

Managing Docker Volumes - Exercise

Goal:

The goal of this exercise is to create and use volumes with containers.

Instructions:

Create a Volume

Create a volume named "localvolume".

```
docker volume create localvolume
```

Use the "docker volume inspect" command to determine where Docker will store the contents of the volume on your local Docker host's filesystem.

```
docker volume inspect localvolume
```

Note the "Mountpoint" path shown as you'll need it for the next step.

Create a text file named "file.txt" in the volume with the contents of "This file exists".

```
echo "This file exists" > /var/lib/docker/volumes/localvolume/_data/file.txt
```

NOTE: Use the output from the above inspect command to determine the path. It will probably be the same, but there is a slight chance that it may be different on your system.

Run a Container and Attach the Volume to It

Start a detached container using the Apache image named "httpd". Mount the volume named "localvolume" inside the container at "/data".

```
docker run -d --name mountvolume --mount src=localvolume,dst=/data httpd
```

NOTE: The abbreviation of source ("src") and destination ("dst") was used. You could also have used this for the "--mount" option: "source=localvolume,destination=/data"

Attach to the "mountvolume" container.

```
docker exec -it mountvolume /bin/bash
```

Check that the volume is available inside of the container and that the "file.txt" exists.

```
df -h  
cat /data/file.txt
```

Ensure that the contents of "/data/file.txt" is "This file exists".

Create a new file on the volume named "from-container.txt" with the contents of "Created from inside the container".

```
echo "Created from inside the container" > /data/from-container.txt
```

Confirm the file was created and that it's contents are as expected:

```
cat /data/from-container.txt
```

Detach from the container.

```
exit
```

Confirm that you can see the file from the Docker host.

```
cat /var/lib/docker/volumes/localvolume/_data/from-container.txt
```

Run a Container with an Ephemeral Volume

Start a container named "tempvolume" in the background. Use the "httpd" image. Also, attach an ephemeral volume to the container at "/tempdata". You'll need to use type "tmpfs" as the volume type.

```
docker run -d --name tempvolume --mount type=tmpfs,dst=/tempdata httpd
```

Check that the temporary volume is created correctly by inspecting the running container.

```
docker inspect tempvolume | grep Mounts -A 10
```

Ensure that the "Type" field shows "tmpfs" and that the "Destination" field shows "/tempdata" as the path.