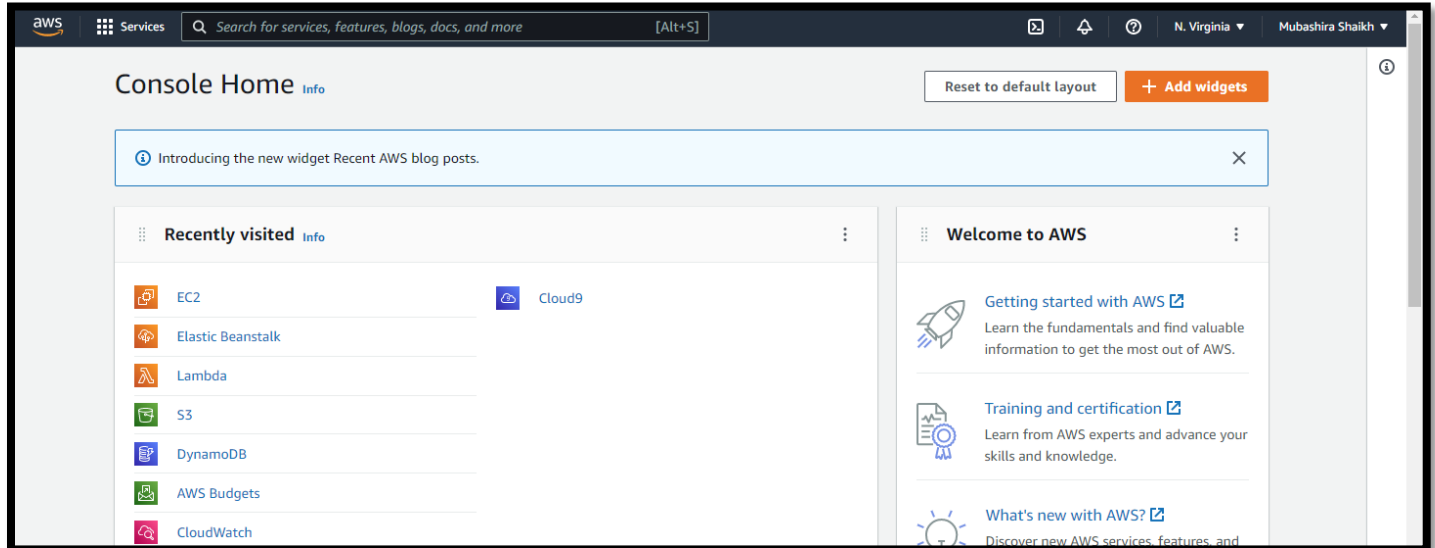


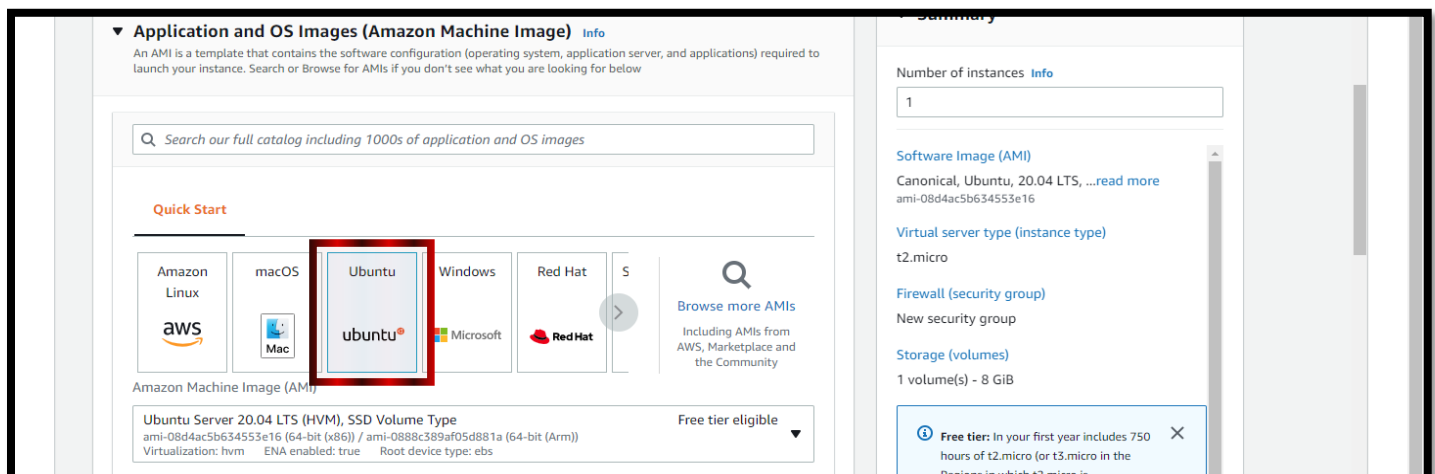
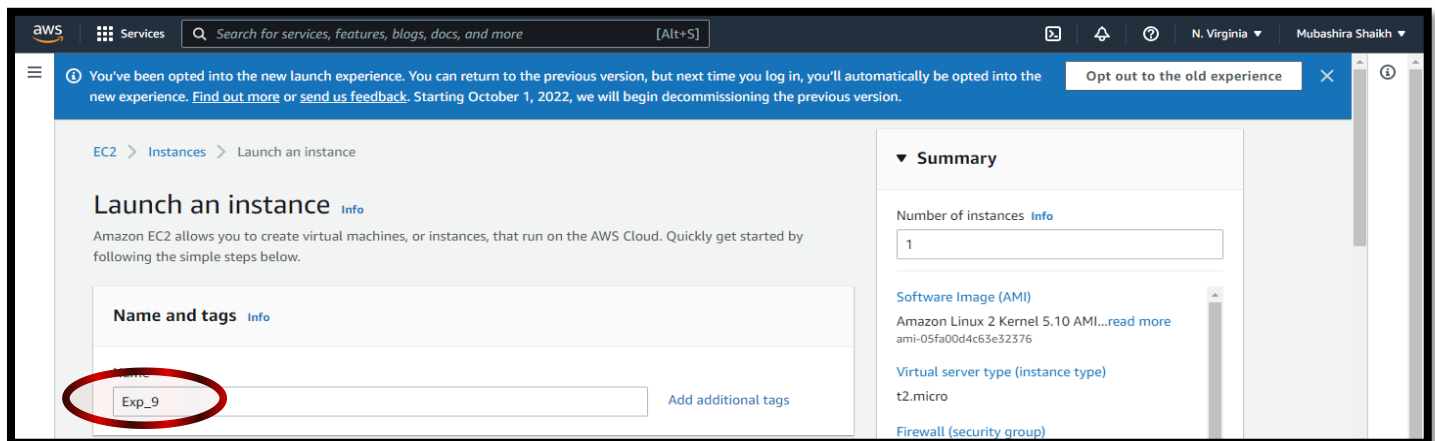
EXPERIMENT-09

NAME: SHAIKH MUBASHIRA TUFEL AHMED
ROLL NO: 612055 **COURSE:** ADVANCE DEVOPS(ITL504)
BRANCH: T.E. INFORMATION TECHNOLOGY (SEM 5)

