**TRAINING MATERIALS - MODULE HANDOUT**

**Contacts**

**robert.crutchley@qa.com**

*team.qac.all.trainers@qa.com*

*www.consulting.qa.com*

**Contents**

[**Overview**](#_tj6lbuhsvl22) **1**

[**Creating and Managing Volumes**](#_8lmtub4f0y8o) **1**

[Create a Volume](#_357z0wtbok7i) 1

[List the Existing Volumes](#_5ef1pliitlk6) 2

[Remove a Volume](#_bpgfn22a96i) 2

[View more information about a Volume](#_yi2ju3tpu799) 2

[**Mounting Volumes**](#_8929xl6ldwvo) **2**

[Using the --volume Flag](#_jm67yzu9y133) 2

[Using the --mount Flag](#_geo1dkdskp5t) 2

[**Volume Drivers**](#_11jaczixfml4) **3**

[**Tasks**](#_6oyopzae8viw) **3**

[Create a New Volume](#_mx6fbyriz4nc) 3

[Create an NGINX Container](#_aagorb741itk) 3

[Create a Webpage](#_nwojyxf7w2nf) 3

[Recreate the NGINX Container](#_pv4ltnurx6vj) 4

[Start Another NGINX Container](#_e98q988mzhfb) 4

[Make a Change to the Webpage](#_fqg654q286sj) 4

[Clean Up](#_irswiuegx3x2) 4

# Overview

Volumes are another way in Docker to persist data from the container. Unlike Bind Mounts, Volumes are managed by Docker. Bind Mounts rely on a the host’s directory structure. Volumes are a newer addition to Docker compared Bind Mounts, because of this they have more functionality and benefits. The main benefit about using volumes is that they are just easier to manage. Bind Mounts are great for just plugging one file into one container but when you want to start sharing files or directories across multiple containers then volumes becomes a much more manageable solution.

# Creating and Managing Volumes

Before even creating a container you can create and manage volumes in Docker.

## Create a Volume

Just provide the name to create a volume in Docker.

|  |
| --- |
| docker volume create **[VOLUME\_NAME]** |
| docker volume create my-volume |

## List the Existing Volumes

We can view a list of all the volumes available on the Docker host.

|  |
| --- |
| docker volume ls |

## Remove a Volume

Provide the name of the volumes to delete it.

|  |
| --- |
| docker volume rm **[VOLUME\_NAME]** |
| docker volume rm my-volume |

## View more information about a Volume

The inspect command can be used to see more details about a volume in Docker.

|  |
| --- |
| docker volume inspect **[VOLUME\_NAME]** |
| docker volume inspect my-volume |

# Mounting Volumes

Just like with Bind Mounts, you can choose to use either the volume or mount flags. Mounting the volumes into containers is very similar to creating Bind Mounts, except for the source we are pointing to the volume that we want to be mounted not a location on the host file system. Options for mounting volumes as read only are available, just like with Bind Mounts.

When specifying a volume name with any of these commands, if the volumes doesn’t exist then Docker will create one for you. This is handy feature as it allows us to skip a step if need be, but make sure to spell your volume name correctly because Docker won’t care if you don’t.

## Using the --volume Flag

When using the volume flag we need provide the name of the volume that is going to be mounted and the location which the volume is going to be mounted in on the container.

|  |
| --- |
| docker run --volume **[VOLUME\_NAME]:[LOCATION\_ON\_CONTAINER]** |
| docker run --volume my-volume:/usr/share/nginx/html |

## Using the --mount Flag

With the mount flag provide the volume name and the destination on the container to mount the volume.

|  |
| --- |
| docker run --mount source=**[VOLUME\_NAME],**destination=**[LOCATION\_ON\_CONTAINER]** |
| docker run --mount source=my-volume,destination=/usr/share/nginx/html |

# Volume Drivers

Whilst we won't be getting hands on with volume drivers here, they are definitely worth knowing about for future reference. When you do a listing for all the existing volumes, you might have noticed a driver column called local. This basically means that the volume is stored on the host machine. You may at some point want to develop a solution where the volume can be stored on a remote host or in a cloud storage solution perhaps. Plugins will allow you run different drivers to attach volumes to remote places like NFS servers or cloud storage.