Electrical and Computer Engineering, Purdue University Northwest
Big Data (ECE49500/ECE59500)
Hands-on Assignment 1

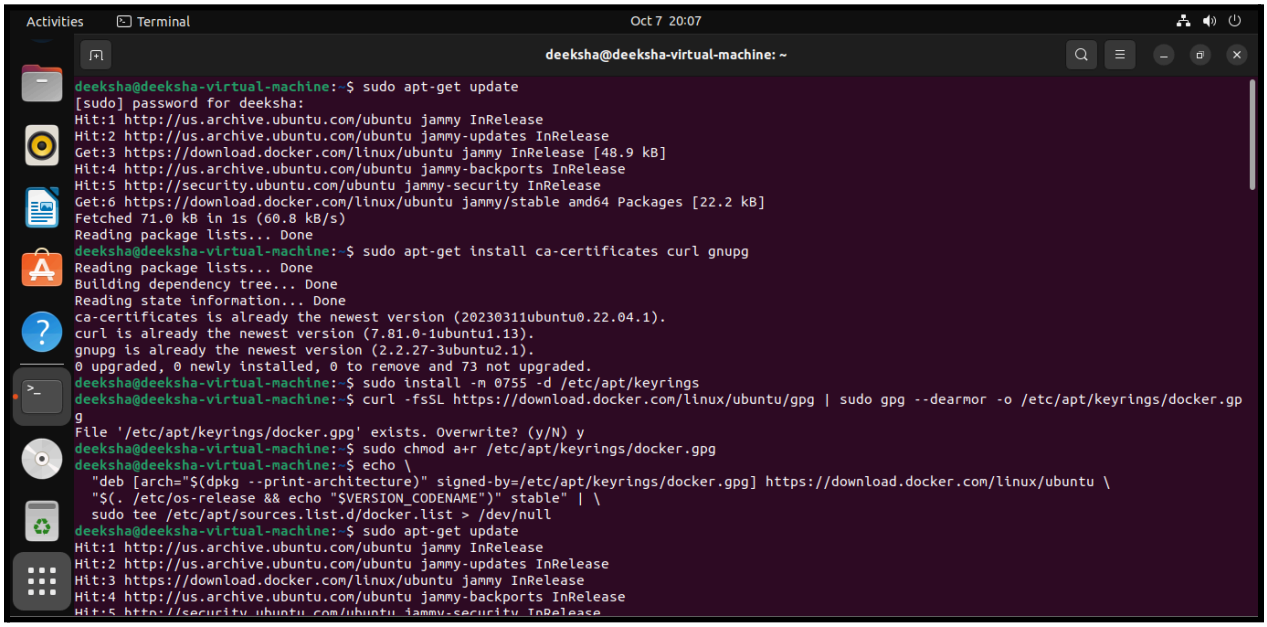
Name - Deeksha Hareesha Kulal

Task 0 [0 points]. Create an Ubuntu virtual machine (VM) and submit the screenshot for it. Please note that I am fine if you want to use Windows for Docker installation, but you have to manage it by yourself. Download VMWare workstation player from- https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html**.**
You may refer [1] for installation.
Download Ubuntu image from- https://releases.ubuntu.com/22.04.3/ubuntu-22.04.3-desktop-amd64.iso?_ga=2.208698834.1557579542.1695500113-1796876152.1695500113**. You may Refer [2] for creating an Ubuntu VM.**

Task 1 [5 points]. Follow the steps mentioned below for Docker installation. Provide screenshots as much as possible for credits. Please watch videos from [4] and [5] to learn Docker and its useful commands. Open the terminal pressing Alt + Ctrl + t.