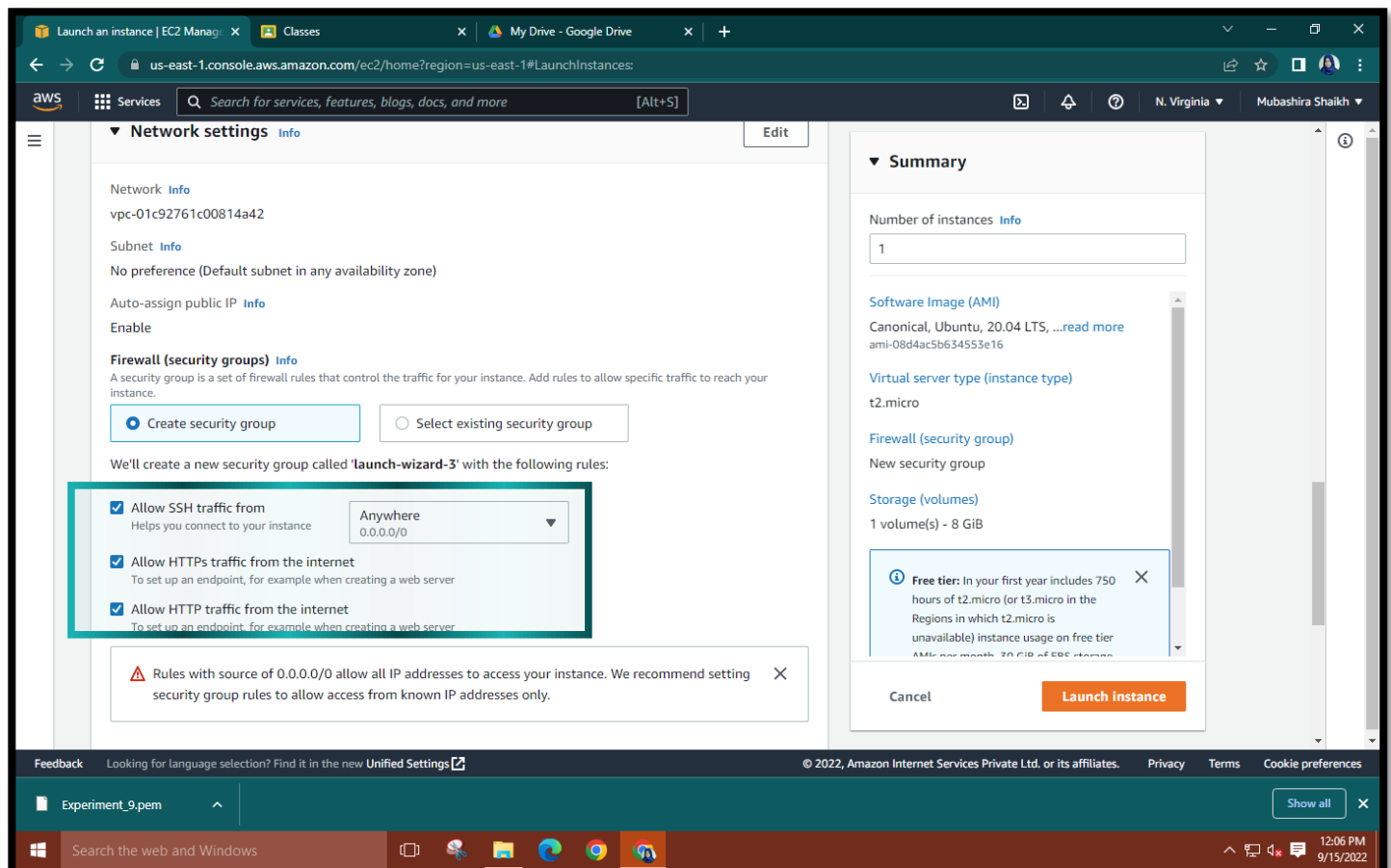
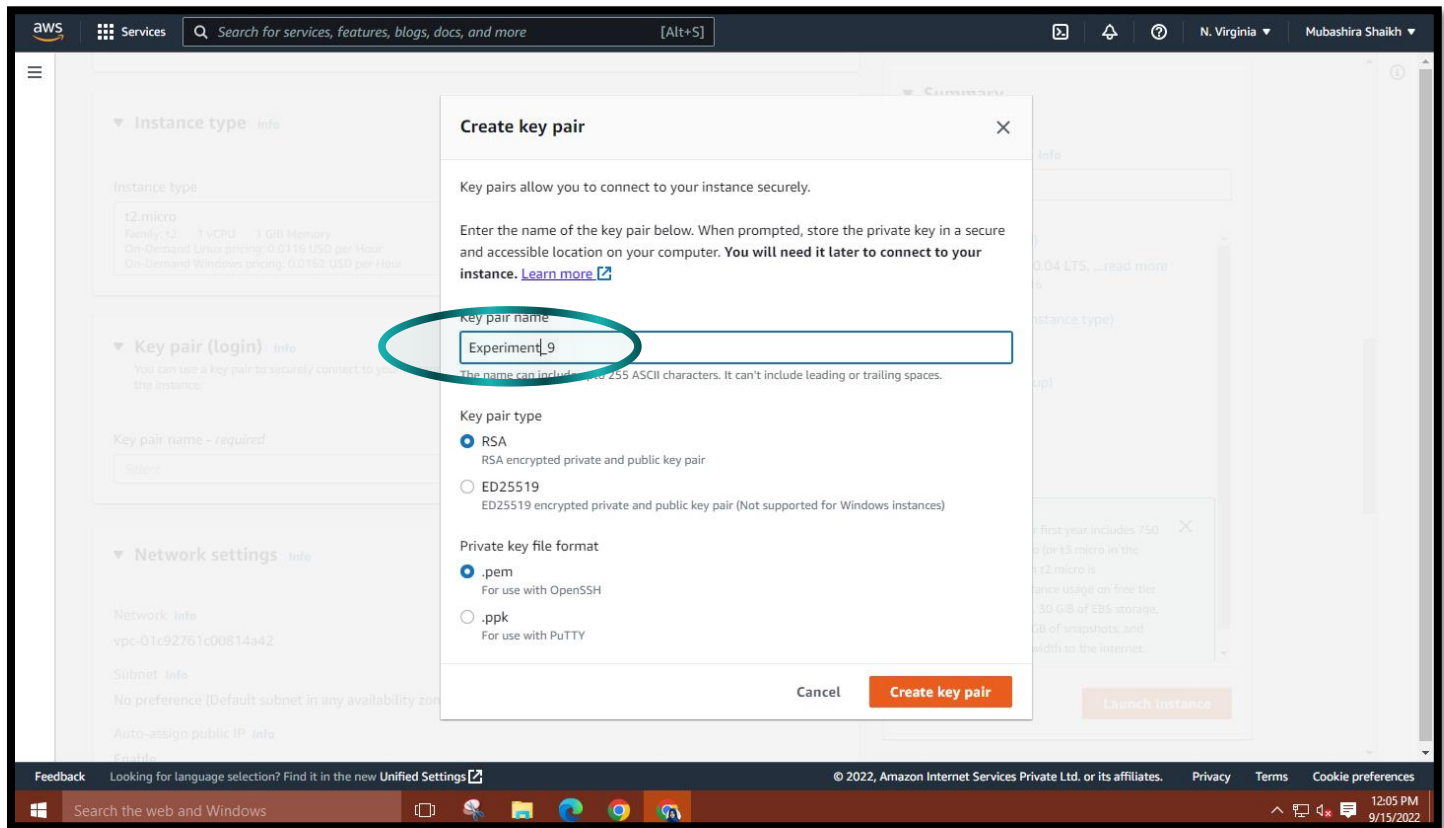
Step 1: AWS Management Console Dashboard.



