

Lab Exercise 1

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

```
C:\Users\Sudhanshu>docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
a378f10b3218: Pull complete
4dfff0708538: Pull complete
2135e49ace4b: Pull complete
c843f6b280ce: Pull complete
6f35ab6f1400: Pull complete
6c538b49fa4a: Pull complete
d57731fb9008: Pull complete
Digest: sha256:b4af4f8b6470febf45dc10f564551af682a802eda1743055a7dfc8332dffa595
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
```

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

sudo docker images

```
C:\Users\Sudhanshu>docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	bc649bab30d1	8 days ago	187MB
nopass	latest	ca8dada2b71d	2 weeks ago	168MB
ubuntu	latest	e4c58950181a	2 weeks ago	77.8MB
hubproxy.docker.internal:5555/docker/desktop-kubernetes	kubernetes-v1.27.2-cni-v1.2.0-critools-v1.27.0-cri-dockerd-v0.3.2-1-debian	c763812a4530	4 months ago	418MB
registry.k8s.io/kube-apiserver	v1.27.2	c5b13e4f7806	5 months ago	121MB
registry.k8s.io/kube-scheduler	v1.27.2	89e70da428d2	5 months ago	58.4MB
registry.k8s.io/kube-controller-manager	v1.27.2	ac2b7465ebba	5 months ago	112MB
registry.k8s.io/kube-proxy	v1.27.2	b8aa50768fd6	5 months ago	71.1MB
docker/desktop-vpnkit-controller	dc331cb22850be0cdd97c84a9cfecaf44a1afb6e	556098075b3d	5 months ago	36.2MB
registry.k8s.io/coredns/coredns	v1.10.1	ead0a4a53df8	8 months ago	53.6MB
registry.k8s.io/etcd	3.5.7-0	86b6af7dd652	8 months ago	296MB
registry.k8s.io/pause	3.9	e6f101688397	12 months ago	744kB
docker/desktop-storage-provisioner	v2.0	99f89471f470	2 years ago	41.9MB

```
C:\Users\Sudhanshu>
```

Step 2: Creating a new container

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

sudo docker run -dt -p 81:81 nginx

```
C:\Users\Sudhanshu>docker run -dt -p 81:81 nginx
9f709f7f795bf0fe19c4e9e9f0ea23f0d31d74a3c4dac88814478979f5767613
```

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

```
C:\Users\Sudhanshu>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
9f709f7f795b	nginx	"/docker-entrypoint..."	24 seconds ago	Up 23 seconds	80/tcp, 0.0.0.0:81->81/tcp	adoring_liskov

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.

```
C:\Users\Sudhanshu>docker stop adoring_liskov
adoring_liskov
```

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

sudo docker ps

```
C:\Users\Sudhanshu>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 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.

```
C:\Users\Sudhanshu>docker start adoring_liskov
adoring_liskov
C:\Users\Sudhanshu>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
9f709f7f795b   nginx     "/docker-entrypoint..." 3 minutes ago    Up 12 seconds    80/tcp, 0.0.0.0:81->81/tcp    adoring_liskov
```

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

sudo docker run -it --name=Test_1 ubuntu

```
F:\dockerlab>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@fe34ce64a0:/#
```

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

```
C:\Users\Sudhanshu>docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
0311537acfb9   ubuntu    "/bin/bash"             58 seconds ago    Up 58 seconds    80/tcp, 0.0.0.0:81->81/tcp    Test_1
9f709f7f795b   nginx     "/docker-entrypoint..." 5 minutes ago    Up About a minute    80/tcp, 0.0.0.0:81->81/tcp    adoring_liskov
601ac6f8b939   nopath    "/bin/bash"             2 weeks ago      Exited (255) 36 minutes ago    80/tcp    crazy_pike
```

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

sudo docker ps -aq

```
C:\Users\Sudhanshu>docker ps -aq
0311537acfb9
9f709f7f795b
601ac6f8b939
```

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

```
C:\Users\Sudhanshu>docker ps -s
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES	SIZE
8311537acf89	ubuntu	"/bin/bash"	About a minute ago	Up About a minute		Test_1	0B (virtual 77.8MB)
9f789f7f795b	nginx	"/docker-entrypoint..."	5 minutes ago	Up 2 minutes	80/tcp, 0.0.0.0:81->81/tcp	adoring_liskov	1.09kB (virtual 187 MB)

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

sudo docker ps -l

```
C:\Users\Sudhanshu>docker ps -l
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
8311537acf89	ubuntu	"/bin/bash"	About a minute ago	Up About a minute		Test_1

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

```
C:\Users\Sudhanshu>docker stop Test_1
Test_1

C:\Users\Sudhanshu>docker container rm Test_1
Test_1
```

Step 6: Removing the image

6.1 Remove the image using the command:

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.

sudo docker image rm nginx

```
C:\Users\Sudhanshu>docker image rm nginx
Untagged: nginx:latest
Deleted: sha256:b4af4f8b6470feb745dc10f564551af682a802eda1743855a7dfc8332dffa595
Deleted: sha256:bc649bab38d158c18a84031a7f54c99a8c31069c7bc324a7899d7125d59cc973
Deleted: sha256:c6f480996a203ed077606cce624f944b041449833e2db3f7d19fe22974fb965b
Deleted: sha256:e4347a01432c5f4350b041632f5703c3dd47de2ec68547b9339d11ea44708389
Deleted: sha256:9d40098fc19fdff9c74fd3c2c0ff49b7d9d04b5d7806d0843d32055d769a
Deleted: sha256:165ae0ef2ddd33b6d5a7f206633b9b0c30cd94ff18a4ed5c3aeb59bf28388526
Deleted: sha256:06dabb44ac4d1f0b5544255e944f15a939178d77aff60a5b296e38bd8743efeb
Deleted: sha256:ee220599571f649e0fb74b40db1615a4c9c1355ac912f9e70087b695617af352
Deleted: sha256:cb4596cc145408fb1f2aa56d41516b39a366ecdee7bf3f9191116444aacd8c98
C:\Users\Sudhanshu>
```