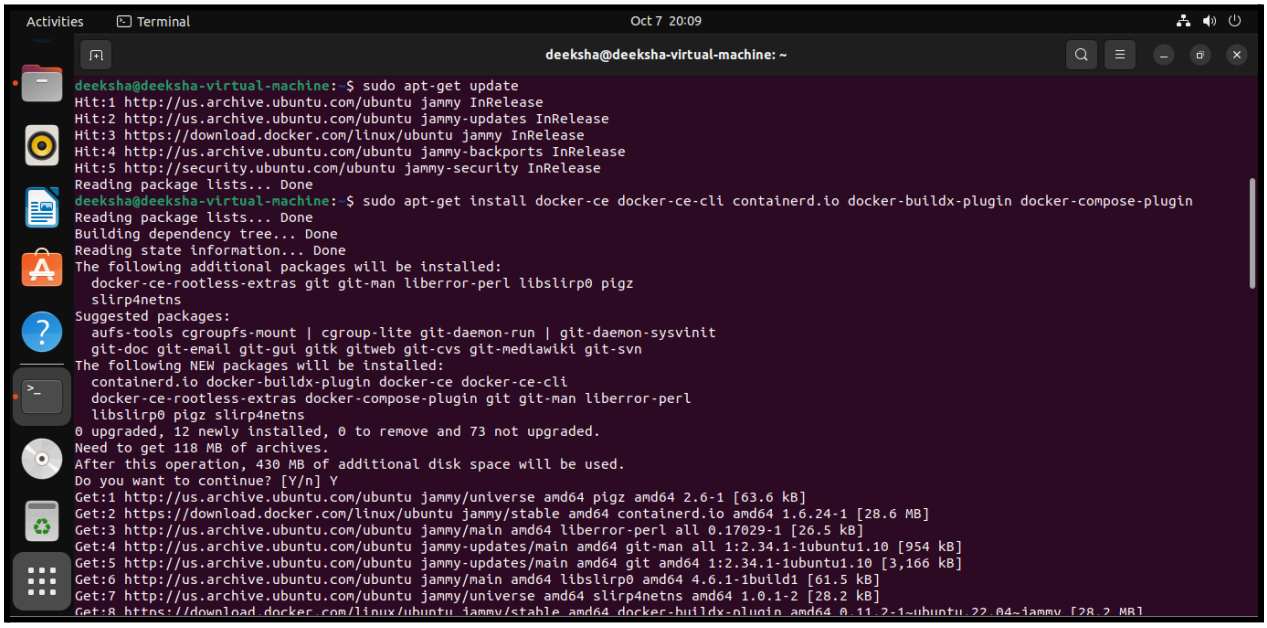
Screenshots for steps for docker installation

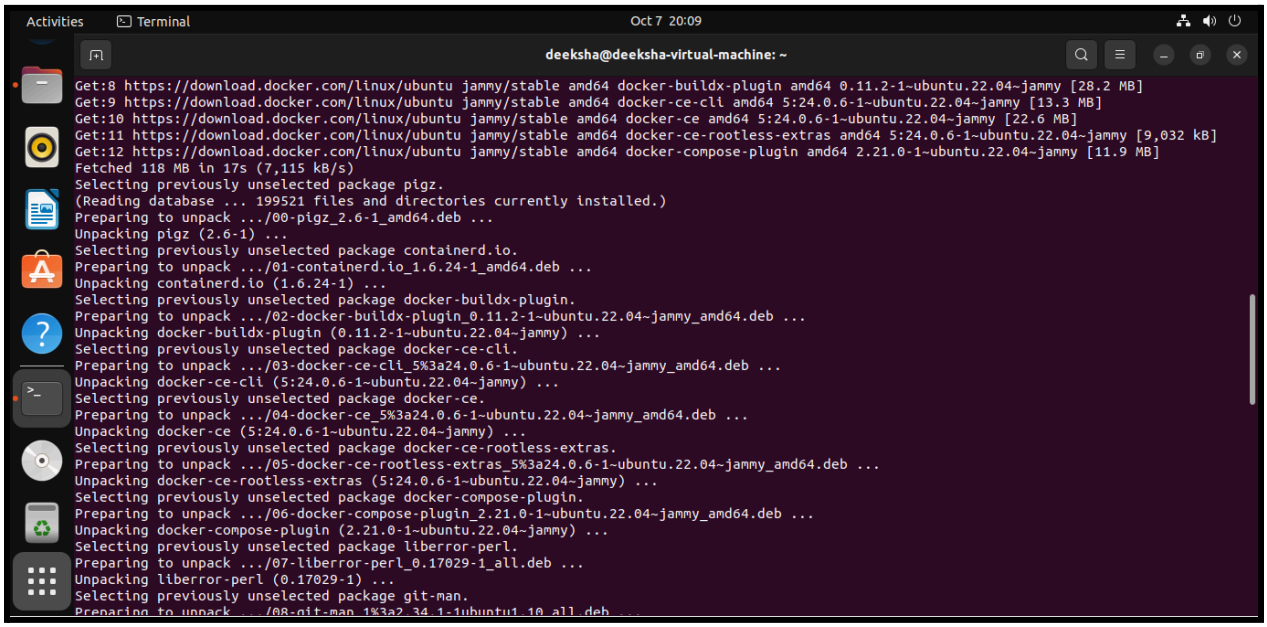
- sudo apt-get update
- sudo apt-get install ca-certificates curl gnupg
- sudo install -m 0755 -d /etc/apt/keyrings
- curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg –dearnor -o /etc/apt/keyrings/docker.gpg
- sudo chmod a+r /etc/apt/keyrings/docker.gpg
- echo \
- sudo apt-get update