Step 2: Search for EC2 → Click on Launch instance → Give a name to your instance and create an Ubuntu instance with 20.04 LTS version.



Step 3: Create new key pair for your instance → Network Settings → allow the HTTPS and HTTP traffic.



Step 4: Launch MobaXterm→Select SSH session→Copy the public DNS of your instance and paste it into the remote host. Use the downloaded key pair as the private key.

Step 5: Run the command 'sudo su' to gain root user access. Then enter commands: →curl -fsSL https://get.docker.com -o get-docker.sh and →sh get-docker.sh

```

ubuntu@ip-172-31-31-71:~$ sudo su
root@ip-172-31-31-71:/home/ubuntu# curl -fsSL https://get.docker.com -o get-docker.sh
root@ip-172-31-31-71:/home/ubuntu# ls
Dockerfile  Mubashira  get-docker.sh
root@ip-172-31-31-71:/home/ubuntu# sh get-docker.sh
# Executing docker install script, commit: 4f282167c42b347a931ccfd95cc91fab041d414f
Warning: the "docker" command appears to already exist on this system.

If you already have Docker installed, this script can cause trouble, which is
why we're displaying this warning and provide the opportunity to cancel the
installation.

If you installed the current Docker package using this script and are using it
again to update Docker, you can safely ignore this message.

You may press Ctrl+C now to abort this script.
+ sleep 20
+ sh -c apt-get update -qq >/dev/null
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq apt-transport-https ca-certificates curl >/dev/null
+ sh -c mkdir -p /etc/apt/keyrings && chmod -R 0755 /etc/apt/keyrings
+ sh -c curl -fsSL "https://download.docker.com/linux/ubuntu/gpg" | gpg --dearmor --yes -o /etc/apt/keyrings/docker.gpg
+ sh -c chmod a+r /etc/apt/keyrings/docker.gpg
+ sh -c echo "deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu focal stable" > /etc/apt/sources.li
st.d/docker.list
+ sh -c apt-get update -qq >/dev/null
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq --no-install-recommends docker-ce docker-ce-cli containerd.io docker-compose-plugin
docker-scan-plugin >/dev/null

```

Step 6: Enter commands 'docker --version' to see current docker version & 'docker images' to see installed images.

```
root@ip-172-31-31-71:/home/ubuntu# docker --version
Docker version 20.10.18, build b40c2f6
root@ip-172-31-31-71:/home/ubuntu#
```

```
root@ip-172-31-31-71:/home/ubuntu# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
<none>	<none>	e490414b5f34	18 minutes ago	921MB
python	latest	e285905a3494	41 hours ago	921MB

Step 7: Now create a directory [your own name], go inside it and create another directory name as "APP" and go inside it too.

