Full Stack Development
Lab Docker 1

Working with Images

1. Lab objectives

This initial lab is designed to help you understand how Docker images work and how to manage your own images and registry

2. Docker installation notes

Docker is a Linux application, so it installs easily on all Linux distributions and Mac. However, to run Docker in Windows, the installation a Linux VM called the *Windows Linux Subsystem,* or WSL. This setup can be complex because it depends on the Windows build you are using and your BIOS virtualization settings. How to set up Docker on Windows is beyond the scope of this lab.

All commands in this and other Docker labs work in Linux, Mac or Windows Docker installation without any modification

To check if you have Docker installed, open a Bash shell or command window or PowerShell and check “docker ps” to be sure the Docker daemon is running with the right privileges

```
D:\Docker>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

D:\Docker>
```

3. Pulling a Docker image

Check that there are no images in your local registry

```
D:\Docker>docker image ls
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE

D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
```

Pull the latest version of the minimal Linux image “Alpine” then check to see if it in the local registry

```
D:\Docker>docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
ca7dd9ec2225: Pull complete
Digest: sha256:b95359c2505145f16c6aa384f9cc74eeff78eb36d308ca4fd902eeeb0a0b161b
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest

D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED      SIZE
alpine       latest    bfe296a52501   4 days ago   5.54MB
```

4. Inspect the image

Using the docker inspect command, do an info dump about the image. Note that the dump is longer than can be show in this lab manual so take some time to go through it and see what sort of information is contained in the image

```
D:\Docker>docker inspect alpine
[
    {
        "Id": "sha256:bfe296a525011f7eb76075d688c681ca4feaad5afe3b142b36e30f1a171dc99a",
        "RepoTags": [
            "alpine:latest"
        ],
        "RepoDigests": [
            "alpine@sha256:b95359c2505145f16c6aa384f9cc74eeff78eb36d308ca4fd902eeeb0a0b161b"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2022-11-12T04:19:23.199716539Z",

… and a whole lot more output..
```

5. Pull two different versions of Ubuntu

Pull the 18.04 Ubuntu and the 20.04 Image of Ubuntu

```
D:\Docker>docker pull ubuntu:18.04
18.04: Pulling from library/ubuntu
a404e5416296: Pull complete
Digest: sha256:ca70a834041dd1bf16cc38dfcd24f0888ec4fa431e09f3344f354cf8d1724499
Status: Downloaded newer image for ubuntu:18.04
docker.io/library/ubuntu:18.04

D:\Docker>docker pull ubuntu:20.04
20.04: Pulling from library/ubuntu
eaead16dc43b: Pull complete
Digest: sha256:450e066588f42ebe1551f3b1a535034b6aa46cd936fe7f2c6b0d72997ec61dbd
Status: Downloaded newer image for ubuntu:20.04
docker.io/library/ubuntu:20.04
```

Check to see that the images are in your local registry

```
D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
alpine       latest    bfe296a52501   4 days ago    5.54MB
ubuntu       20.04     680e5dfb52c7   3 weeks ago   72.8MB
ubuntu       18.04     71eaf13299f4   3 weeks ago   63.1MB
```

Add a second tag to the alpine image. Note that the IMAGE ID might be different if the latest image of alpine has been updates since this manual was prepared. Make sure you use the IMAGE ID on listing of images. Since the tag represents a repository, you should include a version after colon, but if you don't, it will default to "latest."

```
D:\Docker>docker tag bfe296a52501 zippy:1.0

D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
alpine       latest    bfe296a52501   4 days ago    5.54MB
zippy        1.0       bfe296a52501   4 days ago    5.54MB
ubuntu       20.04     680e5dfb52c7   3 weeks ago   72.8MB
ubuntu       18.04     71eaf13299f4   3 weeks ago   63.1MB
```

Notice that both **alpine:latest** and **zippy:1.0** refer to the same image id.

6. Deleting images

Delete the 18.04 Ubuntu image using the image tag

```
D:\Docker>docker image rm ubuntu:18.04
Untagged: ubuntu:18.04
Untagged:ubuntu@sha256:ca70a834041dd1bf16cc38dfcd24f0888ec4fa431e09f3344f354c
       f8d1724499
Deleted: sha256:71eaf13299f415122ad887b4592146a6b6f5e80cd69e0cd4650102fa3a99972c
Deleted: sha256:69f57fbceb1b420d7e4697e0f6514887b0805ee0059bea7d51e0a832962e74bf

D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
zippy        1.0       bfe296a52501   4 days ago    5.54MB
alpine       latest    bfe296a52501   4 days ago    5.54MB
ubuntu       20.04     680e5dfb52c7   3 weeks ago   72.8MB
```

Note that the tag is removed and then the image is deleted

Now do the same for the other Ubuntu image but this time using the image id.

```
D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
alpine       latest    bfe296a52501   4 days ago    5.54MB
zippy        1.0       bfe296a52501   4 days ago    5.54MB
ubuntu       20.04     680e5dfb52c7   3 weeks ago   72.8MB

D:\Docker>docker rmi 680e5dfb52c7
Untagged: ubuntu:20.04
Untagged:ubuntu@sha256:450e066588f42ebe1551f3b1a535034b6aa46cd936fe7f2c6b0d7
       2997ec61dbd
Deleted: sha256:680e5dfb52c74a1fbc99c2922c8e25b5736e6cd1a3d9430890d52a4f8f44087a
Deleted: sha256:f4462d5b2da2985f37409c9b257afd2b9fb82356ce4e43e804ee34214242e34a
```

Try removing the alpine image using the image id. Notice that you can't because there are two different tags for the image. You can only remove an image if it has one or zero tags. The two repositories referenced are **alpine** and **zippy**

```
D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
alpine       latest    bfe296a52501   4 days ago    5.54MB
zippy        1.0       bfe296a52501   4 days ago    5.54MB

D:\Docker>docker rmi bfe296a52501
Error response from daemon: conflict: unable to delete bfe296a52501 (must be forced) - image is
referenced in multiple repositories
```

First you must remove one of the tags, notice that the **docker image rmi** command in this case only erases a tag. It won't delete the image unless there is at most one tag associated with it.

```
D:\Docker>docker rmi alpine
Untagged: alpine:latest
Untagged:alpine@sha256:b95359c2505145f16c6aa384f9cc74eeff78eb36d308ca
4fd902eeeb0a0b161b

D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED      SIZE
zippy        1.0       bfe296a52501   4 days ago   5.54MB
```

Now you can delete the last image using the image ID.

```
D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID       CREATED      SIZE
zippy        1.0       bfe296a52501   4 days ago   5.54MB

D:\Docker>docker rmi bfe296a52501
Untagged: zippy:1.0
Deleted:sha256:bfe296a525011f7eb76075d688c681ca4feaad5afe3b142b36e30f1a1
         71dc99a
Deleted: sha256:e5e13b0c77cbb769548077189c3da2f0a764ceca06af49d8d558e759f5c232bd

D:\Docker>docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE

D:\Docker>
```

End of Lab