Experiment 4
Docker Installation & Basic Commands of Docker

Part A:

Steps for Installing Docker:

1. Open the terminal on Ubuntu.

2. Remove any Docker files that are running in the system, using the following command:

$ sudo apt-get remove docker docker-engine docker.io

After entering the above command, you will need to enter the password of the root and press enter.

3. Check if the system is up-to-date using the following command:
$ sudo apt-get update

4. Install Docker using the following command:
$ sudo apt install docker.io
You'll then get a prompt asking you to choose between y/n - choose y

5. Install all the dependency packages using the following command:
$ sudo snap install docker

```
      alternate commands to install docker are

      $ sudo apt-get install \
       apt-transport-https \
       ca-certificates \
       curl \
 software-properties-common

To nstall packages to allow apt to use a repository over HTTPS
```

$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add
commad to add Docker's official GPG key

```
$ sudo apt-key fingerprint 0EBFCD88
Verify that you now have the key with the fingerprint
```

6. Before testing Docker, check the version installed using the following command:
$ docker –version

7. Pull an image from the Docker hub using the following command:
$ sudo docker run hello-world

Here, hello-world is the docker image present on the Docker hub. Output will be like this as in figure.

8. The actual Hellow World command of docker is
$ docker run docker/whalesay cowsay boo
The default image of docker appears with the message boo.

Check if the docker image has been pulled and is present in your system using the following command:
$ sudo docker images

To display all the containers pulled, use the following command:
$ sudo docker ps -a

Part B: Docker search, docker Pull and docker run

Use the command docker search to search for public images on the Docker hub. It will return information about the image name, description, stars, official and automated.

Now that we know the name of the image, we can pull that from the Docker hub using the command docker pull. Here, we are setting the platform option as well.
$ sudo docker search mysql

or alternate

$ sudo docker pull -platform linux/x86_64 mysql

or alternate

```
$sudo docker pull mysql/mysql-erver:tag
```

Log into MySQL within the docker container using the `docker exec` command:
$ sudo docker exec -it mysql bash

Now run this command

```
$ sudo docker run --name mysql -p 3406:3306 -e MYSQL_ROOT_PASSWORD=anypassword -
d mysql/mysql-server:5.7
```

You can check it by running the following command…The first image as you can see in the snippet is the mysql-server image in anew terminal

```
$ sudo docker ps -a
```

Remember, when we created and ran the MySQL container, we provided `MYSQL_ROOT_PASSWORD=anypassword`. Create a database and user, and grant privileges in MySQL (from within the container). og into MySQL within the docker container using the `docker exec` command, Log into MySQL if you haven't already. After login, the `mysql>` prompt shows up:

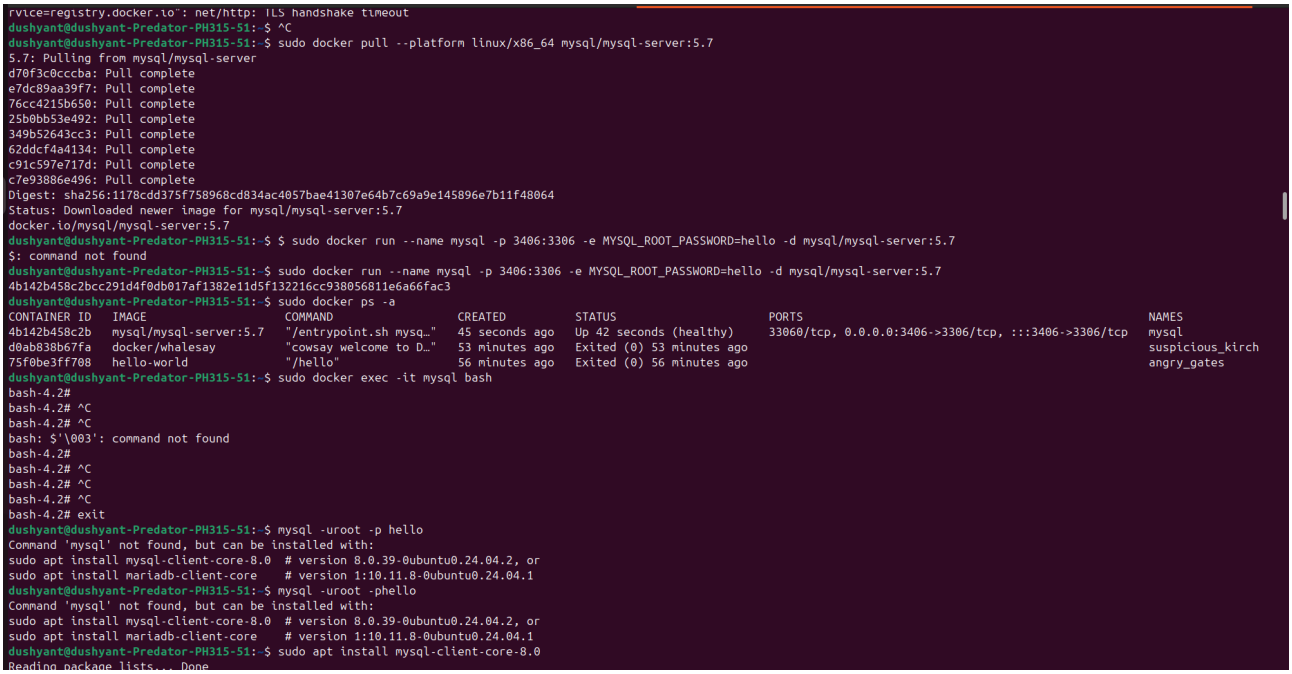
```
$ mysql -uroot -panypassword

$ SHOW DATABASES ;

$ use database;

$ exit

$ exit
```

