

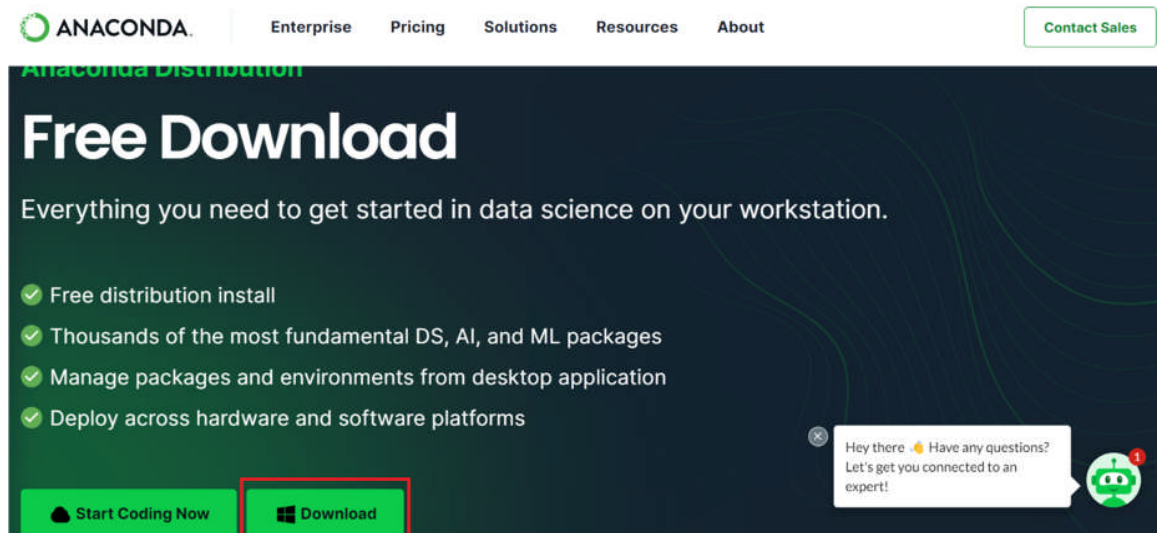


LAB MANUAL 1

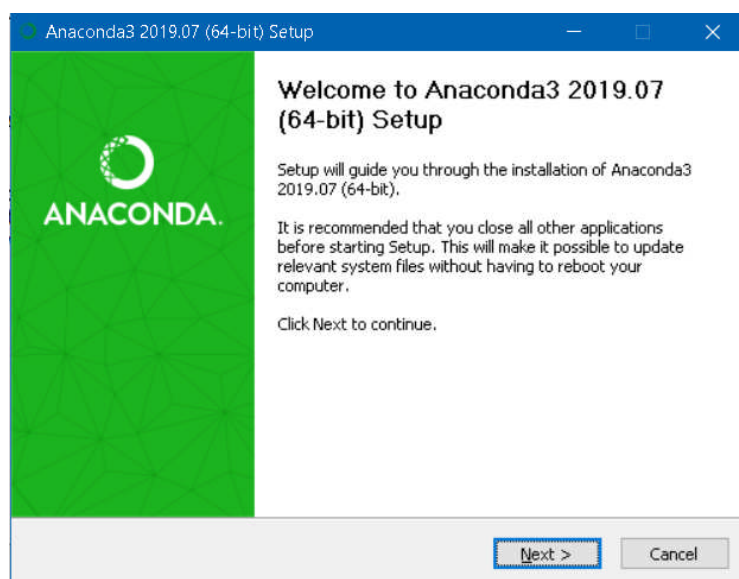
Demonstration of Anaconda Installation

Practical 1- Installing Anaconda on Windows

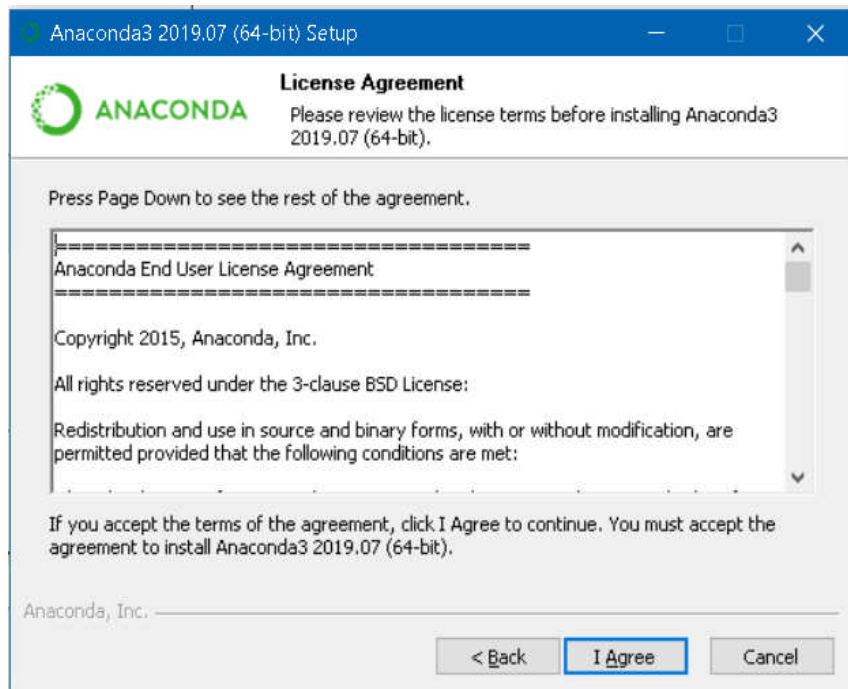
Step 1- At first, visit the following link: <https://www.anaconda.com/download> and