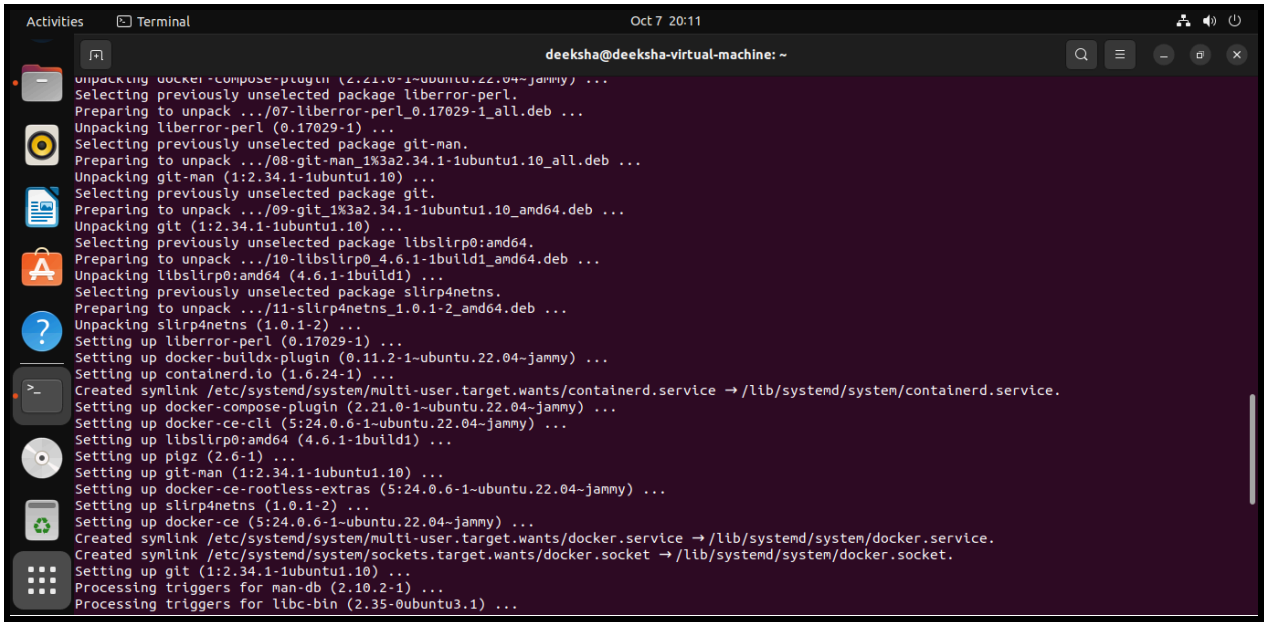


Command -

- sudo apt-get update
- sudo apt-get docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

