**Docker commands**

To pull simple hello-world docker image use command:

docker run hello world.

To see all the images curruntly you have in your machine.

docker images

To push and pull own images

Pushing:

1)Login:

$ docker login

2)Tag the image

$ docker tag <image\_name>:version username/image\_name

$ docker tag sample-project:latest romapatel7826/sample-project

3)After docker images you'll see the image listed

$ docker images

op: romapatel7826/sample-project

**4)Push to docker hub**

$ docker push roma7826/sample-project

Now after few seconds the image will be listed in the repository section of dockerhub.

**Pulling:**

To pull the image see public view and grab the command and use to pull the image.

To remove an image from your machine

$ docker rmi image\_name

To run the image

After pull we can run the image using command:

$ docker run -p port docker-image-name

Example: $ docker run -p 7000:7000 romapatel7826/sample-project

**Dockerizing the django application**

1)Create a django application.

2)Create requirements.txt file using:

pip freeze > requirements.txt

3)Create a Dockerfile with no extension inside the project file where manage.py is.

4)Edit the Dockerfile:

# Its a lightweight linux OS with python already installed

FROM python:3.8-slim-buster

# The working directory

WORKDIR /app

# Copy requirements.txt

COPY requirements.txt requirements.txt

# Running the requirements.txt file

RUN pip3 install -r requirements.txt

# Copy all the folders

COPY . .

# Run the commands

CMD ["python3", "manage.py", "runserver", "0.0.0.0:7000"]

5)Build the image using the command:

$ docker build --tag name-of-image .

#tag is used to give name to the image

#dot refers to the current folder

Example:

$ docker build --tag sample-project .

6)Publish using the 7000 port

$ docker run --publish 7000:7000 python-django

After this command we can see our application running on port 7000 on browser.

Docker will run each commands given in Dockerfile when we run it.

To list all the running containers

This will list all the running docker images.

$ docker ps

$ docker ps --all #this will show all the exited container

To stop the container

1)Do the docker ps, grab the id of the particular container which we want to stop.

2)Run docker stop container\_id(refer your container\_id)

$ docker stop 66d892081f75 (this is my container id )

To kill the running container

$ docker kill container\_id

To list all the files and folders inside the app.

$ docker run filename ls

EXample:

$ docker run python-django ls

To run the commands as in ls, pwd etc

This will run the shell and we can execute the commands

$ sudo docker exec -i -t 78cdb05b5163 sh

Run the commands

ls

pwd

To check the logs

We can see the logs of the container which is running.

$ docker logs container\_id

To completely remove docker from your system along with images and container, follow the below link

https://askubuntu.com/questions/935569/how-to-completely-uninstall-docker