

Lab Exercise 1

NAME- RADHIKA RAJESH THAKKAR

SAPID- 50098212

ROLL NO- R2142211496

BATCH- DEVOPS B4

Performing CRUD Operation on Containers

Objective: Performing CRUD Operation on Containers

Tools required: Docker Configuration

Pre-requisites: Ubuntu Configuration, Docker

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:

sudo docker pull nginx

```
MINGW64; c:/Users/RADHIKA
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
a378f10b3218: Pulling fs layer
5b5e4b85559a: Pulling fs layer
508092f60780: Pulling fs layer
59c24706ed13: Pulling fs layer
1a8747e4a8f8: Pulling fs layer
59c24706ed13: Waiting
ad85f053b4ed: Pulling fs layer
3000e3c97745: Pulling fs layer
1a8747e4a8f8: Waiting
3000e3c97745: Waiting
ad85f053b4ed: Waiting
508092f60780: Verifying Checksum
508092f60780: Download complete
59c24706ed13: Download complete
1a8747e4a8f8: Verifying Checksum
1a8747e4a8f8: Download complete
ad85f053b4ed: Download complete
3000e3c97745: Verifying Checksum
3000e3c97745: Download complete
```

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

sudo docker images

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
nginx latest 593aee2afb64 15 hours ago 187MB

RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

Step 2: Creating a new container

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

sudo docker run -dt -p 81:81 nginx

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker run -dt -p 81:81 nginx
bce81ca04f93ff9b1847e3ba5877b8826e9a52d231dfbea5e66476353d0548ce

RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

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.

sudo docker ps

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS
PORTS         NAMES
bce81ca04f93   nginx     "/docker-entrypoint..." 27 seconds ago Up 26 seconds
80/tcp, 0.0.0.0:81->81/tcp interesting_goldstine

RADHIKA@RADHIKA MINGW64 ~ (master)
$ |
```

Step 3: Stopping the container

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

sudo docker stop CONTAINER_NAME

Note: Replace CONTAINER_NAME with the name of the newly created container. In this case CONTAINER_NAME is stoic_darwin. The container name may differ from the one shown in the image below.

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker stop interesting_goldstine
interesting_goldstine

RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

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

sudo docker ps

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS         NAMES
RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

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

sudo docker start CONTAINER_NAME

sudo docker ps

Note: Replace CONTAINER_NAME with the name of the newly created container. In this case CONTAINER_NAME is stoic_darwin. The container name may differ from the one shown in the image below.

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker start interesting_goldstine
interesting_goldstine

RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS
PORTS         NAMES
bce81ca04f93   nginx    "/docker-entrypoint...." 5 minutes ago  Up 27 seconds
80/tcp, 0.0.0.0:81->81/tcp  interesting_goldstine

RADHIKA@RADHIKA MINGW64 ~ (master)
$ |
```

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

sudo docker run -it --name=Test_1 ubuntu

```
C:\Users\RADHIKA>docker run -it --name=Test_1 ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
aece8493d397: Pull complete
Digest: sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e496e5c1f05f
Status: Downloaded newer image for ubuntu:latest
root@d7e37ac01b5c:/# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@d7e37ac01b5c:/# |
```

Step 4: Listing all the containers

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

sudo docker ps -a

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS      PORTS
bce81ca04f93   nginx    "/docker-entrypoint...." 10 minutes ago  Up 5 minutes  80/tcp, 0.0.0.0:81->81/tcp
interesting_goldstine
0a2f1a74db3c   eea7b3dcba7e "/docker-entrypoint...." 2 months ago  Exit (0) 2 months ago
inspiring_williamson

RADHIKA@RADHIKA MINGW64 ~ (master)
$ |
```

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

sudo docker ps -aq

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps -aq
d7e37ac01b5c
bce81ca04f93
0a2f1a74db3c
RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

You can see the containers with ID are listed.

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

sudo docker ps -s

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps -s
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS
PORTS         NAMES      SIZE
d7e37ac01b5c   ubuntu    "/bin/bash"             3 minutes ago Up 3 minutes
Test_1         0B (virtual 77.8MB)
bce81ca04f93   nginx     "/docker-entrypoint.    12 minutes ago Up 7 minutes
80/tcp, 0.0.0.0:81->81/tcp interesting_goldstine 1.09kB (virtual 187MB)
RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

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

sudo docker ps -l

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker ps -l
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
NAMES
d7e37ac01b5c   ubuntu    "/bin/bash"             3 minutes ago Up 3 minutes
Test_1
RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

Step 5: Deleting the container

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

sudo docker stop CONTAINER_NAME

sudo docker container rm CONTAINER_NAME

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker stop interesting_goldstine
interesting_goldstine

RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker container rm interesting_goldstine
interesting_goldstine

RADHIKA@RADHIKA MINGW64 ~ (master)
$
```

Note: Replace CONTAINER_NAME with the name of the newly created container. In this case CONTAINER_NAME is sweet_brown. The container name may differ from the one shown in the image below.

Step 6: Removing the image

6.1 Remove the image using the command:

sudo docker image rm nginx

```
RADHIKA@RADHIKA MINGW64 ~ (master)
$ docker image rm nginx
Untagged: nginx:latest
Untagged: nginx@sha256:add4792d930c25dd2abf2ef9ea79de578097a1c175a16ab25814332fe33622de
Deleted: sha256:593aee2afb642798b83a85306d2625fd7f089c0a1242c7e75a237846d80aa2a0
Deleted: sha256:5f40d2c2af6bd6c289f284b2ffb1022db6984edf7005ac791c3723730dd40111
Deleted: sha256:bd85a340ea268d71b49c16a693f1764267228a5787fb0da2b81dc6c19fa9f97f
Deleted: sha256:89fafeeb9db96ca37f3214c0d14ce160ce82041b284ecf3a26641d3b3b5359e5
Deleted: sha256:ec22d9ca60728c69acca7909d96922dfe5ceb4665ecd6c91b3282f18c3a58424
Deleted: sha256:3745a2d825305d88cd6b4aed9b5b2de8eb2a36d237dd471368d18cf5d96e907f
Deleted: sha256:87d5589860cd08d504e32e6e17498af935f95dca535bbb5439ea50049047dfb4
Deleted: sha256:cb4596cc145400fb1f2aa56d41516b39a366ecdee7bf3f9191116444aacd8c90

RADHIKA@RADHIKA MINGW64 ~ (master)
$
```