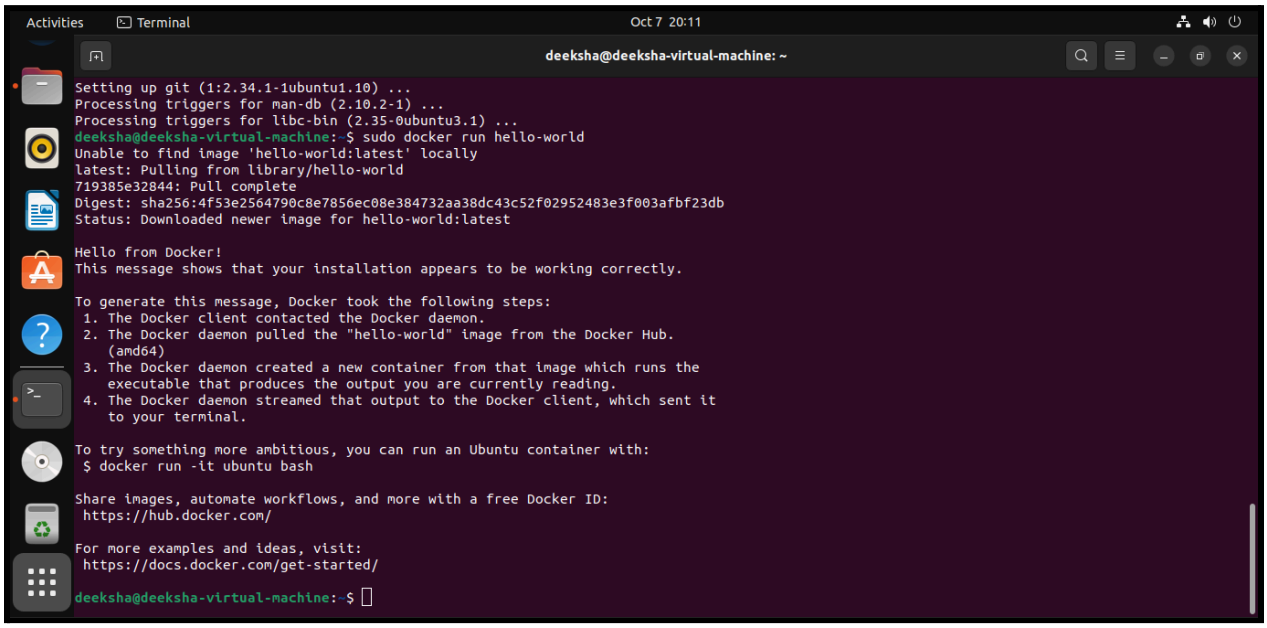






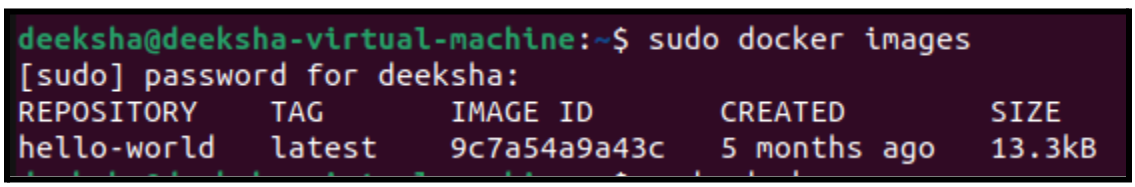
Command -

- sudo docker run hello-world



Follow the steps 1-3 mentioned in section Install Using the Apt Repository of Ref. [3]. Once you finish installation of Docker, execute the following commands:

