

Running Jupyter Notebook on a Remote Server and accessing it using your local machine

Installing conda on your remote machine.

1. Login to your Remote Machine the usual way you do using SSH tunnel. Download and install conda from [Anaconda](#).
2. Once you have downloaded conda, you will have to set it up in linux using the below command.

```
conda init
```

3. Create a new conda environment using the below command.

```
# Replace X with the appropriate python version you need.  
conda create -n your_env_name python=3.X  
# You may omit the python= if you like the latest version.
```

4. Activate this environment using the below command.

```
conda activate your_env_name
```

5. Once the environment is activated, your `python` and `pip` commands would be remapped to the anaconda versions. You may check this using `which python` and `which pip`.

Setting Jupyter Notebook on a Remote Server and accessing it using your system will provide you a GUI interface to interact with your Remote Server and use its resources.

1. Login to your Remote Machine the usual way you do using SSH tunnel. Create and activate virtual environment on your Remote Server.

```
# Replace <REMOTE_USER> with the remote server username.  
# Replace <REMOTE_HOST> with your remote server address.  
# Replace <ENV NAME> with the environment name of your choice.  
  
ssh <REMOTE_USER>@<REMOTE_HOST>  
conda create --name <ENV NAME>  
conda activate <ENV NAME>
```

2. Install Jupyter Notebook on your Remote Server (HPC, Phantom etc.)

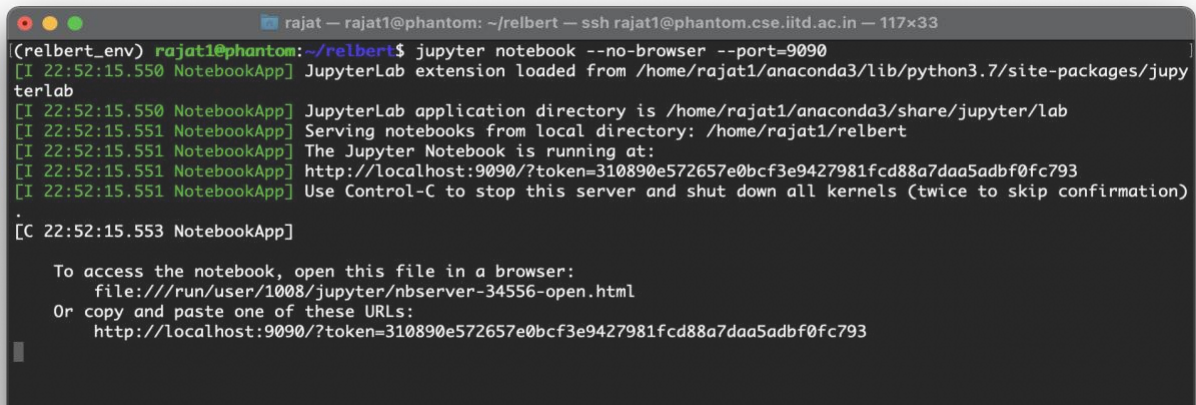
```
pip install jupyter
```

3. Launch Jupyter Notebook on your remote server, Select a port number for <PORT>

```
# Replace <PORT> with port number of your choice.  
# --no-browser will starts the notebook without opening a browser.  
  
jupyter notebook --no-browser --port=<PORT>
```