the page will pop up like this, just click on Download.



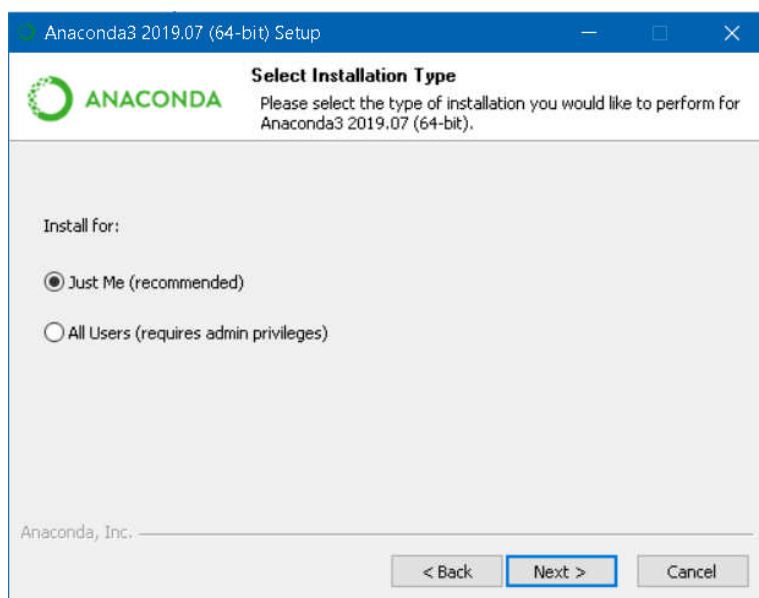
Step 2- After downloading the file, run the file. The file will open, Click **Next**



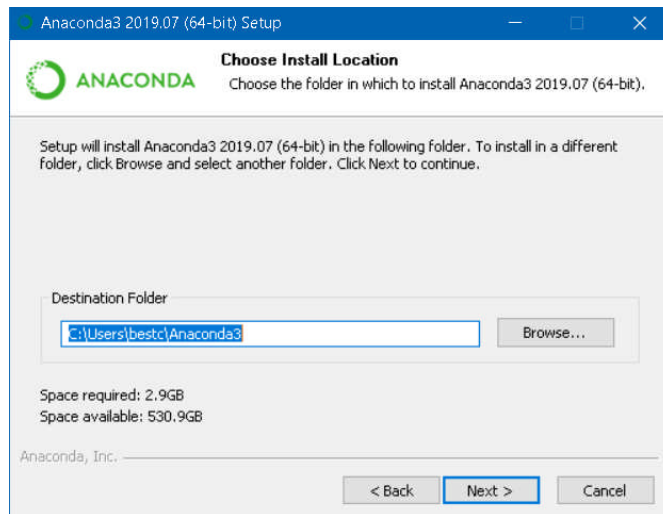
Step 3- And click **I Agree** to the license.



