**Name :** Aarushi **Sap lD:** 500105028 **Roll no:** R2142220004

# Lab Exercise 3: Working with Docker Volumes

**Objective:**

- Learn how to create and manage Docker volumes.
- Understand how Docker volumes can be used to persist data across container restarts.
- Practice mounting Docker volumes to containers.

**Prerequisites:**

- Docker installed on your system.
- Basic understanding of Docker commands and container concepts.

**Step 1: Create a Docker Volume**

Create a new Docker volume:

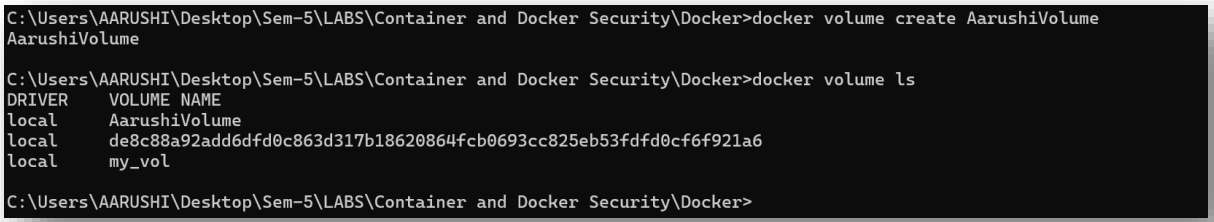
```
docker volume create my_data_volume
```

This command creates a Docker volume named my_data_volume.

Verify that the volume was created:

```
docker volume ls
```

You should see my_data_volume listed among the volumes.



## Step 2: Run a Container with the Volume Mounted

Run an Nginx container with the volume mounted:

docker run -d --name my_nginx -v my_data_volume:/usr/share/nginx/html -p 8008:80 nginx

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker run -d --name my_nginx -v AarushiVolume:
/usr/share/nginx/html -p 8008:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
a480a496ba95: Already exists
f3ace1b8ce45: Pull complete
11d6fdd0e8a7: Pull complete
f1091da6fd5c: Pull complete
40eea07b53d8: Pull complete
6476794e50f4: Pull complete
70850b3ec6b2: Pull complete
Digest: sha256:28402db69fec7c17e179ea87882667f1e054391138f77ffaf0c3eb388efc3ffb
Status: Downloaded newer image for nginx:latest
ae5a469ef1a08a25f16c5c307be1d9d93ad6e15f03fc320e411025cc6c291b44

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>
```


This command starts an Nginx container named my_nginx and mounts the my_data_volume volume to the /usr/share/nginx/html directory inside the container.

Verify that the container is running:

docker ps

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                    NAMES
ae5a469ef1a0   nginx     "/docker-entrypoint.…"   40 seconds ago   Up 37 seconds   0.0.0.0:8008->80/tcp     my_nginx
1800e4710ff3   redis     "docker-entrypoint.s…"   4 days ago       Up 4 days       6379/tcp                 Aarushi_redis
```

You should see my_nginx listed as one of the running containers.

## Step 3: Interact with the Volume

Create a simple HTML file in the volume:

docker exec -it my_nginx bash

echo "<h1>Hello, Docker Volume</h1>" > /usr/share/nginx/html/index.html

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker exec -it my_nginx bash
root@ae5a469ef1a0:/# echo "<h1>Hello, Docker Volume</h1>" > /usr/share/nginx/html/index.html
```

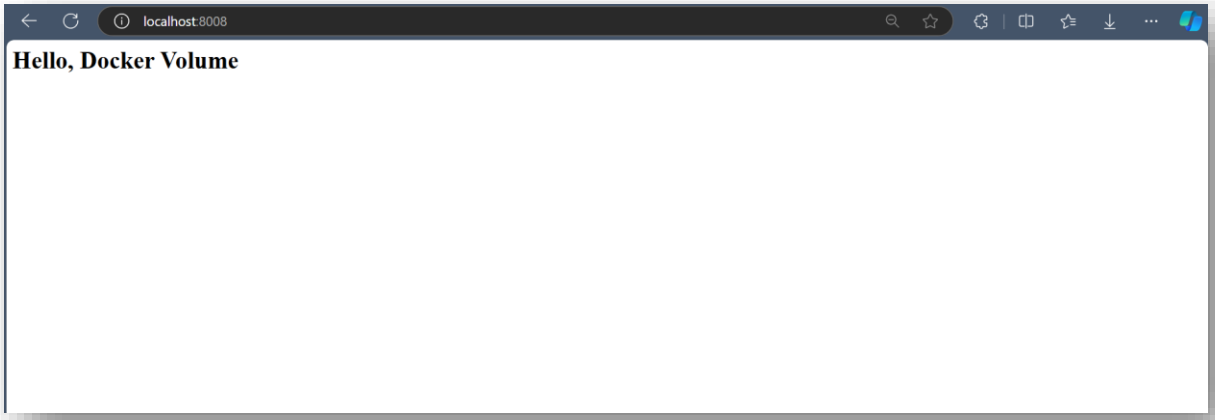
exit

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker exec -it my_nginx bash
root@ae5a469ef1a0:/# ls
bin                   etc    mnt   sbin  var
boot                  home   opt   srv
dev                   lib    proc  sys
docker-entrypoint.d   lib64  root  tmp
docker-entrypoint.sh  media  run   usr
root@ae5a469ef1a0:/# cd usr/share/nginx/html
root@ae5a469ef1a0:/usr/share/nginx/html# cat index.html
<h1>Hello, Docker Volume</h1>
root@ae5a469ef1a0:/usr/share/nginx/html# exit
```

This command creates an HTML file inside the /usr/share/nginx/html directory, which is backed by my_data_volume.