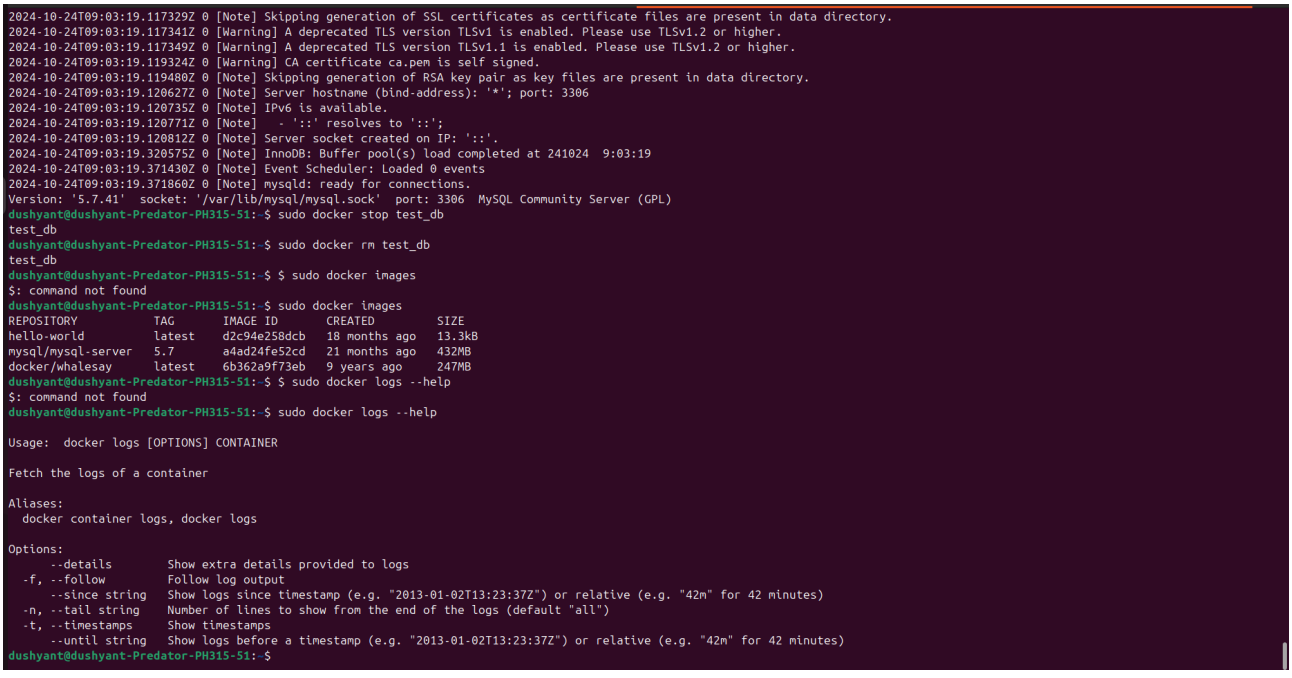
```
$docker restart
Let's restart our stopped contained by using the following command. We may want
to use this after we reboot our machine.
docker restart f8c52bedeecc

$docker rename
Now, let's change the container name from compassionate_fermi to test_db. We may
want to change the name to keep track of our containers more easily.
docker rename compassionate_fermi test_db

$docker exec
Access the running container test_db by running the following command. It's
helpful, if we want to access the MySQL command line and execute MySQL queries.

docker exec -it test_db bash
mysql -uroot -pmy-secret-pw
SHOW DATABASES;

The -i and -t options are used to access the container in an interactive mode.
Then we provide the name of the container we want to access, which in this case
test_db. Finally, the bash command is used to get a bash shell inside the
container.

$docker logs
This command is helpful to debug our Docker containers. It will fetch logs from
a specified container.

$docker logs test_db

If we want to continue to stream new output,use the option -follow.
docker logs -follow test_db
$docker rm

To remove a container, we can use the following command.
docker rm test_db
You may encounter an error like
Error response from daemon: You cannot remove a running container ……
Stop the container before attempting removal or force remove
As it recommends, we can stop the container first and then remove it or use
option -f to remove a running container forcefully.
```