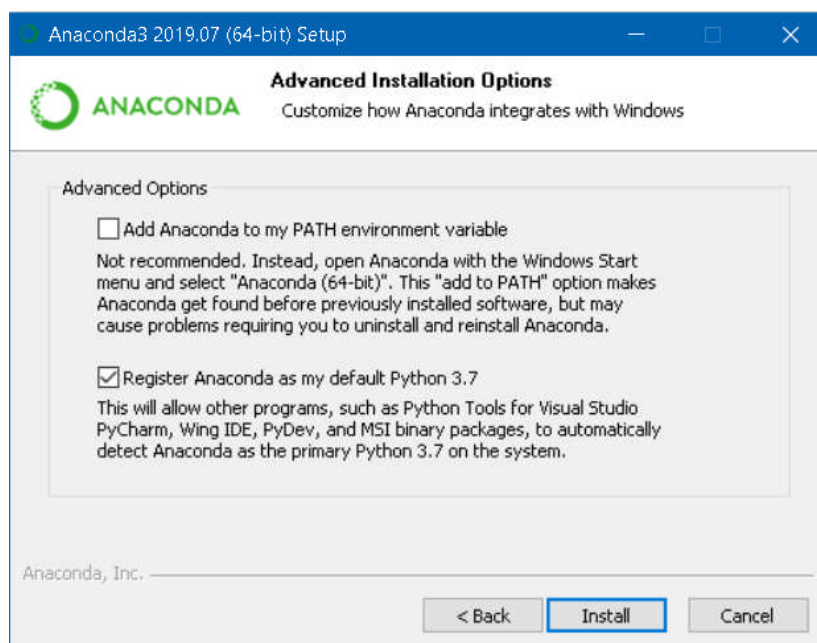
Step 4- Choose Just Me and click Next.



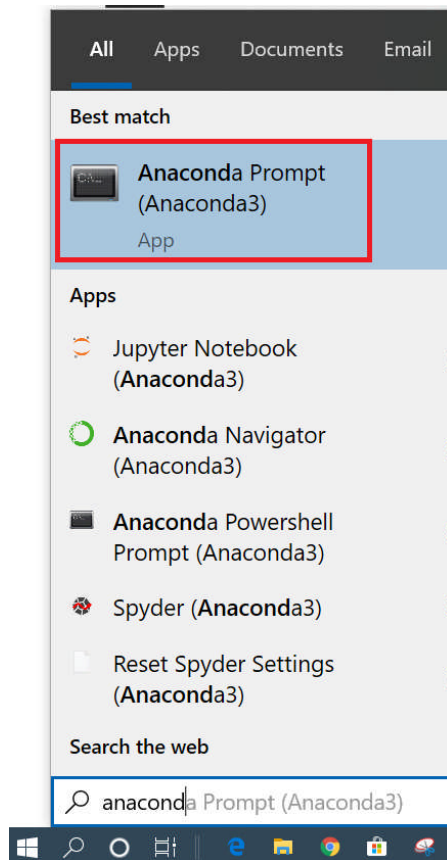
Step 5- Choose the installation location by clicking **Browse** or leave it as is (default location) and continue to click **Next**.



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.