```
root@ip-172-31-31-71:/home/ubuntu# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
<none> <none> e490414b5f34 18 minutes ago 921MB
python latest e285905a3494 41 hours ago 921MB

root@ip-172-31-31-71:/home/ubuntu# mkdir Mubashira
root@ip-172-31-31-71:/home/ubuntu# cd Mubashira
root@ip-172-31-31-71:/home/ubuntu/Mubashira# mkdir app
root@ip-172-31-31-71:/home/ubuntu/Mubashira# cd app
root@ip-172-31-31-71:/home/ubuntu/Mubashira/app# nano app.py
```

Step 8: Now open nano editor by using command [nano app.py] and add the code of flask given in the classroom.

The screenshot shows the MobaXterm interface with a terminal window and a file explorer. The terminal window displays the following code in the nano editor:

```
GNU nano 4.8
from flask import Flask

app = Flask(__name__)

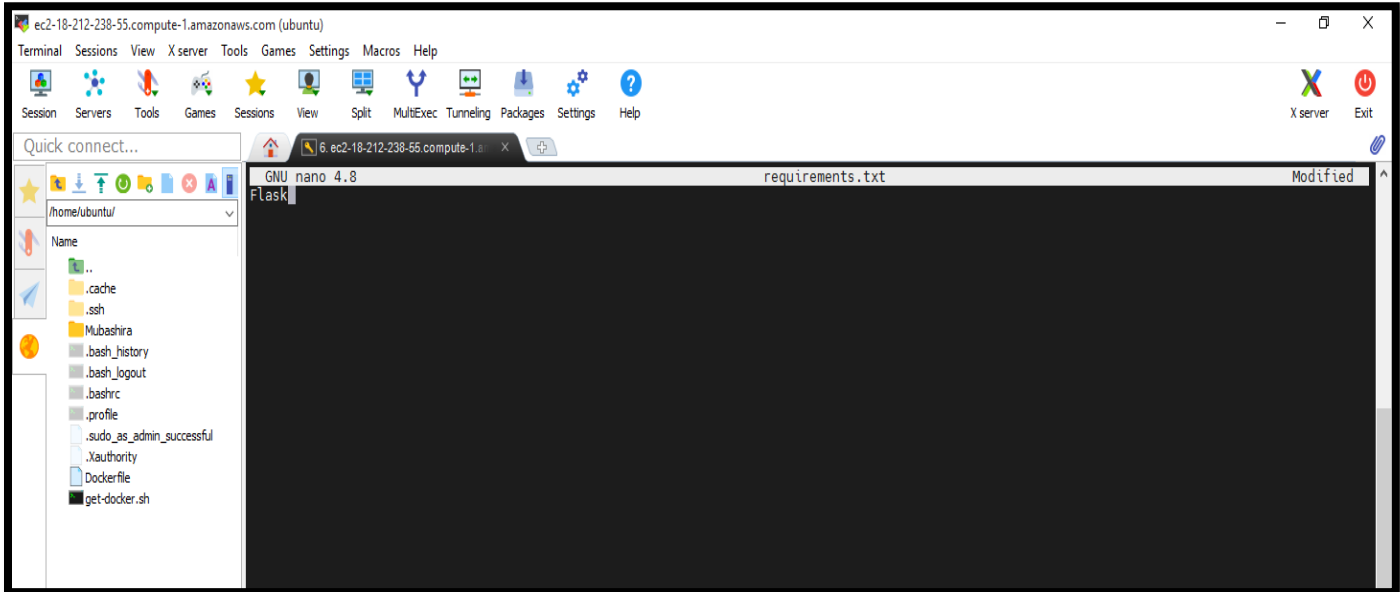
@app.route('/')
def hello():
    return "Hello from Mubashira, Demo Flask & Docker application is up and running!"