You can launch the Jupyter Notebook using default port. Note if that default port is not free then it will move to next available port and will print the port it is using.

```
# Default port for your notebook is 8888. If it is occupied then it will find the next  
available port.  
  
jupyter notebook --no-browser
```



```
rajat — rajat1@phantom: ~/relbert — ssh rajat1@phantom.cse.iitd.ac.in — 117x33  
(relbert_env) rajat1@phantom:~/relbert$ jupyter notebook --no-browser --port=9090  
[I 22:52:15.550 NotebookApp] JupyterLab extension loaded from /home/rajat1/anaconda3/lib/python3.7/site-packages/jupyterlab  
[I 22:52:15.550 NotebookApp] JupyterLab application directory is /home/rajat1/anaconda3/share/jupyter/lab  
[I 22:52:15.551 NotebookApp] Serving notebooks from local directory: /home/rajat1/relbert  
[I 22:52:15.551 NotebookApp] The Jupyter Notebook is running at:  
[I 22:52:15.551 NotebookApp] http://localhost:9090/?token=310890e572657e0bcf3e9427981fcd88a7daa5adb0fc793  
[I 22:52:15.551 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation)  
  
[C 22:52:15.553 NotebookApp]  
  
To access the notebook, open this file in a browser:  
file:///run/user/1008/jupyter/nbserver-34556-open.html  
Or copy and paste one of these URLs:  
http://localhost:9090/?token=310890e572657e0bcf3e9427981fcd88a7daa5adb0fc793
```

You will get this output after running the above command

4. Access the remote notebook from your machine over SSH by setting up a SSH tunnel.

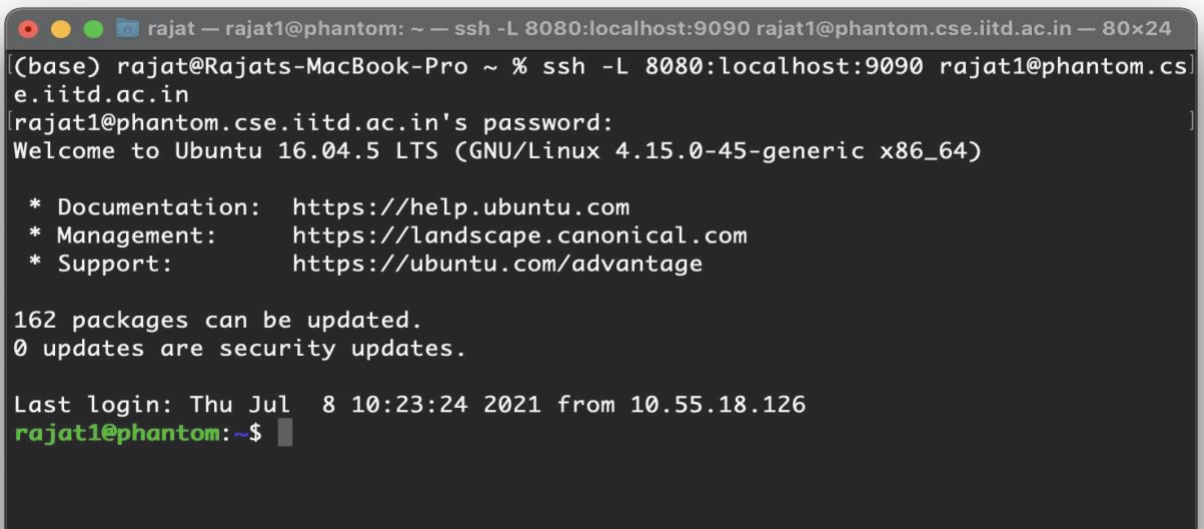
```
# Replace <PORT> with the port number you have selected in your above step.  
# Replace <REMOTE_USER> with the remote server username.  
# Replace <REMOTE_HOST> with your remote server address.  
  
ssh -L 8080:localhost:<PORT> <REMOTE_USER>@<REMOTE_HOST>
```

Access the remote notebook from your machine over SSH by setting up a SSH tunnel in background.

```
# Replace <PORT> with the port number you have selected in your above step.  
# Replace <REMOTE_USER> with the remote server username.  
# Replace <REMOTE_HOST> with your remote server address.  
  
ssh -N -f -L 8080:localhost:<PORT> <REMOTE_USER>@<REMOTE_HOST>
```

```
# -f : this request the ssh command to go to the background before execution.  
# In your Remote Server the notebook is running at the port <PORT> but in your machine that  
# notebook is forwarded to port 8080. Thus you can access the remote notebook at your machine  
# using 8080 port.  
# Instead of 8080 you can use any port that is available in your system.
```