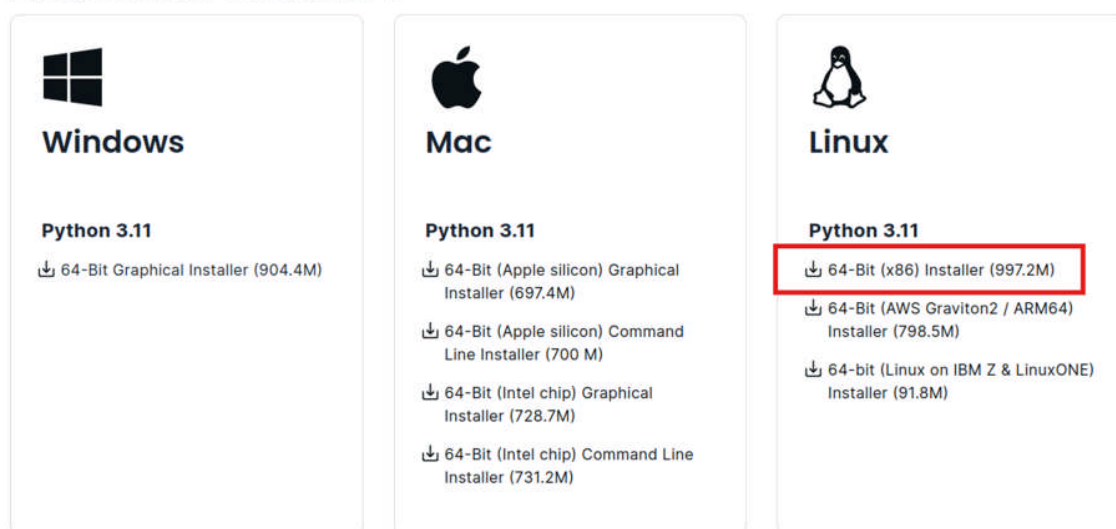
Step 8- Anaconda Prompt is shell similar to Windows Command Prompt (Windows Terminal) powered by Anaconda distribution. To check whether we have successfully installed Anaconda or not, type **python** command in the shell.

```
Anaconda Prompt (Anaconda3) - python
(base) C:\Users\besto>python
Python 3.7.3 (default, Apr 24 2019, 15:29:51) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```


Practical 1- Installing Anaconda on Linux

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

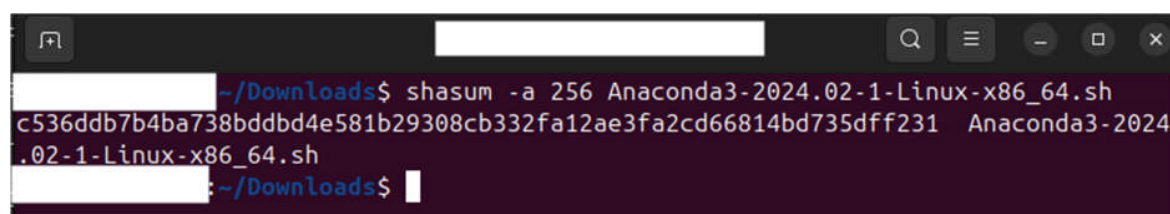
Anaconda Installers



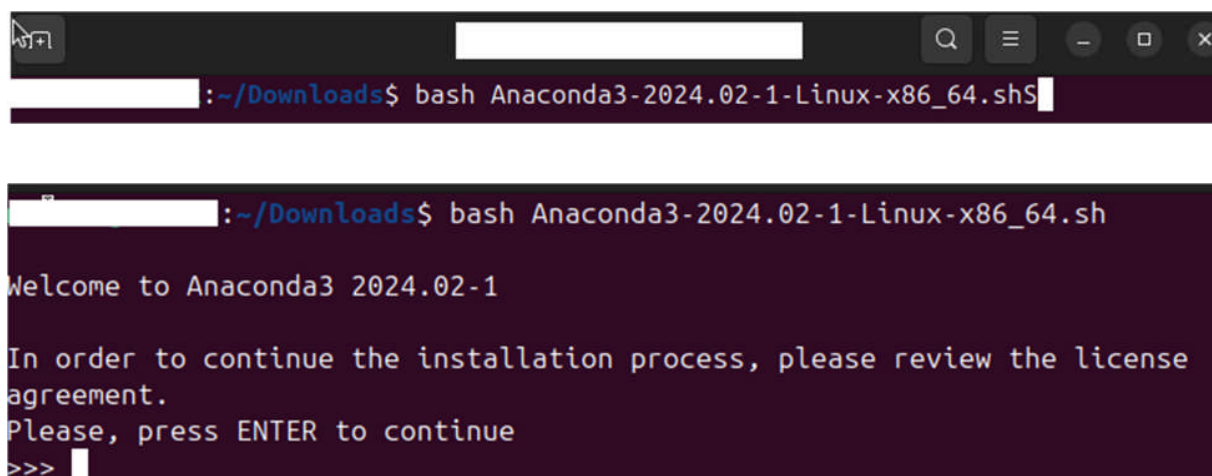
bit x86 download link as shown below.

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

A terminal window with a dark background and light text. The window title bar shows standard Linux window controls. The terminal content shows the execution of a shell script to install Anaconda3. The prompt is at the root directory in the Downloads folder. The output includes a welcome message and a license agreement screen where the user is prompted to press ENTER to continue. The cursor is visible at the end of the prompt line.

```
~/Downloads$ bash Anaconda3-2024.02-1-Linux-x86_64.sh

~/Downloads$ bash Anaconda3-2024.02-1-Linux-x86_64.sh

Welcome to Anaconda3 2024.02-1

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>> 
```

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


```
purchase orders, vendor forms, invoices, policies, confirmation, or similar form  
, even if signed by the Parties  
hereafter, will have no effect under this EULA. In the event of any conflict bet  
ween the terms of this EULA and the  
terms of any Order, the terms of this EULA will control unless otherwise explici  
tly set forth in an Order. This EULA may  
be executed in one or more counterparts, each of which will be an original, but  
taken together constituting one and the  
same instrument. Execution of a facsimile/electronic copy will have the same for  
ce and effect as execution of an  
original, and a facsimile/electronic signature will be deemed an original and va  
lid signature. No modification, consent  
or waiver under this EULA will be effective unless in writing and signed by both  
Parties. The failure of either Party to  
enforce its rights under this EULA at any time for any period will not be constr  
ued as a waiver of such rights. If any  
provision of this EULA is determined to be illegal or unenforceable, that provis  
ion will be limited or eliminated to the  
minimum extent necessary so that this EULA will otherwise remain in full force a  
nd effect and enforceable.  
  
