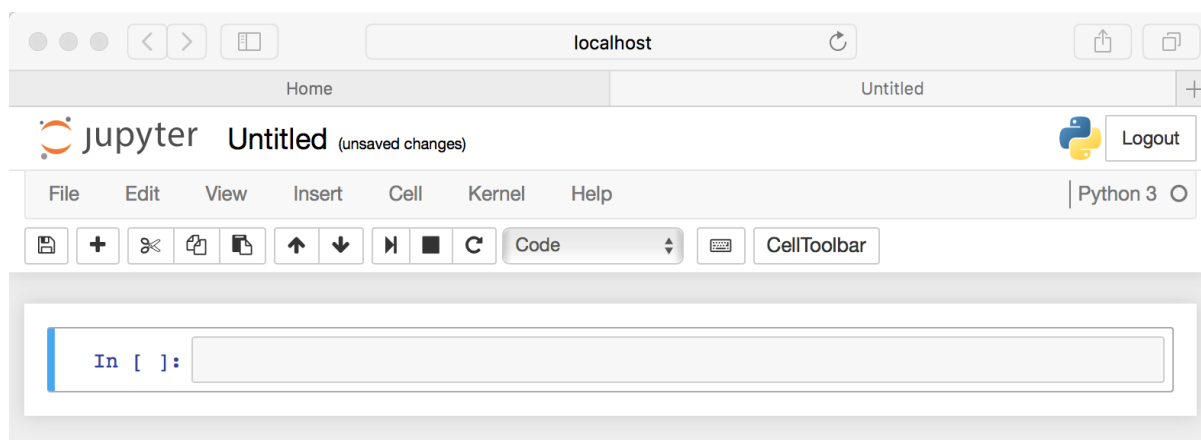
Step 4- Naming

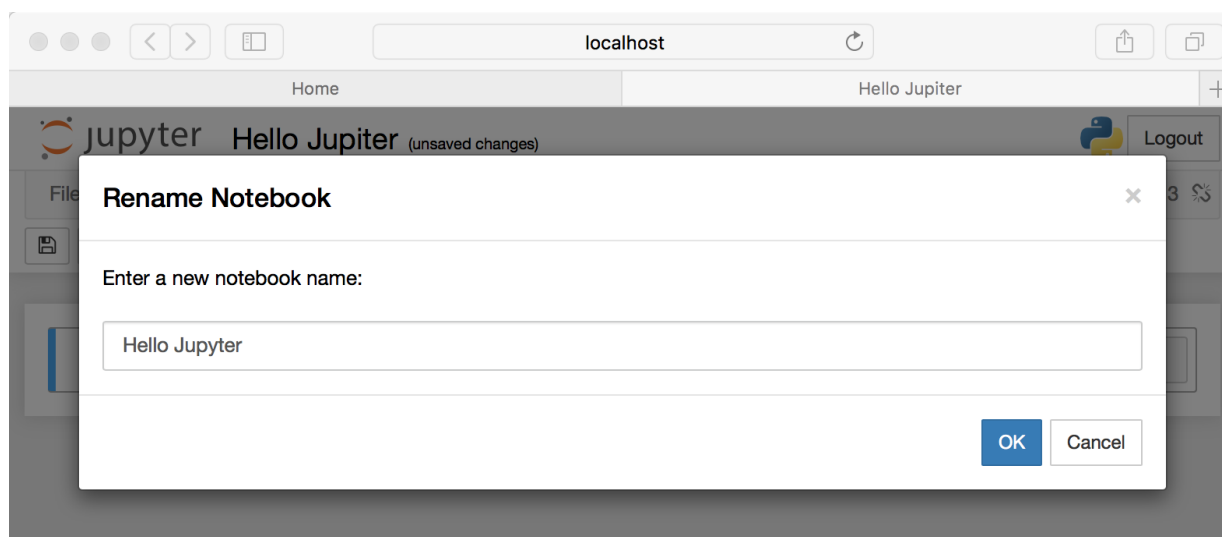You will notice that at the top of the page is the word Untitled. This is the title for the page and the name of your Notebook. Since that isn't a very descriptive name, let's change it!

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:

Step 5- Running Cells

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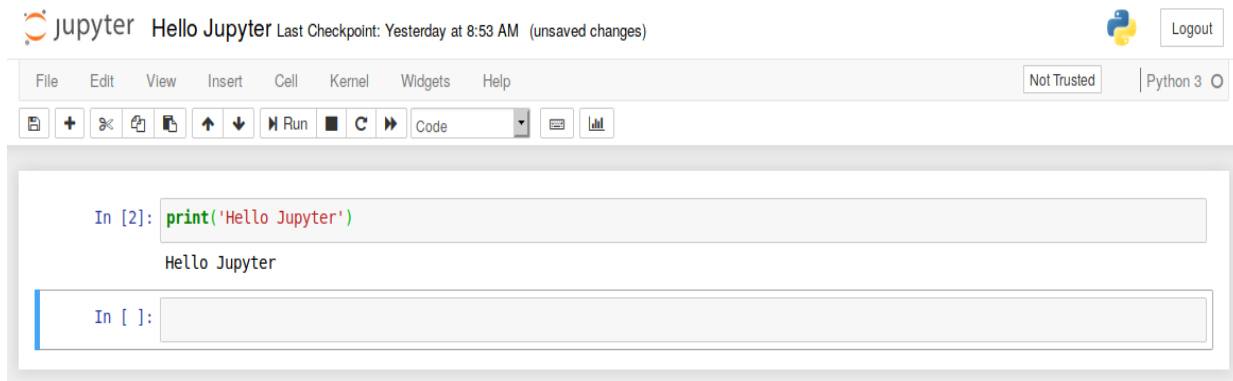
If you have multiple cells in your Notebook, and you run the cells in order, you can share your variables and imports across cells. This makes it easy to separate out your code into logical chunks without needing to reimport libraries or recreate variables or functions in every cell.