

Experiment-1.4

Student Name:Aditi Pandey

Branch: CSE(DevOps)

Semester:5 Sem

Subject Name:Docker and Kuberensats

UID:22BDO10031

Section/Group:22BCD1/A

Date of Performance:1sept2024

Subject Code: 22CSH-343

1.Aim/Overview of the practical: To manage Volumes and Containers for storing and retrieval of data in Docker.

2.Apparatus: VM ware, Linux , docker

3.Steps for experiment/practical:

1)Create the volume and mount the created volume inside a container's file system. This allows the container to store and retrieve data from that volume.

Syntax: docker volume create <vol_name>

```
Aditis-MacBook-Air:vol aditipandey$ docker volume create my-volu
my-volu
Aditis-MacBook-Air:vol aditipandey$ docker volume ls
DRIVER      VOLUME NAME
local       91e3a2c57a5030b090749cc8f3b6a0b905a6a5a902e9b2ccce2b0feb96be95ab
local       my-vol
local       my-volu
```

2) Mounting volume with a container:

docker run -it -v <volume-path-in-local-machine>:<dest-path-in-container> <image-name>

```
Aditis-MacBook-Air:vol aditipandey$ docker run -it --name new-container-name -v my-volu:/tmp ubuntu
bash
root@ef21058c8085:/# ls
bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
boot  etc  lib   media  opt  root  sbin  sys  usr
```

To verify if the volume has been successfully mounted or not, you can move to the destination directory inside the Container.

```
root@ef21058c8085:/# cd tmp/  
root@ef21058c8085:/tmp# s  
bash: s: command not found  
root@ef21058c8085:/tmp# ls  
root@ef21058c8085:/tmp# mkdir exp  
root@ef21058c8085:/tmp# cd exp  
root@ef21058c8085:/tmp/exp# touch hello.txt
```

- Listing all the docker volumes:

Syntax: docker volume ls

```
Aditis-MacBook-Air:vol aditipandey$ docker volume ls  
DRIVER      VOLUME NAME  
local       91e3a2c57a5030b090749cc8f3b6a0b905a6a5a902e9b2ccce2b0feb96be95ab  
local       my-vol  
local       my-volu
```

- Inspecting docker volumes: Syntax: docker volume inspect <volume-name>

```
Aditis-MacBook-Air:vol aditipandey$ docker volume inspect my-volu  
[  
  {  
    "CreatedAt": "2024-09-15T18:21:19Z",  
    "Driver": "local",  
    "Labels": null,  
    "Mountpoint": "/var/lib/docker/volumes/my-volu/_data",  
    "Name": "my-volu",  
    "Options": null,  
    "Scope": "local"  
  }  
]
```

- Removing specific docker volume or all the volumes

Syntax: docker volume rm <volume-name>

```
Aditis-MacBook-Air:vol aditipandey$ docker stop $(docker ps -aq)
ef21058c8085
6b5607c3bc47
38ede92d64f2
88cd28e4c5c5
b43ea509849c
59636d5257ec
ce4f4439f795
b843d9db73fd
5fa2147f1e68
```

Syntax: docker volume rm \$(sudo docker volume ls -q)

```
Aditis-MacBook-Air:vol aditipandey$ docker rm $(docker ps -aq)
ef21058c8085
6b5607c3bc47
38ede92d64f2
88cd28e4c5c5
b43ea509849c
59636d5257ec
ce4f4439f795
b843d9db73fd
5fa2147f1e68
```

```
Aditis-MacBook-Air:vol aditipandey$ docker volume rm my-volu
my-volu
Aditis-MacBook-Air:vol aditipandey$ docker volume ls
DRIVER      VOLUME NAME
local       91e3a2c57a5030b090749cc8f3b6a0b905a6a5a902e9b2ccce2b0feb96be95ab
local       my-vol
Aditis-MacBook-Air:vol aditipandey$
```

4.Result/Output/Writing Summary:

1. `Volumes` are stored in the host filesystem that is managed by Docker.
2. `Bind mounts` are stored anywhere on the host system.
3. `tmpfs mounts` are stored in the host memory only.
4. Originally, the `— mount` flag was used for Docker Swarm services and the `— volume` flag was used for standalone containers.
5. If the container no longer exists, the data is lost

Learning outcomes (What I have learnt):

1. I have learnt the concept of containerization.
2. I have learnt to configure Docker to work with different environments.
3. I have learnt how to build docker images using Dockerfile.
4. I have learnt the purpose of Docker volumes and their role in data persistence.
5. I have learnt how to use Docker Hub to pull and push Docker images.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			