sudo docker images



sudo docker ps -a

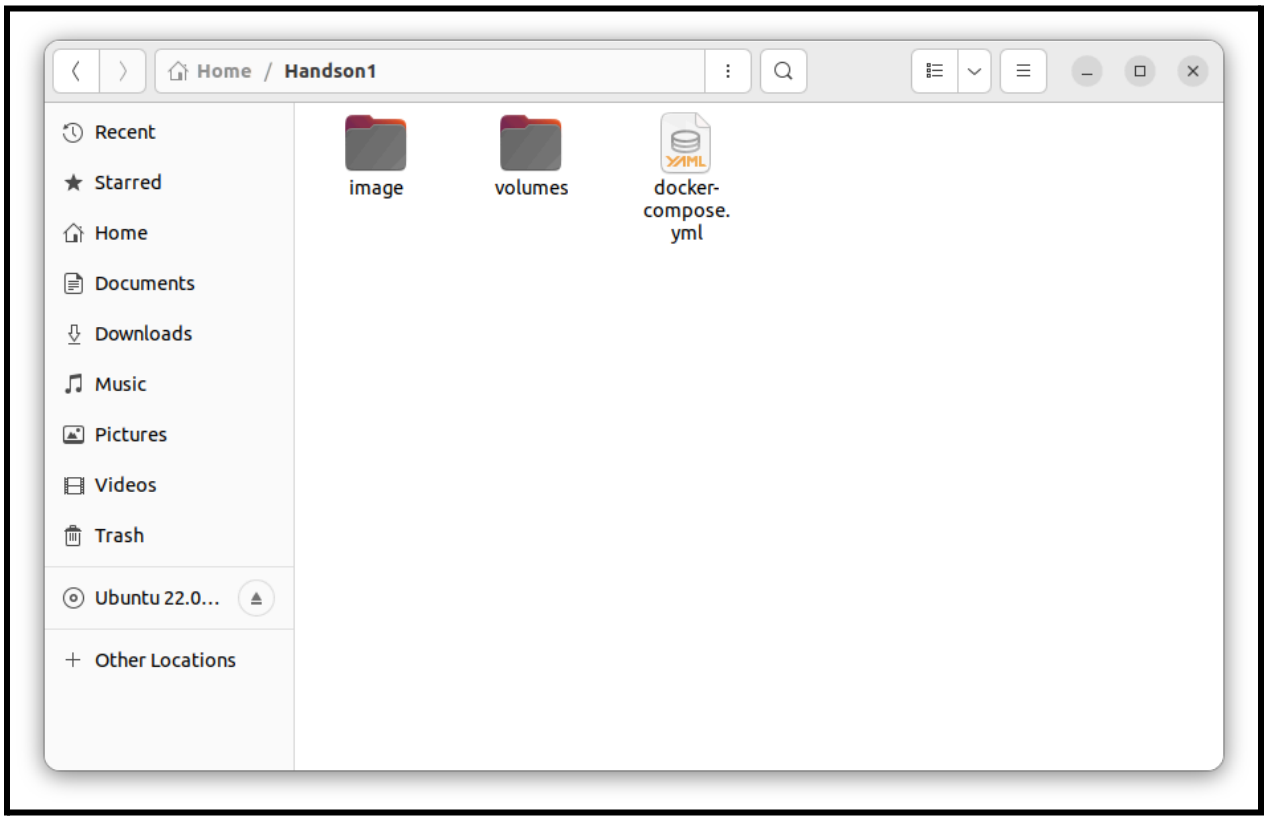
```
deeksha@deeksha-virtual-machine:~$ sudo docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED          STATUS                      PORTS     NAMES
9a918e46a816   hello-world   "/hello"   21 minutes ago   Exited (0) 21 minutes ago             focused_sammet
```

sudo docker network ls

```
deeksha@deeksha-virtual-machine:~$ sudo docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
fa3e335c935f   bridge    bridge    local
87d08010b2f6   host      host      local
16c875099905   none      null      local
```

