Name: Bhavesh Sanjiv Kapur

SapID: 500105635

Roll no. R2142220057

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

Docker volume ls #(to see the list of volumes )
```

```
(base) ➜  ~ docker volume create volumeBhavesh
volumeBhavesh
(base) ➜  ~ docker volume ls
DRIVER    VOLUME NAME
local     0ae4ad30cd2b1dd56f7e698bc93cdbe4b368540ca1f0c7159816205c32e56cc0
local     0efcd0c1093f345927a6ad6c7710ba9a09aed8289101555dedbfa34e97279e9a
local     4b931dacbce5bdc3e997ae5eae2a681273a6633bf95482e47b01774ee8069bc6
local     7e9557c5d1e7c9eb0c8c2e46c6f727dac6be44f9f1785622435da4b679efe6a5
local     049833bc5a7c9828c4d042d79c28dfb6793620bc5f2f8b3a8c0724d4d24ebfd8
local     d036f39a36132dd8281dec905fe75522b4eb53c8e46ab6b67612b4f6688b0809
local     volumeBhavesh
(base) ➜  ~ _
```

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.

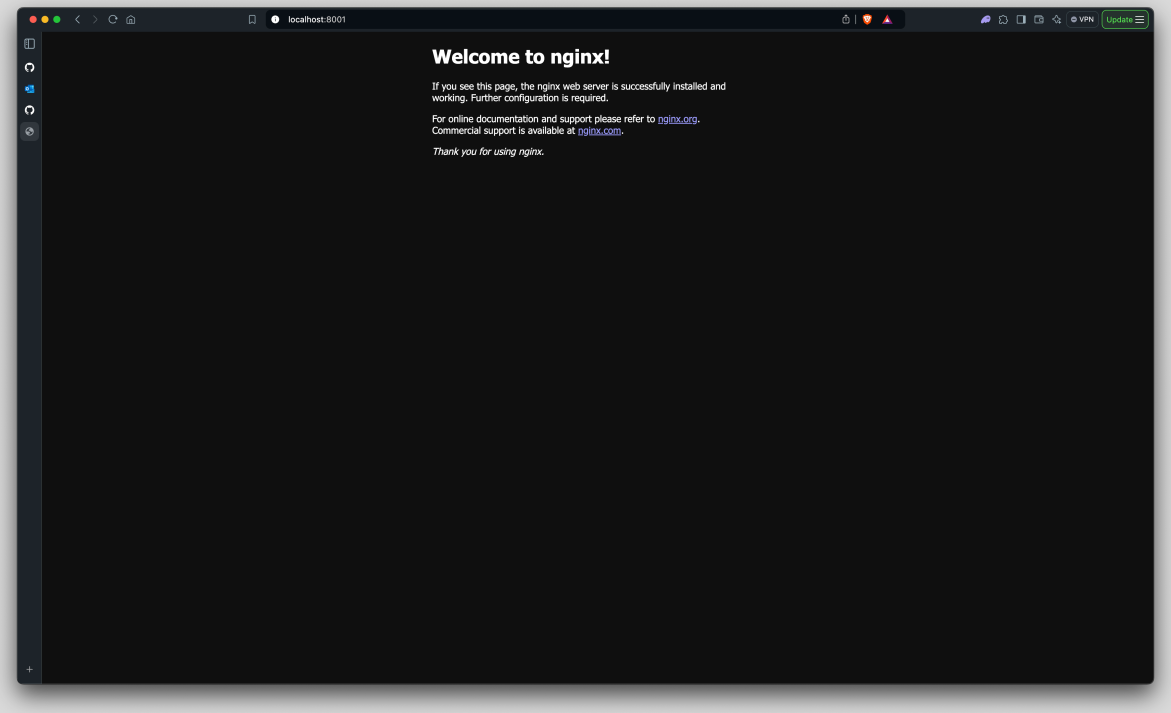
Step 2: Run a Container with the Volume Mounted

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

```
(base) ➜  ~ docker run -d --name nginxBhavesh -v volumeBhavesh:/usr/share/nginx/html -p 8001:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
92c3b3500be6: Pull complete
ee57511b3c68: Pull complete
33791ce134bf: Pull complete
cc4f24efc205: Pull complete
3cad04a21c99: Pull complete
486c5264d3ad: Pull complete
b3fd15a82525: Pull complete
Digest: sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3
Status: Downloaded newer image for nginx:latest
65d82be070ed414fd115791aaecccd1ce0887f00dd9ed88135b907ddb6686391
(base) ➜  ~ _
```

Nginx container running on port 8001



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

