1.docker pull mongo

==> pulls mongo image from dockerhub

2.docker pull mongo-express

==> pulls mongo-express image from dockerhub

3.docker images

==> lists all images

4.docker network ls

==> shows docker networks

5.docker network create mongo-network

==> creates a network named : mongo-network

6.docker run -d \
-p 27017:27017 \
-e MONGO_INITDB_ROOT_USERNAME=mongoadmin \
-e MONGO_INITDB_ROOT_PASSWORD=password \
--name mongodb \
--net mongo-network2 \
mongo

==>runs a mongodb image

7.docker logs "containerid"
==> show logs

8.docker run -d \
-p 8081:8081 \
-e ME_CONFIG_MONGODB_ADMINUSERNAME=adminmongo \
-e ME_CONFIG_MONGODB_ADMINPASSWORD=password \
--net mongo-network2 \
--name mongo-express \
-e ME_CONFIG_MONGODB_SERVER=mongodb
mongo-express

==>runs a mongodb container

9.docker ps

==> lists running instances

---

10. docker-compose -f mongo.yaml up

==> start the docker compose file

11.docker-compose -f mongo.yaml down

==>stops the docker compose file

12.docker build -t my-app:1.1 .

==>(build image, first parameter image name,second parameter path to docker file)

13.docker run ***:1.0

14.docker rmi ****

==> removes images by their ID

15.docker exec -it **** /bin/sh

==> goes inside the container

16.docker tag my-app:1.0 XXXXXXamazonaws.com/my-app:1.0

==> makes a copy of my-app and names it: XXXXamazonaws
(identical images)

17.docker push XXXXXXamazonaws.com/my-app:1.0

==> takes image and pushes it into repository at the address
