Application Containerization And Orchestration Lab

Submitted By – Chitwan Singh
SAP ID – 500097009
Enrolment no. – R2142211291
Batch – DevOps B4

Submitted to

– Dr. Hitesh Kumar Sharma

Lab Exercise 1

Objective: Performing CRUD Operation on Containers

Tools required: Docker Configuration

Pre-requisites: Ubuntu Configuration, Docker

Performing CRUD Operation on Containers

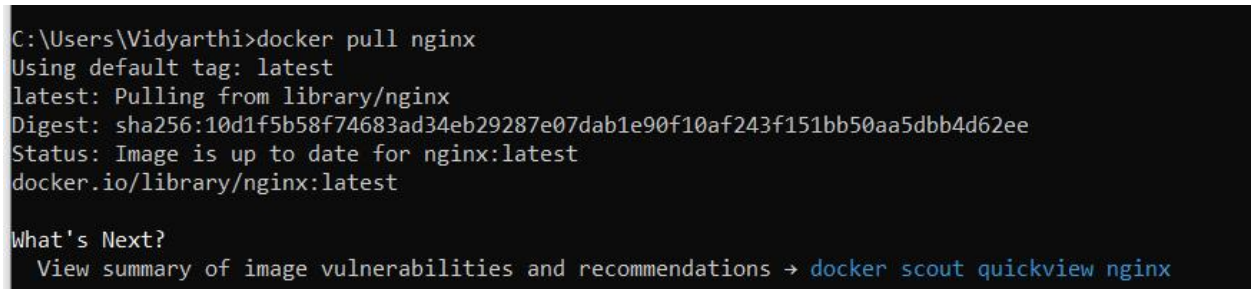
Steps to be followed:

1. Pulling a Docker image
2. Creating a new container
3. Stopping the container
4. Listing all the containers
5. Deleting the container
6. Removing the image

Step 1: Pulling a Docker image

1.1 Open the terminal and pull an image using the command:

docker pull nginx



1.2 List all the docker images to check the newly pulled *nginx* image:

docker images

```
C:\Users\Vidyarthi>docker images
REPOSITORY                                          TAG
              IMAGE ID       CREATED         SIZE
chitwan08/docker_experiment05                       latest
              f08efdd77cc2   4 hours ago     61MB
docker_experiment05                                 latest
              f08efdd77cc2   4 hours ago     61MB
myimage                                             1.0.0
              f08efdd77cc2   4 hours ago     61MB
my-web-app                                          1.0.0
              db4b11c375c1   2 days ago      168MB
alpine                                              latest
              b541f2080109   2 days ago      7.34MB
<none>                                              <none>
              96ab69a0b5c8   8 days ago      467MB
nginx                                               latest
              a6bd71f48f68   12 days ago     187MB
custom-jenkins-jdk17                                latest
              e7254eebb2da   2 weeks ago     994MB
redis                                               latest
              961dda256baa   3 weeks ago     138MB
```

Step 2: Creating a new container

2.1 Create a new container from the *nginx* image:

docker run -dt -p 81:81 nginx

```
C:\Users\Vidyarthi> docker run -dt -p 81:81 nginx
aec77374e69c24fc712116c10ee794e74e44c7163f170dd1179c339c6b195880
```

2.2 List all the running containers to check the newly created container. You can find various details like port of container, it's time of creation and ID.

docker ps

```
C:\Users\Vidyarthi> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                          NAMES
aec77374e69c   nginx     "/docker-entrypoint.…"   33 seconds ago   Up 32 seconds   80/tcp, 0.0.0.0:81->81/tcp   admiring_goldberg
```

Step 3: Stopping the container

3.1 Use the following command to stop the running container. ***(***You can also us the container ID to stop the container: *sudo docker stop CONTAINER_ID****)***

docker stop CONTAINER_NAME

```
C:\Users\Vidyarthi>docker rm -f aec77374e69c
aec77374e69c
```

3.2 Use the following command to list all the running containers and verify if the container has stopped running:

docker ps

```
C:\Users\Vidyarthi>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

3.3 You can start the container again and check the running containers. ***(***You can also us the container ID to start the container: *sudo docker start CONTAINER_ID****)***

sudo docker start CONTAINER_NAME

sudo docker ps

```
C:\Users\Vidyarthi>docker start e1f01d0f83f8
e1f01d0f83f8

C:\Users\Vidyarthi>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS     NAMES
e1f01d0f83f8   nginx     "/docker-entrypoint.…"   3 minutes ago   Up 4 seconds   80/tcp    frosty_bell
```

3.4 To start the container in interactive mode, use the –i and –t options.

docker run -it --name=Test_1 ubuntu