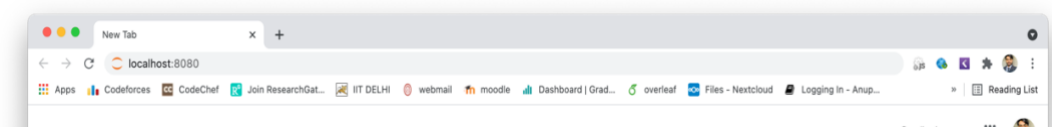
This step will link the Jupyter Notebook port to port 8080 of your local machine thus you can access that remote Jupyter Notebook from local machine using 8080 port.

A terminal window titled 'rajat — rajat1@phantom: ~ — ssh -L 8080:localhost:9090 rajat1@phantom.cse.iitd.ac.in — 80x24'. The output shows the user 'rajat' logging into 'Rajats-MacBook-Pro' via SSH. The prompt is '(base) rajat@Rajats-MacBook-Pro ~ %'. The user enters the command 'ssh -L 8080:localhost:9090 rajat1@phantom.cse.iitd.ac.in'. The terminal shows the password prompt, a welcome message for Ubuntu 16.04.5 LTS, system information, update status (162 packages can be updated), and the last login time. The prompt is now 'rajat1@phantom:~\$'.

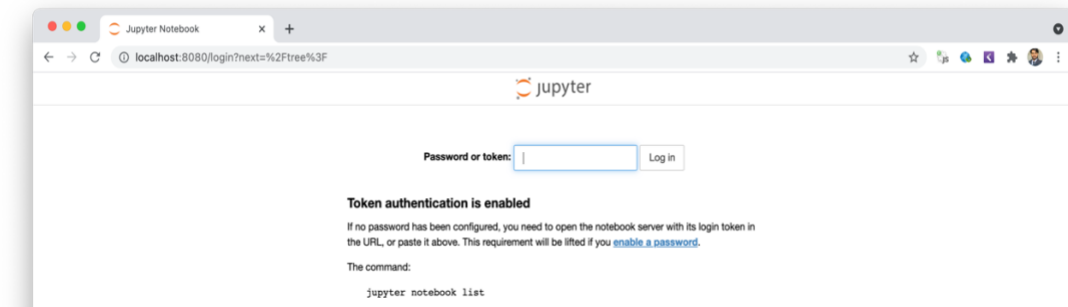
You will get this output after running the above command

5. To access the Remote Notebook you need to fire-up the Notebook from any browser in your machine.

```
localhost:8080
```



- After firing up the notebook you will be forwarded to token authentication page of the jupyter notebook as shown below.



