Name: Raghav Kamboj

SAP ID: 500102126

Roll no. R2142220137

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.

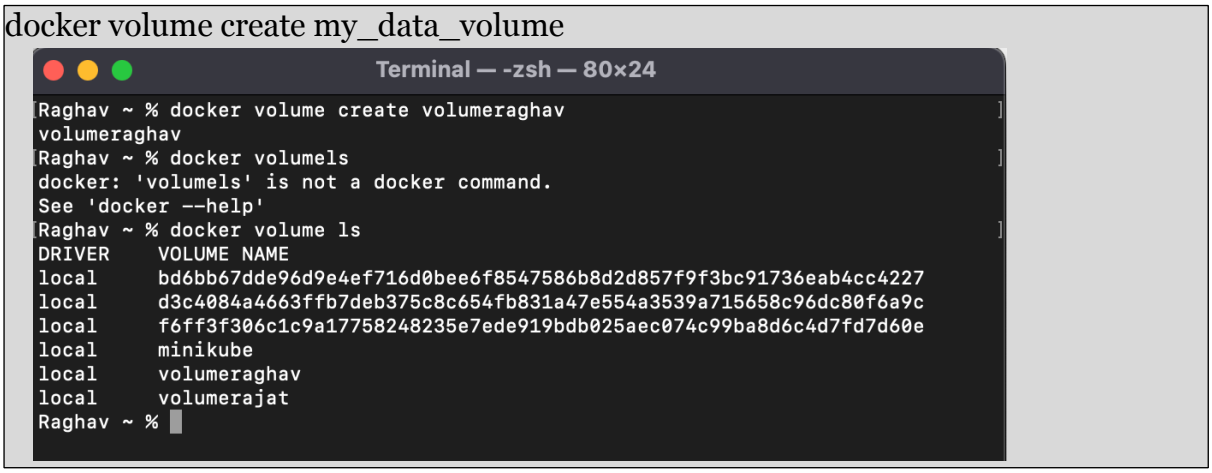
Prerequisites:

- 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)

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.

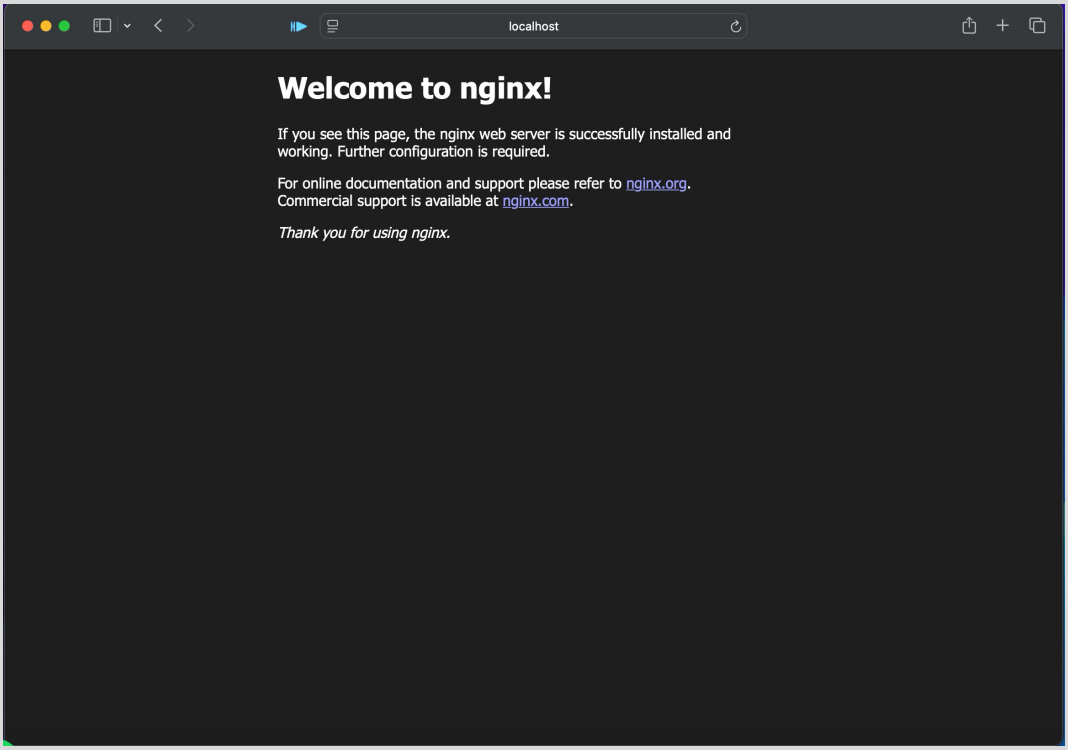
Step 2: Run a Container with the Volume Mounted