```
(base) ➜  ~ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS                   NAMES
65d82be070ed   nginx     "/docker-entrypoint.…"   About a minute ago   Up About a minute   0.0.0.0:8001->80/tcp   nginxBhavesh
(base) ➜  ~ _
```

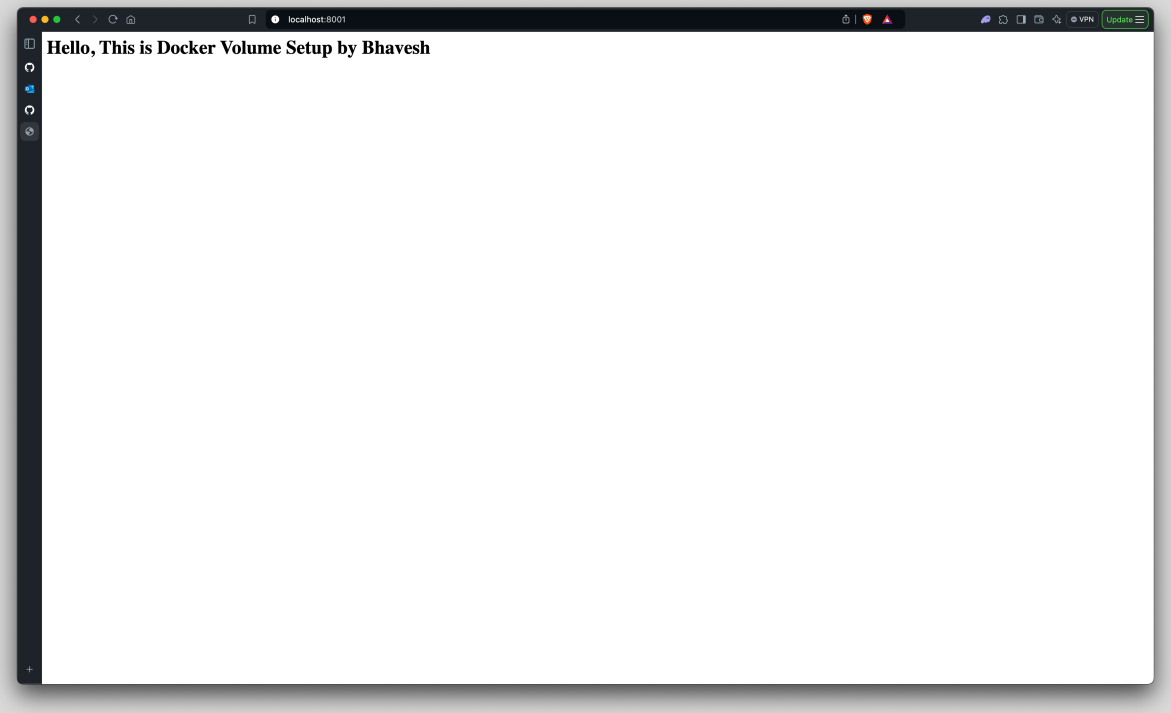
You should see my_nginx listed as one of the running containers.

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

exit
```

```
(base) ➜  ~ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS                    NAMES
65d82be070ed   nginx     "/docker-entrypoint.…"   About a minute ago   Up About a minute   0.0.0.0:8001->80/tcp   nginxBhavesh
(base) ➜  ~ docker exec -it nginxBhavesh bash
root@65d82be070ed:/# echo "<h1>Hello, This is Docker Volume Setup by Bhavesh </h1>" > /usr/share/nginx/html/index.html
root@65d82be070ed:/# cd /usr/share/nginx/html
root@65d82be070ed:/usr/share/nginx/html# cat index.html
<h1>Hello, This is Docker Volume Setup by Bhavesh </h1>
root@65d82be070ed:/usr/share/nginx/html# _
```



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

```
(base) ➜  ~ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                   NAMES
65d82be070ed   nginx     "/docker-entrypoint.…"   5 minutes ago   Up 5 minutes   0.0.0.0:8001->80/tcp   nginxBhavesh
(base) ➜  ~ docker stop nginxBhavesh
nginxBhavesh
(base) ➜  ~ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
(base) ➜  ~ _
```