Access the Nginx server to see your file: Open a browser and navigate to http://localhost:8008. You should see the message "Hello, Docker Volume!" displayed

on the page.




## Step 4: Test Data Persistence

Stop and remove the container:

docker stop my_nginx

docker rm my_nginx

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker stop my_nginx
my_nginx

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker rm my_nginx
my_nginx

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED      STATUS      PORTS      NAMES
1800e4710ff3   redis     "docker-entrypoint.s…"   4 days ago   Up 4 days   6379/tcp   Aarushi_redis
```

Run a new Nginx container using the same volume:

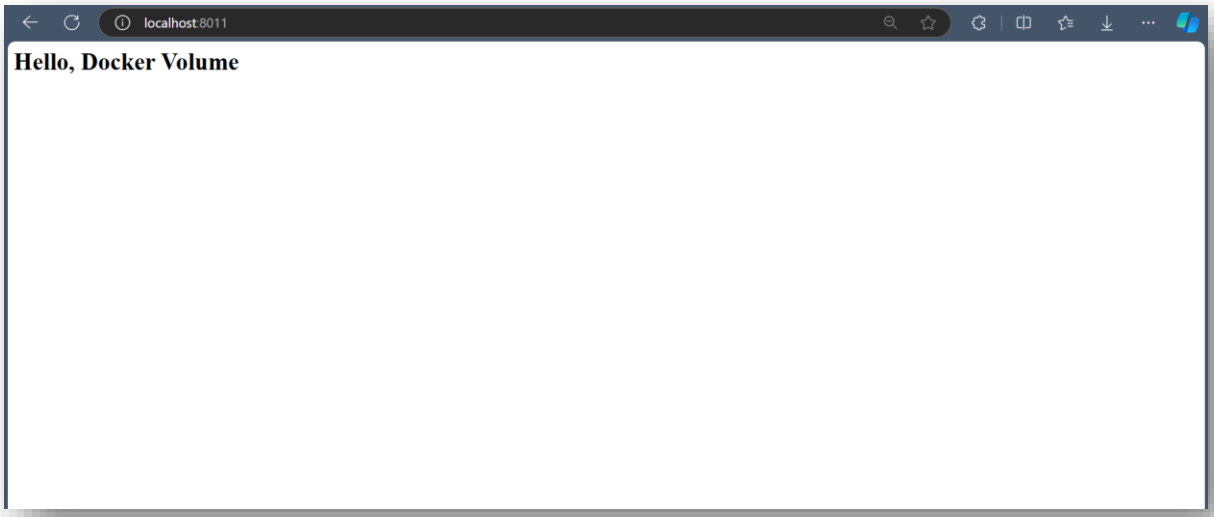
```
docker run -d -p 8011:80 -v my_data_volume:/usr/share/nginx/html nginx
```

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker run -d -p 8011:80 -v AarushiVolume:/usr/share/nginx
/html nginx
6433225cd938164d64c1b8279c8eda914a58df92859fbb7b284c1f4ff68bdf6b

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS
 NAMES
6433225cd938   nginx     "/docker-entrypoint.…"   27 seconds ago   Up 25 seconds   0.0.0.0:8011->80/tcp   compassionate_jang
1800e4710ff3   redis     "docker-entrypoint.s…"   4 days ago       Up 4 days       6379/tcp
 Aarushi_redis

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker exec -it 6433225cd938 bash
root@6433225cd938:/# cd /usr/share/nginx/
root@6433225cd938:/usr/share/nginx# cd html/
root@6433225cd938:/usr/share/nginx/html#
```

Access the Nginx server again: Navigate to http://localhost in your browser. You should still see the "Hello, Docker Volume!" message, demonstrating that the data persisted across container instances.




## Step 5: Clean Up

Stop and remove the container:

```
docker stop new_nginx
```

```
docker rm new_nginx
```

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker stop 6433225cd938
6433225cd938

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED      STATUS      PORTS      NAMES
1800e4710ff3   redis     "docker-entrypoint.s…"   4 days ago   Up 4 days   6379/tcp   Aarushi_redis

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker images ls
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker rm 6433225cd938
6433225cd938

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>
```

Remove the Docker volume:

```
docker volume rm my_data_volume
```

Verify that the volume is removed:

```
docker volume ls
```

Ensure that my_data_volume is no longer listed.

```
C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker rm 6433225cd938
6433225cd938

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker volume ls
DRIVER    VOLUME NAME
local     AarushiVolume
local     de8c88a92add6dfd0c863d317b18620864fcb0693cc825eb53fdfd0cf6f921a6
local     my_vol

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker volume rm AarushiVolume
AarushiVolume

C:\Users\AARUSHI\Desktop\Sem-5\LABS\Container and Docker Security\Docker>docker volume ls
DRIVER    VOLUME NAME
local     de8c88a92add6dfd0c863d317b18620864fcb0693cc825eb53fdfd0cf6f921a6
local     my_vol
```