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=80)
```

The file explorer on the left shows the directory structure: /home/ubuntu/ > Mubashira > app. The terminal window also shows the command prompt for the nano editor.

***To save the file press CTRL+O → ENTER → CTRL+ X.**

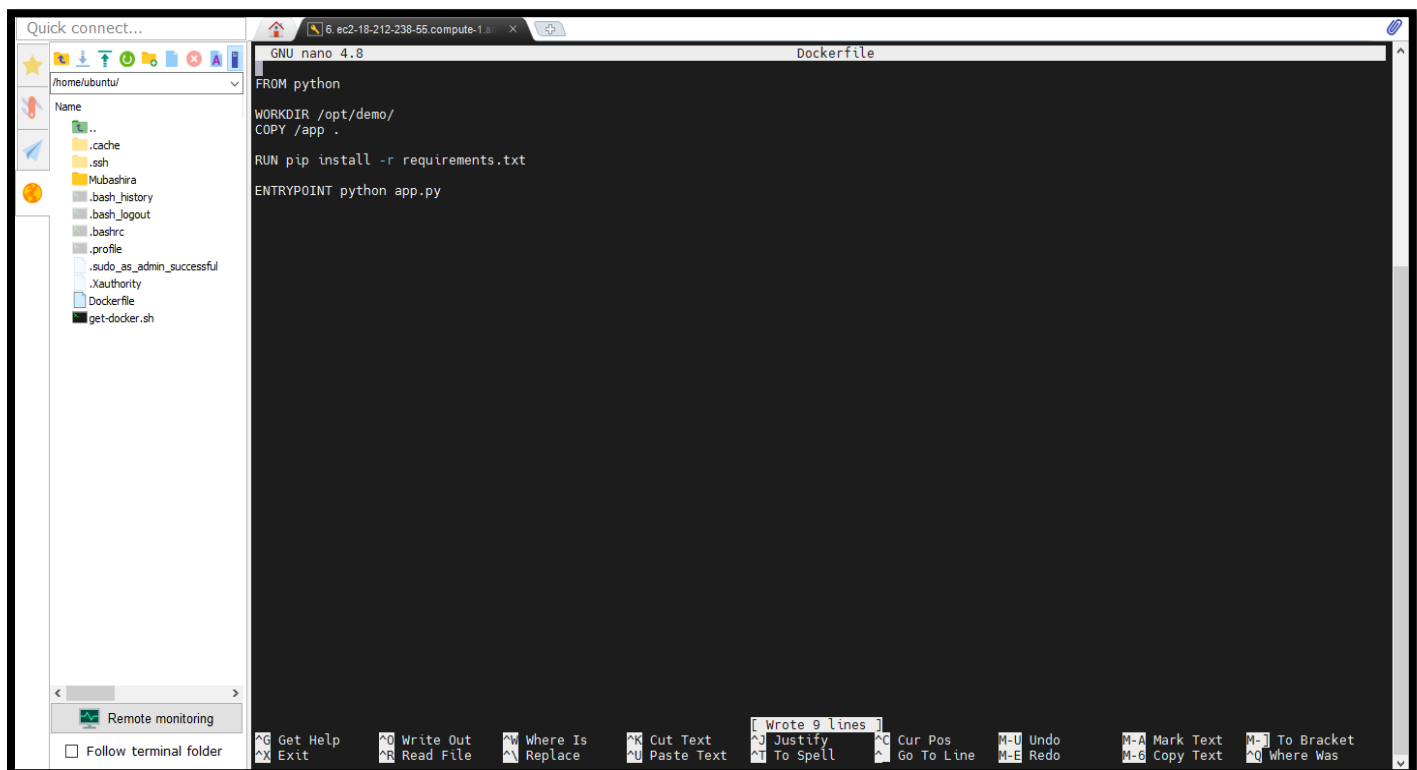
Step 9: Now open nano editor by using command [requirements.txt] and write the requirements of the file to be installed on other PC.



Step 10: Now go back to your Parent Directory and write command [nano Dockerfile] and copy the code give in the classroom.

```
root@ip-172-31-31-71:/home/ubuntu/Mubashira/app# nano app.py
root@ip-172-31-31-71:/home/ubuntu/Mubashira/app#
root@ip-172-31-31-71:/home/ubuntu/Mubashira/app# nano requirements.txt
root@ip-172-31-31-71:/home/ubuntu/Mubashira/app#
root@ip-172-31-31-71:/home/ubuntu/Mubashira/app# cd ..
root@ip-172-31-31-71:/home/ubuntu/Mubashira# nano Dockerfile
```

Write code in the Dockerfile.



Step 11: Now enter the command “docker build -t [file name]:latest.”