Do you accept the license terms? [yes|no]  
>>> 
```

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

```
mainak@mainak: ~/Downloads  
  
Anaconda3 will now be installed into this location:  
/home/[redacted]/anaconda3  
  
- Press ENTER to confirm the location  
- Press CTRL-C to abort the installation  
- Or specify a different location below  
  
[/home/[redacted]/anaconda3] >>> 
```

Wait for the installation to finish.

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

A terminal window titled 'mainak@mainak: ~/Downloads' with a dark background. The text inside shows the end of a conda installation. It says 'done' and 'installation finished.' followed by a question: '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:'. Below this is the command 'conda config --set auto_activate_base false'. Then it asks 'You can undo this by running `conda init --reverse \$SHELL`? [yes|no]' and the user has responded with '[no] >>>'.

```
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] >>>
```

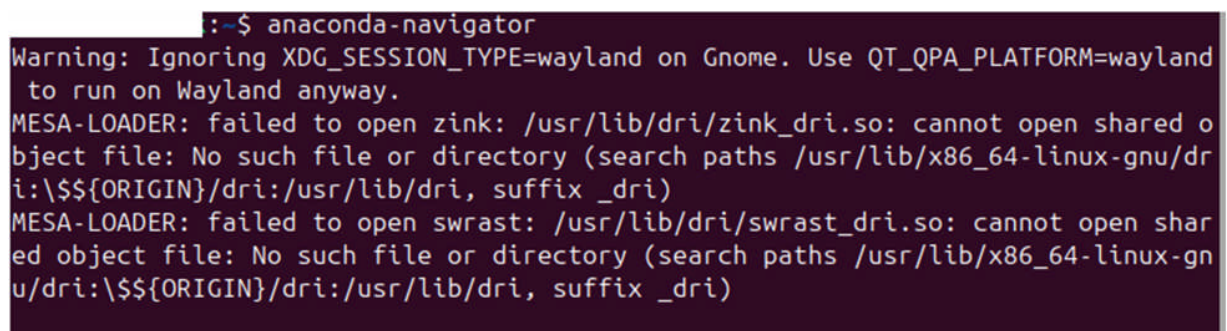
Step 8: Close the terminal and open a new terminal (ctrl+alt+t) to check if anaconda was installed properly.

A terminal window with a dark background. The user has typed 'conda --version' and the output is 'conda 24.1.2'. The prompt is now ':-\$'.

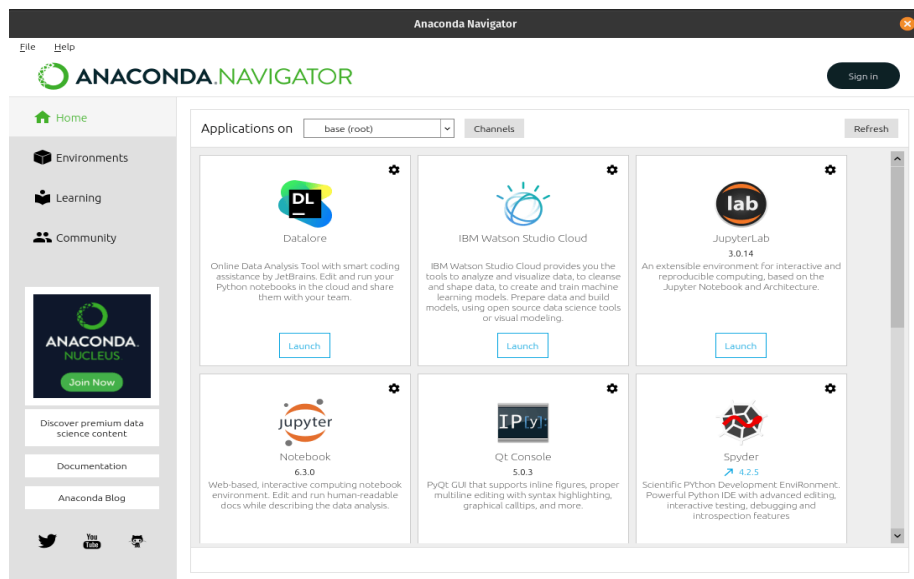
```
:-$ conda --version
conda 24.1.2
:-$
```

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

anaconda-navigator

A terminal window with a dark background. The user has typed 'anaconda-navigator'. The output shows a warning about XDG_SESSION_TYPE=wayland on Gnome, followed by two MESA-LOADER error messages about failed to open zink and swrast drivers. The prompt is now ':-\$'.