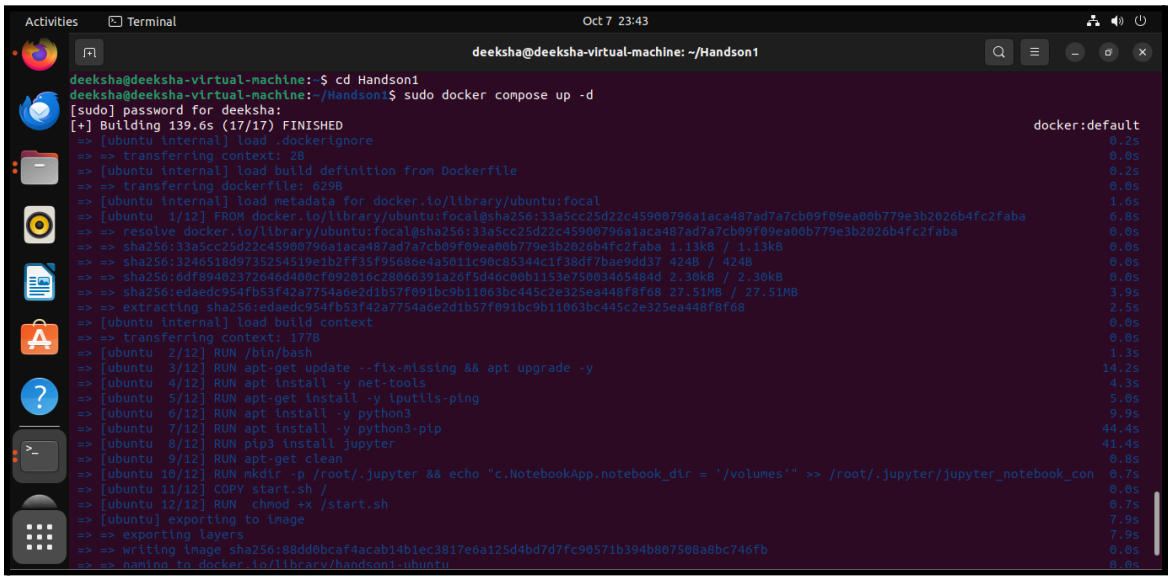
Task 2 [7.5 points]. Follow the steps mentioned below for setting-up the Docker container. Provide screenshots as much as possible for credits.

Download Handson1.zip file in your Ubuntu VM and extract it as Handson1 directory (preferably in your home directory).

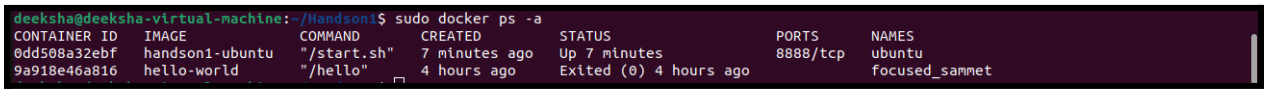


Go to the extracted directory, you will see two directories-- image and volumes; and docker-compose.yml file inside the directory. Open the docker-compose.yml and review

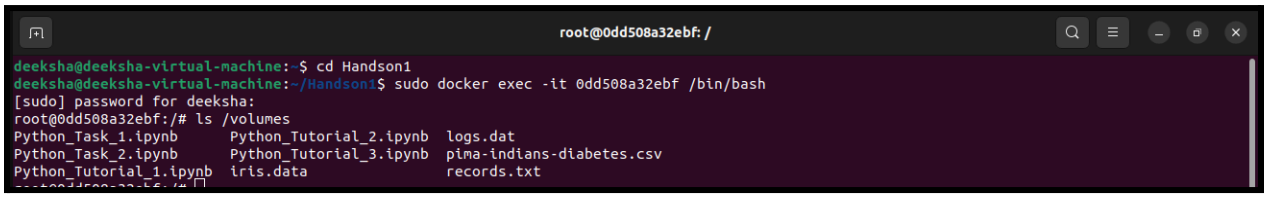
its content. Similarly, review the content of DockerFile inside the image directory.
Now, open the terminal and go to Handson1 directory. Type the command: sudo docker compose up -d to build and start the container.



Type the command: sudo docker ps -a to see the information for the container process, e.g. container ID, command, ports, etc.



Open another terminal in a separate tab using Ctrl+Shift+t. Type command: sudo docker exec -it <first few digits of CONTAINER ID retrieved from docker ps command> /bin/bash to launch the container's shell. Once the shell is launched, type the command: ls /volumes inside container's shell to see the content of /volumes directory.
Do you see .ipynb files?