Run an Nginx container with the volume mounted: docker run -d --name

8008:80 nginx

```
Raghav ~ % docker run -d --name nginxRaghav -v volumeraghav:/usr/share/nginx/html
l -p 8008:80 nginx
e2e6442e2b01a32211dd15e9563e5c84e6345ac4fb9905c6f67e6e66d7ca347f
Raghav ~ %
```

Nginx container running on port 8001



my_nginx -v my_data_volume:/usr/share/nginx/html -p

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
```

Step 3: Interact with the Volume

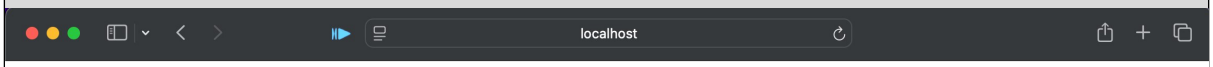
Create a simple HTML file in the volume:

```
docker exec -it my_nginx bash

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

exit
```

```
[Raghav ~ % docker ps                                                                                ]
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS
        PORTS                    NAMES
e2e6442e2b01   nginx          "/docker-entrypoint.…"   3 minutes ago    Up 3 min
utes    0.0.0.0:8008->80/tcp     nginxRaghav
4b663aeb9c71   redis:latest   "docker-entrypoint.s…"   7 minutes ago    Up 7 min
utes    6379/tcp                 redisrajatkaboj
56824fdef902   redis          "docker-entrypoint.s…"   15 minutes ago   Up 15 mi
nutes   6379/tcp                 ecstatic_mclean
[Raghav ~ % docker exec -it nginxRaghav bash                                                         ]
[root@e2e6442e2b01:/# echo "<h1>Hello, this is Docker Volume Setup by Raghav </h1]
>"> /usr/share/nginx/html/index.html
[root@e2e6442e2b01:/# cd /usr/share/nginx/html                                                       ]
[root@e2e6442e2b01:/usr/share/nginx/html# cat index.html                                             ]
<h1>Hello, this is Docker Volume Setup by Raghav </h1>
root@e2e6442e2b01:/usr/share/nginx/html#
```



Hello, This is Docker Volume Setup by Rajat

```
Raghav ~ % docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS
        PORTS                  NAMES
e2e6442e2b01   nginx          "/docker-entrypoint.…"   3 minutes ago    Up 3 min
utes    0.0.0.0:8008->80/tcp   nginxRaghav
4b663aeb9c71   redis:latest   "docker-entrypoint.s…"   7 minutes ago    Up 7 min
utes    6379/tcp               redisrajatkaboj
56824fdef902   redis          "docker-entrypoint.s…"   15 minutes ago   Up 15 mi
nutes   6379/tcp               ecstatic_mclean
Raghav ~ %
```

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

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