```
:-$ 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)
:-$
```

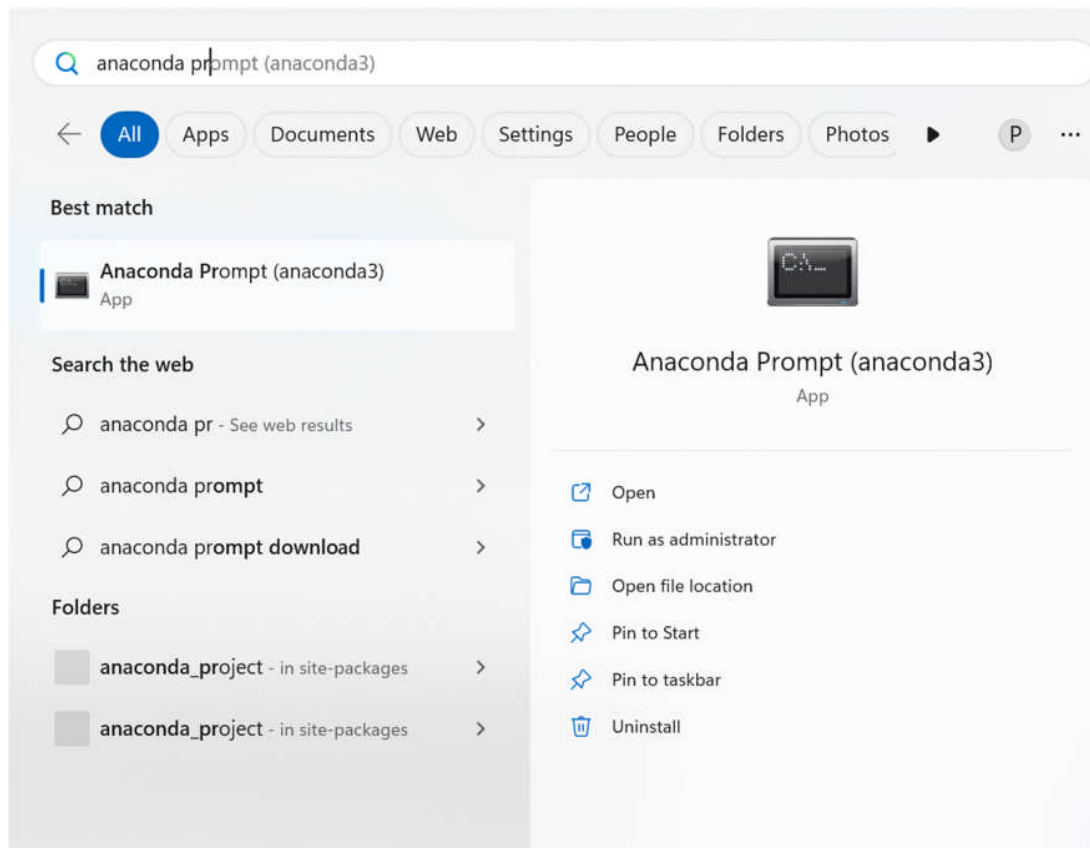


Practical 2– Getting Familiar with Jupyter Notebook

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](https://projectjupyter.org/).

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.



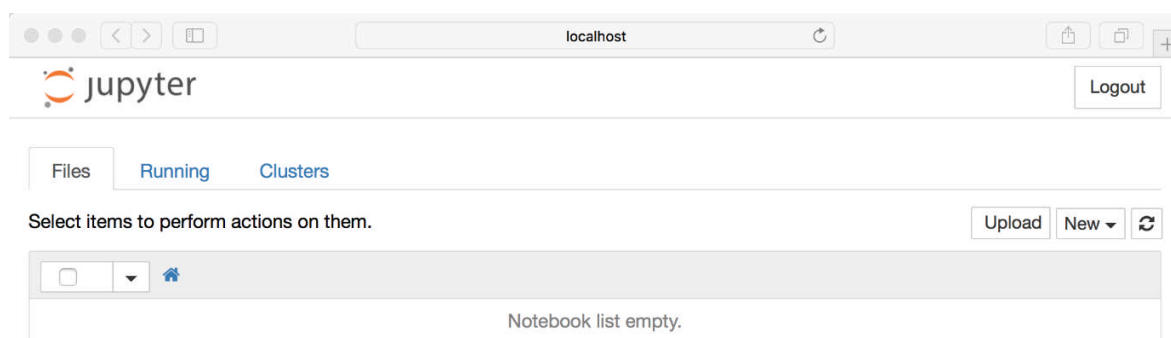
Step 2- Just Run command Jupyter notebook and hit enter