```
Build an image from a Dockerfile
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker build -t mubashira:latest .
Sending build context to Docker daemon  4.68kB
Step 1/5 : FROM python
--> e285995a3494
Step 2/5 : WORKDIR /opt/demo/
--> Using cache
--> e490414b5f34
Step 3/5 : COPY /app .
--> 3c46f9f2ea88
Step 4/5 : RUN pip install -r requirements.txt
--> Running in 1a13e91dbb01
Collecting Flask
  Downloading Flask-2.2.2-py3-none-any.whl (101 kB)
    101.5/101.5 kB 3.0 MB/s eta 0:00:00
Collecting Werkzeug>=2.2.2
  Downloading Werkzeug-2.2.2-py3-none-any.whl (232 kB)
    232.7/232.7 kB 31.5 MB/s eta 0:00:00
Collecting Jinja2>=3.0
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
    133.1/133.1 kB 2.9 MB/s eta 0:00:00
Collecting click>=8.0
  Downloading click-8.1.3-py3-none-any.whl (96 kB)
    96.6/96.6 kB 15.4 MB/s eta 0:00:00
Collecting itsdangerous>=2.0
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting MarkupSafe>=2.0
  Downloading MarkupSafe-2.1.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Installing collected packages: MarkupSafe, itsdangerous, click, Werkzeug, Jinja2, Flask
Successfully installed Flask-2.2.2 Jinja2-3.1.2 MarkupSafe-2.1.1 Werkzeug-2.2.2 click-8.1.3 itsdangerous-2.1.2
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommen
ded to use a virtual environment instead: https://pip.pypa.io/warnings/venv
Removing intermediate container 1a13e91dbb01
--> 87729ba59de0
Step 5/5 : ENTRYPOINT python app.py
--> Running in c16dd93860d3
Removing intermediate container c16dd93860d3
--> d8624cee440c
Successfully built d8624cee440c
Successfully tagged mubashira:latest
root@ip-172-31-31-71:/home/ubuntu/Mubashira#
```

Step 12: Now for checking the images run command as “docker images”.

```
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mubashira     latest   d8624cee440c   About a minute ago   933MB
python        latest   e285995a3494   41 hours ago       921MB
root@ip-172-31-31-71:/home/ubuntu/Mubashira#
```

Step 13: Run command “docker run -d -p 80:80 [directory name]”.

```
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mubashira     latest   d8624cee440c   About a minute ago   933MB
python        latest   e285995a3494   41 hours ago       921MB
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker run -d -p 80:80 mubashira
fd68eaf9421553453676430eb6734d2d0b7eda506ddc35105387fa2222dd50cb
root@ip-172-31-31-71:/home/ubuntu/Mubashira#
```

copy the IPV4 address from the EC2 instance details and paste it into a web browser.

Instances | EC2 | Experiment 9 | Dockerfile - Go | Experiment-9 - | My Drive - Go | empty folder in | let quotes - Go | ec2-18-212-238-55.compute-1.amazonaws.com

ec2-18-212-238-55.compute-1.amazonaws.com

Hello from Mubashira, Demo Flask & Docker application is up and running!

Step 14: Run the commands '`docker ps`' to check the number of containers and their ID's.
You can use the command '`docker ps -a`' to check the status of the container.
To stop a container use command: '`docker stop (container id)`'.

```
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mubashira      latest    d8624cee440c   About a minute ago  933MB
python         latest    e285995a3494   41 hours ago    921MB
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker run -d -p 80:80 mubashira
fd68eaf9421553453676430eb6734d2d0b7eda506ddc35105387fa2222dd96cb
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS                               NAMES
fd68eaf94215   mubashira "/bin/sh -c 'python ..." 48 seconds ago Up 47 seconds 0.0.0.0:80->80/tcp, :::80->80/tcp suspicious_kare
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS                               NAMES
fd68eaf94215   mubashira "/bin/sh -c 'python ..." 54 seconds ago Up 53 seconds 0.0.0.0:80->80/tcp, :::80->80/tcp suspicious_kare
root@ip-172-31-31-71:/home/ubuntu/Mubashira# docker stop fd68
fd68
```

Quit MobaXterm and then delete your EC2 instance.