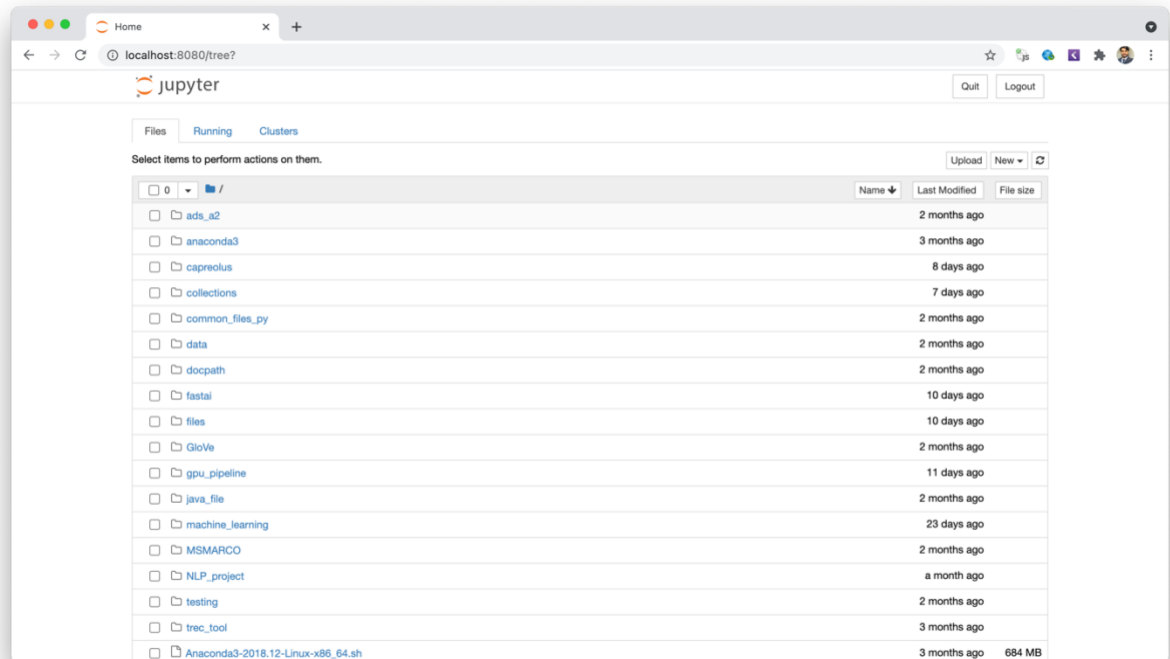
- Paste the token that you will get at step 3 (marked with green box) and log in.

```
rajat1@phantom:~$ ls
Anaconda3-2018.12-Linux-x86_64.sh  NLP_project  capreolus  data  fastai  java_file  trec_tool
GloVe  ads_a2  collections  docpath  files  machine_learning
MSMARCO  anaconda3  common_files_py  examples.desktop  gpu_pipeline  testing

rajat1@phantom:~$ jupyter notebook --no-browser --port=9090
[I 09:48:10.965 NotebookApp] Writing notebook server cookie secret to /run/user/1008/jupyter/notebook_cookie_secret
[I 09:48:11.108 NotebookApp] JupyterLab extension loaded from /home/rajat1/anaconda3/lib/python3.7/site-packages/jupyterlab
[I 09:48:11.108 NotebookApp] JupyterLab application directory is /home/rajat1/anaconda3/share/jupyter/lab
[I 09:48:11.109 NotebookApp] Serving notebooks from local directory: /home/rajat1
[I 09:48:11.109 NotebookApp] The Jupyter Notebook is running at:
[I 09:48:11.109 NotebookApp] http://localhost:9090/?token=5518969de4d262639dc57c849aa9cd72d336c3ddb21283a4
[I 09:48:11.109 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 09:48:11.111 NotebookApp]

To access the notebook, open this file in a browser:
file:///run/user/1008/jupyter/nbserver-19626-open.html
Or copy and paste one of these URLs:
http://localhost:9090/?token=5518969de4d262639dc57c849aa9cd72d336c3ddb21283a4
[I 10:16:48.009 NotebookApp] 302 GET / (127.0.0.1) 0.91ms
[W 10:16:48.020 NotebookApp] Clearing invalid/expired login cookie username=localhost-8080
[W 10:16:48.021 NotebookApp] Clearing invalid/expired login cookie username=localhost-8080
[I 10:16:48.022 NotebookApp] 302 GET /tree? (127.0.0.1) 1.91ms
[I 10:17:24.695 NotebookApp] 302 GET / (127.0.0.1) 0.82ms
[I 10:17:24.707 NotebookApp] 302 GET /tree? (127.0.0.1) 0.97ms
```

8. After login you will be directed to the Jupyter Notebook user interface as shown in the snippet. Enjoy the Jupyter Notebook. Note that the homepage of this notebook will be the location where we have launched the notebook in the remote server (Step 3).



References:

1. <https://ljvmiranda921.github.io/notebook/2018/01/31/running-a-jupyter-notebook/>
2. <https://jupyter.org/install>
3. <https://docs.anaconda.com/anaconda/user-guide/tasks/remote-jupyter-notebook/>
4. <https://poorvi.cse.iitd.ac.in/~anupam/using-conda/>