Working with Docker Commands and Docker Hub

**1. Pull latest nginx image from Docker hub**

# docker pull nginx

Using default tag: latest

latest: Pulling from library/nginx

45b42c59be33: Pull complete

8acc495f1d91: Pull complete

ec3bd7de90d7: Pull complete

19e2441aeeab: Pull complete

f5a38c5f8d4e: Pull complete

83500d851118: Pull complete

Digest: sha256:f3693fe50d5b1df1ecd315d54813a77afd56b0245a404055a946574deb6b34fc

Status: Downloaded newer image for nginx:latest

docker.io/library/nginx:latest

2. **Check if image have been downloaded**

# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

nginx latest 35c43ace9216 2 weeks ago 133MB

**3. Create nginx container from the image**

# docker run -d nginx:latest

7c6e7c6730b94de9a7c92809989d73c7a55303614da25ba34b86419e259275db

**4. Check if nginx container has been created**

# docker ps or docker container ls

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

7c6e7c6730b9 nginx:latest "/docker-entrypoint.…" About a minute ago Up About a minute 80/tcp objective\_sinoussi

**5. Check the ipaddress details of the container by using docker inspect command**

#docker inspect 7c6e7c6730b9 | grep -i ip

"IpcMode": "private",

"LinkLocalIPv6Address": "",

"LinkLocalIPv6PrefixLen": 0,

"SecondaryIPAddresses": null,

"SecondaryIPv6Addresses": null,

"GlobalIPv6Address": "",

"GlobalIPv6PrefixLen": 0,

"IPAddress": "172.17.0.2",

"IPPrefixLen": 16,

"IPv6Gateway": "",

"IPAMConfig": null,

"IPAddress": "172.17.0.2",

"IPPrefixLen": 16,

"IPv6Gateway": "",

"GlobalIPv6Address": "",

"GlobalIPv6PrefixLen": 0,

**6. Logon to the nginx container and create some sample files**

# docker exec -it 7c6e7c6730b9 bash

touch example.txt

# ls

bin dev docker-entrypoint.sh example.txt lib media opt root sbin sys usr

boot docker-entrypoint.d etc home lib64 mnt proc run srv tmp var

To exit :Ctrl+P+Q

**7. Stop the container and again login and check if the example.txt file still exists**

docker stop 7c6e7c6730b9

docker start 7c6e7c6730b9

# docker exec -it 7c6e7c6730b9 bash

ls

example.txt

**Pushing container changes to Docker Hub**

**1. Commit the container changes to a local image**

# docker commit 7c6e7c6730b9 img01

sha256:6520e6642eeb4dc88fea260645e35888a775f396c826540948571060a16a2377

**2. Check if img01 has been created**

# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

img01 latest 6520e6642eeb 27 seconds ago 133MB

nginx latest 35c43ace9216 2 weeks ago 133MB

**3. Create a Public repository named as repo0603 in Docker Hub**

**4.Tag local repository to repository named repo0603 in Docker Hub**

# docker tag img01 chaitanyagaajula/repo0603:nginxlatest

**5.Check if image has been tagged**

# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

chaitanyagaajula/repo0603 nginximage 6520e6642eeb 2 minutes ago 133MB

img01 latest 6520e6642eeb 2 minutes ago 133MB

nginx latest 35c43ace9216 2 weeks ago 133MB

**6.Push the image to repository dockerhub**

# docker push chaitanyagaajula/repo0603:nginxlatest

The push refers to repository [docker.io/chaitanyagaajula/repo0603]

032fe9ab5172: Preparing

2acf82036f38: Preparing

9f65d1d4c869: Preparing

0f804d36244d: Preparing

9b23c8e1e6f9: Preparing

ffd3d6313c9b: Waiting

9eb82f04c782: Waiting

denied: requested access to the resource is denied

**7.Authenticate to dockerhub and again push**

# docker login

Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.

Username: chaitanyagaajula

Password:

WARNING! Your password will be stored unencrypted in /root/.docker/config.json.

Configure a credential helper to remove this warning. See

https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded

**8. Again push to dockerhub**

# docker push chaitanyagaajula/repo0603:nginxlatest

The push refers to repository [docker.io/chaitanyagaajula/repo0603]

032fe9ab5172: Pushed

2acf82036f38: Layer already exists

9f65d1d4c869: Layer already exists

0f804d36244d: Layer already exists

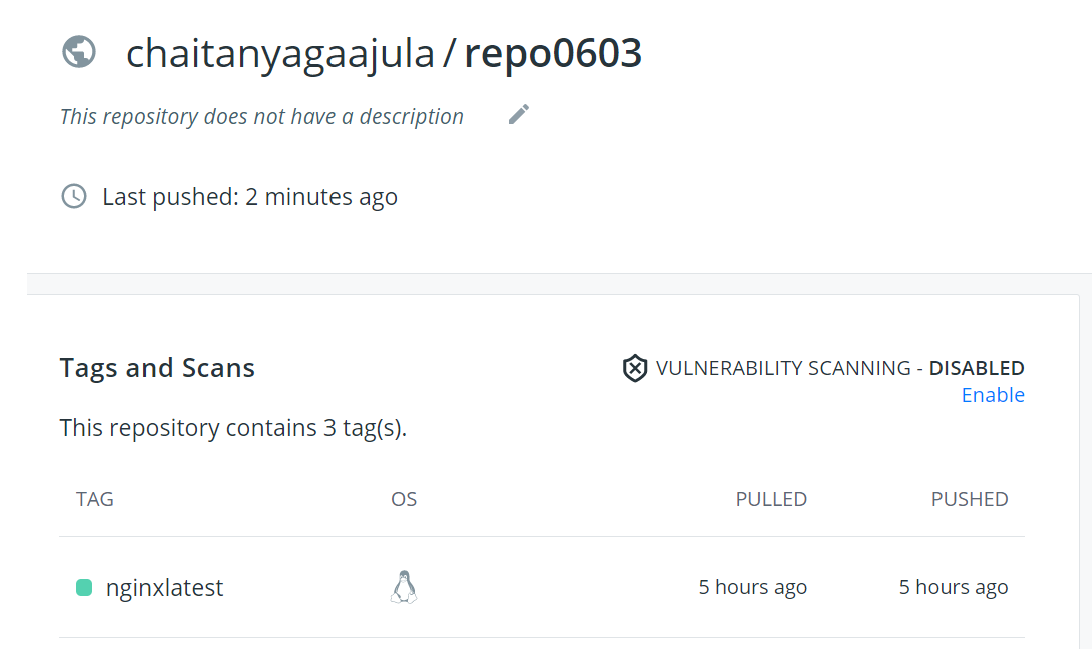
9b23c8e1e6f9: Layer already exists

ffd3d6313c9b: Layer already exists

9eb82f04c782: Layer already exists

nginximage: digest: sha256:c6e21f23264701cdd4a7f91f84ead23b63bf6c84ee70889e94075d4ae24518a2 size: 1777

**9. Check if image has been pushed to docker**



**10. Check if other team members are able to see the example.txt in the container which they created from the image.**

# docker pull chaitanyagaajula/repo0603:nginxlatest

# docker run -d chaitanyagaajula/repo0603:nginxlatest

# docker exec -it containername /bin/sh

ls

example.txt