

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.

```
PS C:\Users\Asus> docker volume create my_data_volume
my_data_volume
PS C:\Users\Asus> |
```

Verify that the volume was created:

```
docker volume ls
```

You should see my_data_volume listed among the volumes.

```
PS C:\Users\Asus> docker volume ls
DRIVER      VOLUME NAME
local      my_data_volume
PS C:\Users\Asus> |
```

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
```

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.

```
PS C:\Users\Asus> docker run -d --name my_nginx -v my_data_volume:/usr/share/nginx/html -p 8008:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
a480a496ba95: Pull complete
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
bbe987067f693ebfc3dc60ccae97bcc99d0580a8f2e5c1678be02ce6b937973b
```

Verify that the container is running:

```
docker ps
```

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

```
PS C:\Users\Asus> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
bbe987067f69   nginx    "/docker-entrypoint.    About a minute ago   Up About a minute   0.0.0.0:8008->80/tcp     my_nginx
PS C:\Users\Asus>
```

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

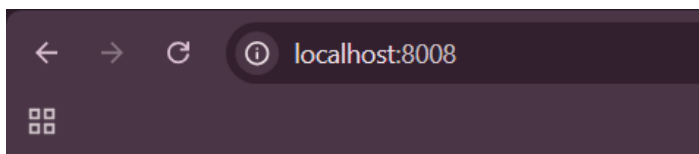
exit
```

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

```
bbe987067f69 /usr/share/nginx/html/index.html About a minute ago Up About a minute 0.0.0.0:8008 -> 0.0.0.0:8008 /usr/sbin/nginx
PS C:\Users\Asus> docker exec -it my_nginx bash
root@bbe987067f69:/# echo "<h1>Hello, Docker Volume</h1>" > /usr/share/nginx/html/index.html
root@bbe987067f69:/# exit
exit

What's next:
  Try Docker Debug for seamless, persistent debugging tools in any container or image → docker debug my\_nginx
  Learn more at https://docs.docker.com/go/debug-cli/
PS C:\Users\Asus> |
```

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.



Hello, Docker Volume

Step 4: Test Data Persistence

Stop and remove the container:

```
docker stop my_nginx
```

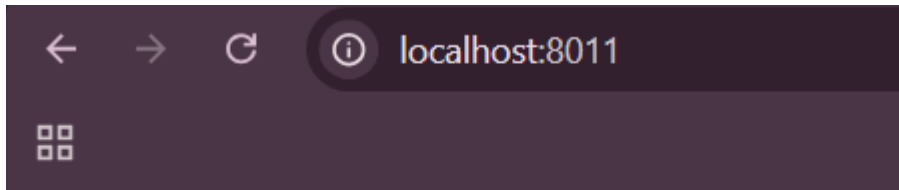
```
docker rm my_nginx
```

```
PS C:\Users\Asus> docker stop my_nginx
my_nginx
PS C:\Users\Asus> docker rm my_nginx
my_nginx
```

Run a new Nginx container using the same volume:

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

Access the Nginx server again: Navigate to <http://localhost:8011> in your browser. You should still see the "Hello, Docker Volume!" message, demonstrating that the data persisted across container instances.



Hello, Docker Volume

Step 5: Clean Up

Stop and remove the container:

```
docker stop new_nginx
```

```
docker rm new_nginx
```

```
ERROR: response from daemon: no such container: new_nginx
PS C:\Users\Asus> docker ps
CONTAINER ID   IMAGE     COMMAND
3cad563c0b36   nginx    "/docker-entrypoint..."
PS C:\Users\Asus> docker stop 3cad563c0b36
3cad563c0b36
PS C:\Users\Asus> docker rm 3cad563c0b36
3cad563c0b36
```

Remove the Docker volume:

```
docker volume rm my_data_volume
```

```
PS C:\Users\Asus> docker volume rm my_data_volume  
my_data_volume  
PS C:\Users\Asus> |
```

Verify that the volume is removed:

```
docker volume ls
```

Ensure that my_data_volume is no longer listed.

```
my_data_volume  
PS C:\Users\Asus> docker volume ls  
DRIVER      VOLUME NAME  
PS C:\Users\Asus> |
```