Type the command: ifconfig in container's shell to know the IP address of the container.

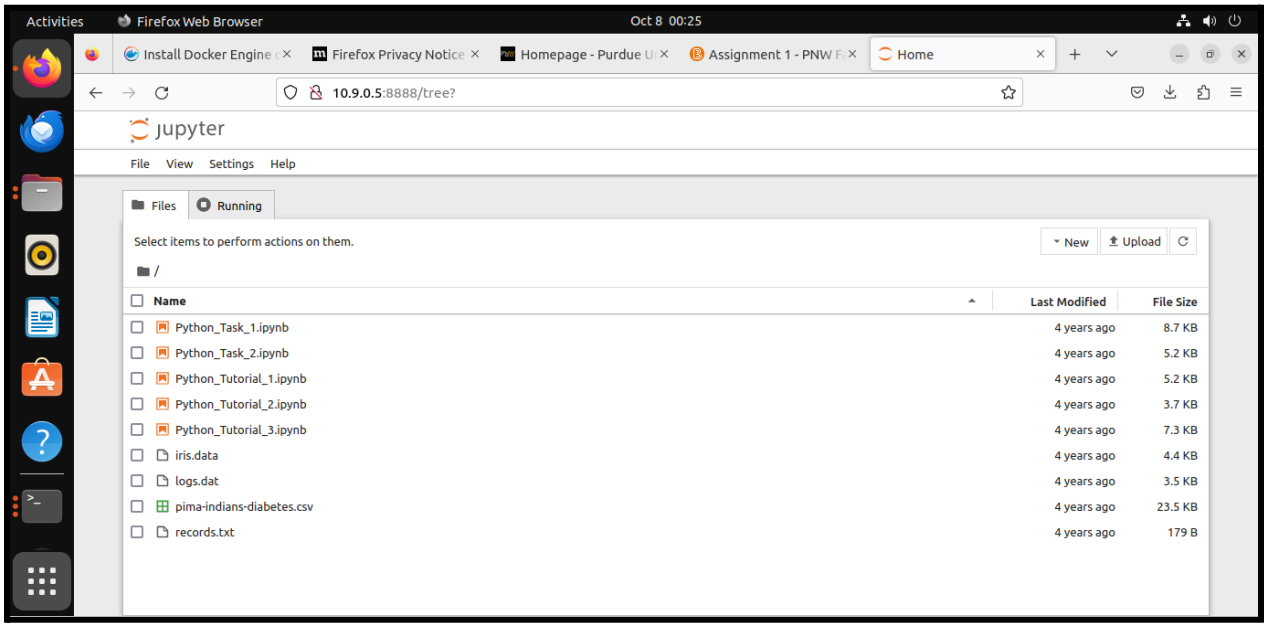
```
root@0dd508a32ebf:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.5  netmask 255.255.255.0  broadcast 10.9.0.255
        ether 02:42:0a:09:00:05  txqueuelen 0  (Ethernet)
        RX packets 76  bytes 13033 (13.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Type another command in container's shell: jupyter notebook list to check the running Jupyter notebook processes inside the container. Please note down the token value that will be needed to access the Jupyter notebooks for tutorials and programming assignments.

```
root@0dd508a32ebf:/# jupyter notebook list
Currently running servers:
http://0dd508a32ebf:8888/?token=32ce626be7573c5b9e04f77ad25453217db8874feb1b5d9c :: /volumes
```

Launch the Firefox browser in your Ubuntu VM and provide this URL: <IP address>:8888. It will launch the Jupyter in your browser. Enter the token value that you noted down earlier if it is being asked. Do you see all the .ipynb files here that you can see in /volumes directory of the container?



Now, you can work on Task 3 and 4. Once you are done with your assignments, you can close the browser, exit the container's shell, and stop the container.

Task 3 [12.5 points]. Complete the programming tasks given in Python_Task_1.ipynb and submit the file for credits.

File submitted.

Task 4 [10 points]. Complete the programming tasks given in Python_Task_2.ipynb and submit the file for credits.

File submitted.