```
C:\Users\Vidyarthi> docker run -it --name=Test_1 ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
5e8117c0bd28: Pull complete
Digest: sha256:8eab65df33a6de2844c9aefd19efe8ddb87b7df5e9185a4ab73af936225685bb
Status: Downloaded newer image for ubuntu:latest
```

Step 4: Listing all the containers

4.1 Use the below command to list all the containers started and the once which are stopped:

docker ps -a

```
C:\Users\Vidyarthi>docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED         STATUS                      PORTS                    NAMES
18c809aaa062   ubuntu                               "/bin/bash"              3 minutes ago   Exited (130) 17 seconds ago                          Test_1
e1f01d0f83f8   nginx                                "/docker-entrypoint.…"   8 minutes ago   Up 5 minutes                80/tcp                   frosty_bell
e9af762ce938   redis                                "docker-entrypoint.s…"   23 minutes ago  Exited (130) 17 minutes ago                          loving_sutherland
ecd11475c740   chitwan08/docker_experiment05        "node index.js"          4 hours ago     Exited (0) 4 hours ago                               agitated_kirch
e78a72e2052f   alpine                               "ping -c1 db"            32 hours ago    Exited (0) 32 hours ago                              jovial_archimedes
25063dde8a46   katacoda/redis-node-docker-example   "npm start"              32 hours ago    Exited (255) 4 hours ago    0.0.0.0:3000->3000/tcp   recursing_robinson
422a1e3e6337   alpine                               "ping -c1 redis"         32 hours ago    Exited (0) 32 hours ago                              bold_burnell
0b2b6f3ba7a2   redis                                "docker-entrypoint.s…"   32 hours ago    Exited (255) 4 hours ago    6379/tcp                 redis
03b7d994acbe   my-web-app:1.0.0                     "ls /usr/local/apach…"   2 days ago      Exited (0) 2 days ago                                elated_curie
ff5fa2a80ee2   my-web-app:1.0.0                     "/bin/bash"              2 days ago      Exited (255) 35 hours ago   80/tcp                   romantic_gauss
```

4.2 To list the containers by their ID, use the below command

docker ps -aq

```
C:\Users\Vidyarthi>docker ps -aq
18c809aaa062
e1f01d0f83f8
e9af762ce938
ecd11475c740
e78a72e2052f
25063dde8a46
422a1e3e6337
0b2b6f3ba7a2
03b7d994acbe
ff5fa2a80ee2
```

You can see the containers with ID are listed.

4.3 To list the total file size of each container, use the below command:

docker ps -s

```
C:\Users\Vidyarthi>docker ps -s
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS         PORTS     NAMES         SIZE
e1f01d0f83f8   nginx     "/docker-entrypoint.…"   10 minutes ago   Up 7 minutes   80/tcp    frosty_bell   1.09kB (virtual 187MB)
```

4.4 To list the latest created containers, use the following command:

docker ps -l

```
C:\Users\Vidyarthi>docker ps -l
CONTAINER ID   IMAGE     COMMAND       CREATED          STATUS                      PORTS     NAMES
18c809aaa062   ubuntu    "/bin/bash"   8 minutes ago    Exited (130) 4 minutes ago            Test_1
```

Step 5: Deleting the container

5.1 Stop the running container and remove it using the following commands:

docker stop CONTAINER_NAME

sudo docker container rm CONTAINER_NAME

```
C:\Users\Vidyarthi>docker stop e1f01d0f83f8
e1f01d0f83f8

C:\Users\Vidyarthi>docker rm  e1f01d0f83f8
e1f01d0f83f8
```

Step 6: Removing the image

6.1 Remove the image using the command:

docker image rm nginx

```
C:\Users\Vidyarthi> docker image rm nginx
Untagged: nginx:latest
Untagged: nginx@sha256:10d1f5b58f74683ad34eb29287e07dab1e90f10af243f151bb50aa5dbb4d62ee
Deleted: sha256:a6bd71f48f6839d9faae1f29d3babef831e76bc213107682c5cc80f0cbb30866
Deleted: sha256:fe7723b2df19ccf75328cb1f39c90c2332679144231501f3d4d00f51b16c2867
Deleted: sha256:c0f3f17b019abbcfeb0c162054786ea8087ca792a2191a79f03040a8cd98f41d
Deleted: sha256:32cfe66e62a5f36abf128703007285e1a3b9078f5b33a367df1534399065cc70
Deleted: sha256:8713bfa322a66040e882b7822dc0c110a68cfafd3bb37332fdbb9426171d7ec9
Deleted: sha256:253c039db964b57be02d9bb0f3d6916b7948687b4f6f4fc681644a419a47979d
Deleted: sha256:2c235ef4cca1fbb74e3a7aa47e654fe943bc0c2becbdd4d6af7cbf9e375fe08a
```