docker rm my_nginx

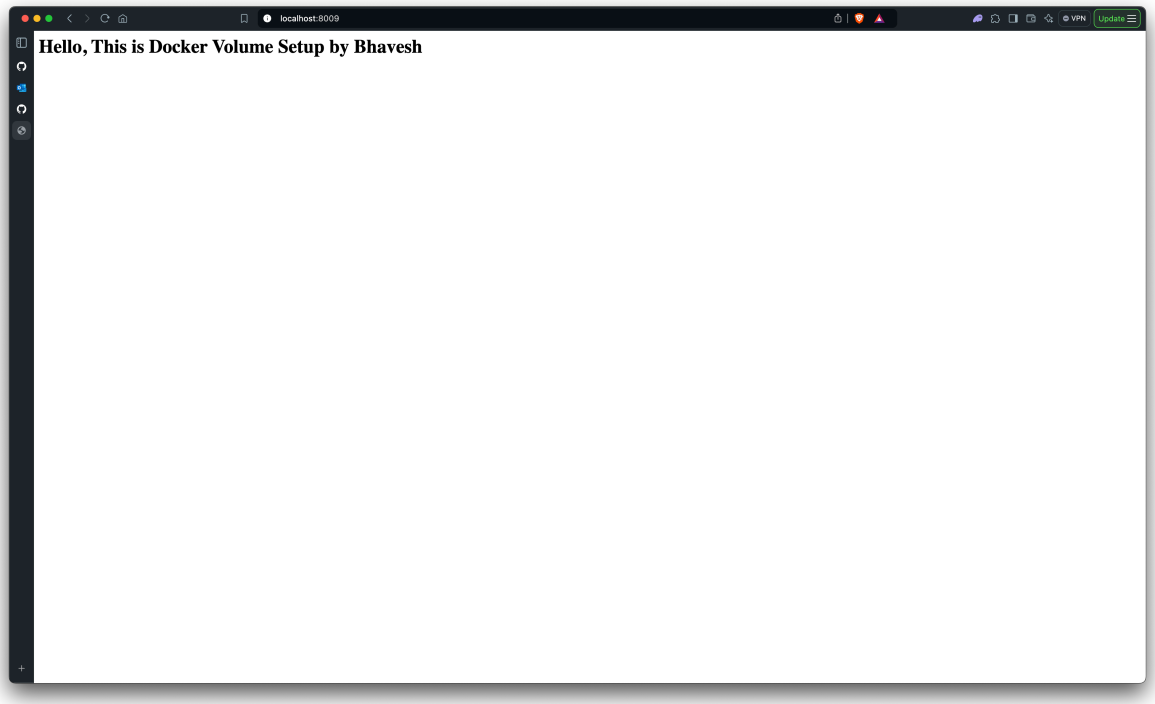
```
(base) ➜  ~ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
(base) ➜  ~ docker rm nginxBhavesh
nginxBhavesh
(base) ➜  ~ docker images ls
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
(base) ➜  ~ _
```

Run a new Nginx container using the same volume:

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

```
(base) ➜  ~ docker run -d -p 8009:80 -v volumeBhavesh:/usr/share/nginx/html nginx
3ad3ed772176bdf154ef402b5bc425c0a45516ca5f66578feff5dae171041e2f
(base) ➜  ~ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                   NAMES
3ad3ed772176   nginx     "/docker-entrypoint.…"   3 seconds ago   Up 2 seconds   0.0.0.0:8009->80/tcp   youthful_newton
(base) ➜  ~ docker exec -it 3ad3ed772176 bash
root@3ad3ed772176:/# cd /usr/share/nginx/
root@3ad3ed772176:/usr/share/nginx# cd html/
root@3ad3ed772176:/usr/share/nginx/html# cat index.html
<h1>Hello, This is Docker Volume Setup by Bhavesh </h1>
root@3ad3ed772176:/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:

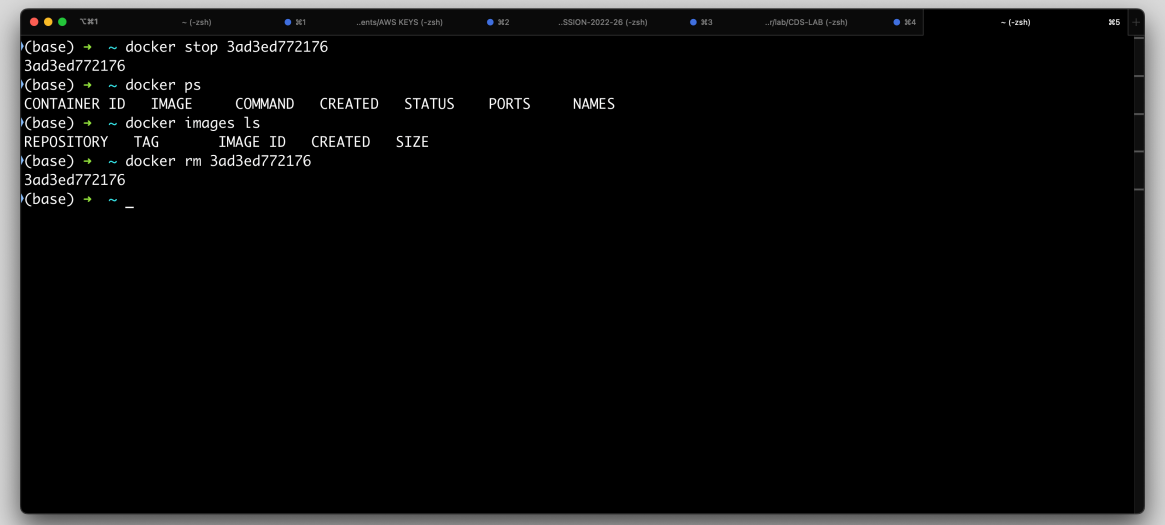
```
docker stop new_nginx

docker rm new_nginx
```

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.