```
Anaconda Prompt (Anaconda3) - Jupyter notebook
(base) C:\Users\PRAVIN>Jupyter notebook
[I 14:30:04.695 NotebookApp] JupyterLab extension loaded from C:\ProgramData\Anaconda3\lib\site-packages\jupyterlab
[I 14:30:04.695 NotebookApp] JupyterLab application directory is C:\ProgramData\Anaconda3\share\jupyter\lab
[I 14:30:04.698 NotebookApp] Serving notebooks from local directory: C:\Users\PRAVIN
[I 14:30:04.698 NotebookApp] Jupyter Notebook 6.1.4 is running at:
[I 14:30:04.698 NotebookApp] http://localhost:8888/?token=fce81d78fb022669006757133ffae92129775d35581a8513
[I 14:30:04.699 NotebookApp] or http://127.0.0.1:8888/?token=fce81d78fb022669006757133ffae92129775d35581a8513
[I 14:30:04.699 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 14:30:04.796 NotebookApp]

To access the notebook, open this file in a browser:
    file:///C:/Users/PRAVIN/AppData/Roaming/jupyter/runtime/nbserver-11204-open.html
Or copy and paste one of these URLs:
    http://localhost:8888/?token=fce81d78fb022669006757133ffae92129775d35581a8513
    or http://127.0.0.1:8888/?token=fce81d78fb022669006757133ffae92129775d35581a8513
```

Jupyter notebook will open in your default browser, should start (or open a new tab) to the following URL: <http://localhost:8888/tree>

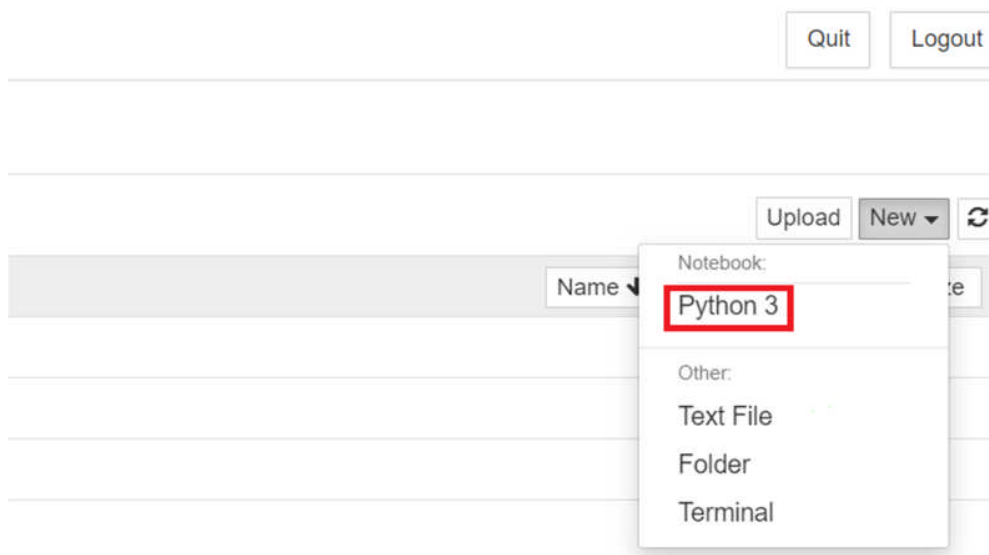
Your browser should now look something like this:



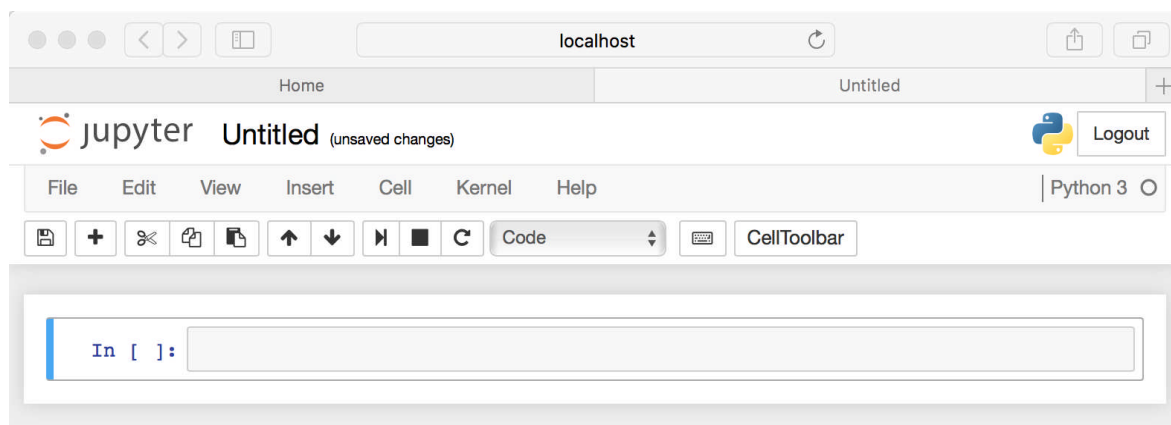
Step 3- To creating a notebook, Click on New and choose Python3

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.



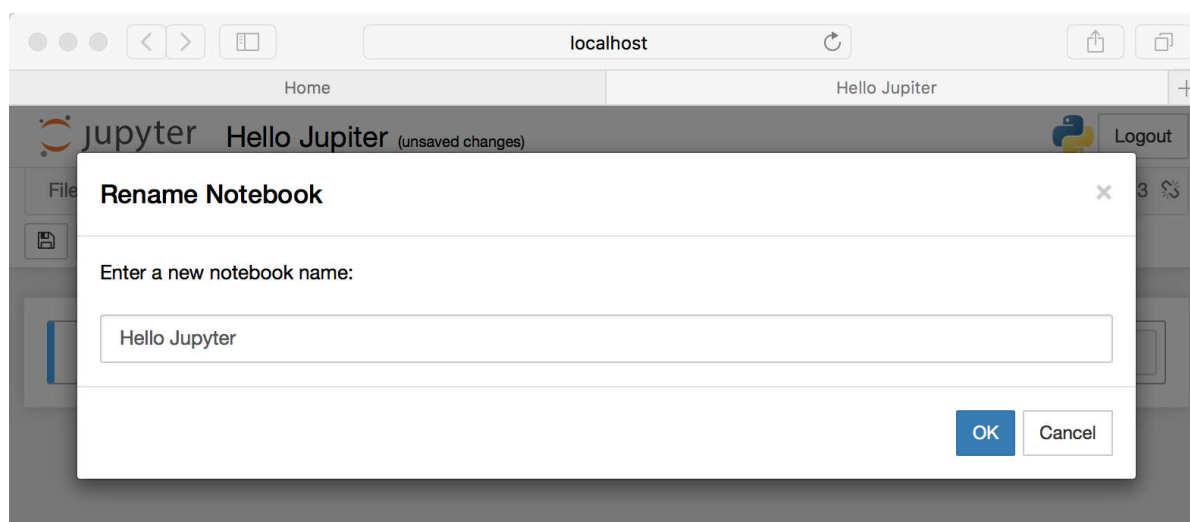
Your web page should now look like this:



Step 4- Naming

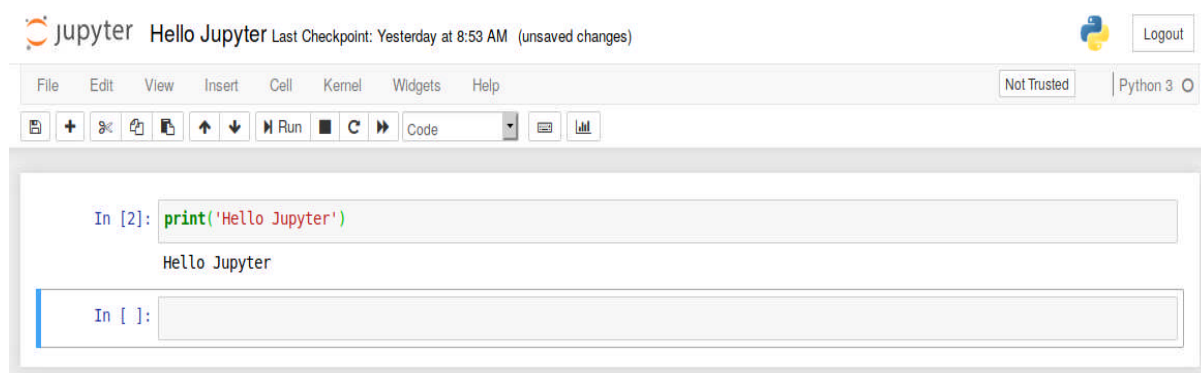
You will notice that at the top of the page is the word **Untitled**. This is the title for the page and the name of your Notebook. Since that isn't a very descriptive name, let's change it!

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.