$docker stop test_db
$docker rm test_db# ordocker rm -f test_db
$docker rmi

To free some disk space, we can use the docker rmi command with the image id to remove an image.
$docker rmi eb0e825dc3cf

These commands come with plenty of helpful options. If you want to know about other available options, run the docker command_name --help command. For example:
docker logs –help





Reference:
1. https://www.simplilearn.com/tutorials/docker-tutorial/how-to-install-docker-on-

ubuntu
2. https://towardsdatascience.com/12-essential-docker-commands-you-should-know-c2d5a7751bb5
3. https://docs.docker.com/engine/reference/commandline/container/
4. https://towardsdatascience.com/15-docker-commands-you-should-know-970ea5203421

```
190  sudo apt-get remove docker docker-engine docker.io
191  sudo apt-get update
192  sudo apt install docker.io
193  sudo snap install docker
194  docker --version
195  sudo docker run hello-world
196  docker run docker/whalesay cowsay boo
197  sudo docker run docker/whalesay cowsay welcome to Docker!
198  sudo docker images
199  sudo docker ps -1
200  sudo docker ps -a
201  sudo docker ps
202  sudo docker search mysql
203  sudo docker exec -it mysql bash
204  mysql -v
205  sudo apt install mysql
206  sudo apt install sql
207  mysql --root
208  sudo docker pull mysql/mysql-server:5.75.7:pullingfrom mysql/mysql-server
209  \cd
210  sudo docker -v
211  sudo docker exec -it mysql bash
212  $ sudo docker search mysql
213  sudo docker search mysql
214  sudo docker pull --platform linux/x86_64 mysql/mysql-server:5.7
215  $ sudo docker run --name mysql -p 3406:3306 -e MYSQL_ROOT_PASSWORD=hello -d mysql/mysql-server:5.7
216  sudo docker run --name mysql -p 3406:3306 -e MYSQL_ROOT_PASSWORD=hello -d mysql/mysql-server:5.7
217  sudo docker ps -a
218  sudo docker exec -it mysql bash
219  mysql -uroot -p hello
220  mysql -uroot -phello
221  sudo apt install mysql-client-core-8.0
222  mysql -uroot -phello
223  mysql -uroot -p hello
224  sudo apt install mysql-server
225  mysql -uroot -p hello
226  sudo systemctl status mysql
227  sudo systemctl start mysql
228  mysql -uroot -p
229  sudo mysql
230  mysql -uroot -p
231  $ sudo docker restart mysql
232  sudo docker restart mysql
233  sudo docker rename compassionate_fermi test_db
234  sudo docker ps -a
235  sudo docker rename mysql test_db
236  sudo docker exec -it test_db bash
237  mysql -uroot -p
238  sudo docker logs test_db
239  sudo docker stop test_db
240  sudo docker rm test_db
241  $ sudo docker images
242  sudo docker images
243  $ sudo docker logs --help
244  sudo docker logs --help
245  sudo docker image
246  sudo docker images
247  sudo docker kill mysql
248  sudo docker ps
249  sudo docker run docker/whalesay cowsay welcome to Docker!
250  sudo docker ps
```