```
Last login: Wed Sep 25 12:02:24 on ttys000
(base) rajatkamboj@rajats-Air ~ % docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                  NAMES
f6e9bc9cff58   nginx     "/docker-entrypoint.…"   16 minutes ago   Up 16 minutes   0.0.0.0:8001->80/tcp   nginxRajat
(base) rajatkamboj@rajats-Air ~ % docker stop nginxRajat
nginxRajat
(base) rajatkamboj@rajats-Air ~ % docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
(base) rajatkamboj@rajats-Air ~ %
```

docker rm my_nginx

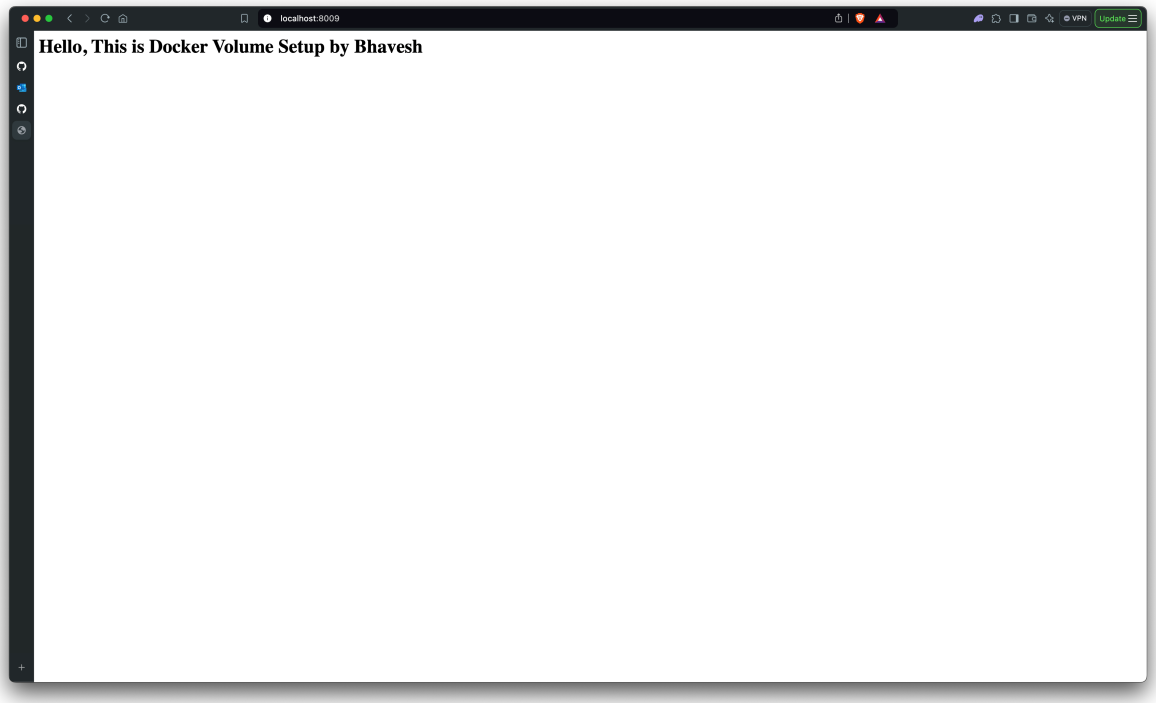
```
Last login: Wed Sep 25 12:02:24 on ttys000
(base) rajatkamboj@rajats-Air ~ % docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                  NAMES
f6e9bc9cff58   nginx     "/docker-entrypoint.…"   16 minutes ago   Up 16 minutes   0.0.0.0:8001->80/tcp   nginxRajat
(base) rajatkamboj@rajats-Air ~ % docker stop nginxRajat
nginxRajat
(base) rajatkamboj@rajats-Air ~ % docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
(base) rajatkamboj@rajats-Air ~ %
```

Run a new Nginx container using the same volume:

```
(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# _
```

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

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:

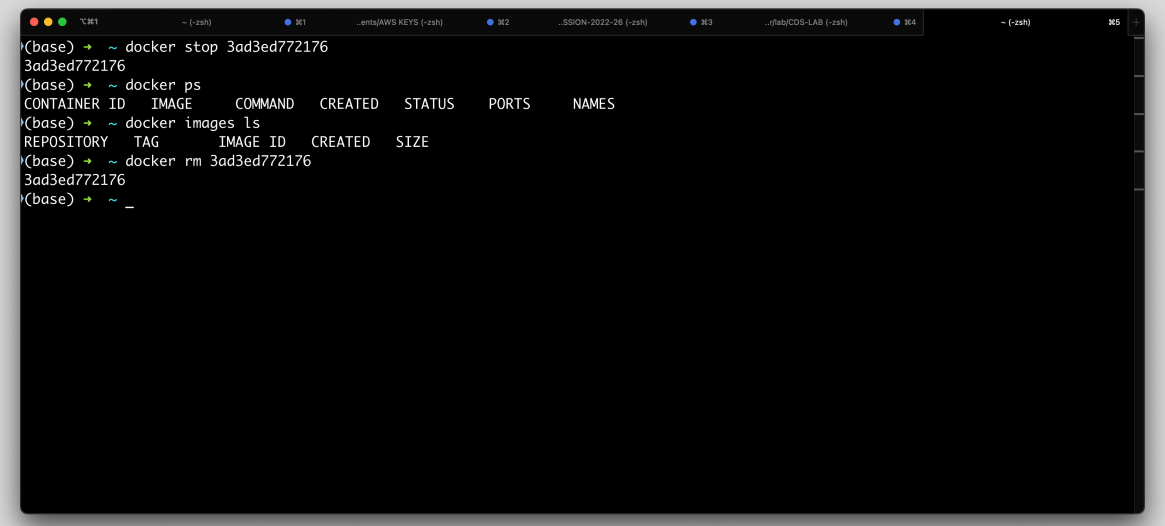
```
docker stop new_nginx docker rm
new_nginx
```

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.