Full Stack Development
Lab Docker 2

Running Containers

1. Lab objectives

This lab covers basic container management in Docker.

2. Setup

You should start this lab with no images or containers installed. If you have any containers running, you should stop them, then run **docker container prune** to remove all the stopped containers. Then ensure all the images you have locally are removed. You should know how to do that from the last lab

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

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

3. Docker "hello-world"

Docker has a hello-world image. Execute the command

docker run --name greetings hello-world

This will pull the image and run a container named "greetings." The image's default command just prints out a hello message to the terminal. The container exits once the command has been executed.

```
D:\Docker>docker run --name greetings hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:faa03e786c97f07ef34423fccceeec2398ec8a5759259f94d99078f264e9d7af
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/
```

Run the command **docker ps -a** or **docker container ls -a** to see information about the container than just ran

```
D:\Docker>docker ps -a
CONTAINER ID    IMAGE         COMMAND     CREATED         STATUS                   PORTS     NAMES
327fc4b8b5f9    hello-world   "/hello"    3 minutes ago   Exited (0) 3 minutes ago           greet-
ings

D:\Docker>docker container ls -a
CONTAINER ID    IMAGE         COMMAND     CREATED         STATUS                   PORTS     NAMES
327fc4b8b5f9    hello-world   "/hello"    3 minutes ago   Exited (0) 3 minutes ago           greet-
ings
```

If you try to remove the image, you can't because it is being used by a container

```
D:\Docker>docker ps -a
CONTAINER ID    IMAGE         COMMAND      CREATED        STATUS                      PORTS        NAMES
327fc4b8b5f9    hello-world   "/hello"     3 minutes ago  Exited (0) 3 minutes ago                 greet-
ings

D:\Docker>docker container ls -a
CONTAINER ID    IMAGE         COMMAND      CREATED        STATUS                      PORTS        NAMES
327fc4b8b5f9    hello-world   "/hello"     3 minutes ago  Exited (0) 3 minutes ago                 greet-
ings
```

Remove the stopped container with docker container rm greetings then delete the image

```
D:\Docker>docker container rm greetings
greetings

D:\Docker>docker container ls -a
CONTAINER ID    IMAGE       COMMAND     CREATED     STATUS      PORTS       NAMES

D:\Docker>docker rmi hello-world
Untagged: hello-world:latest
Untagged: hello-world@sha256:faa03e786c97f07ef34423fccceeec2398ec8a5759259f94d99078f264e9d7af
Deleted: sha256:feb5d9fea6a5e9606aa995e879d862b825965ba48de054caab5ef356dc6b3412
Deleted: sha256:e07ee1baac5fae6a26f30cabfe54a36d3402f96afda318fe0a96cec4ca393359
```

Confirm you have no images or containers. You should know how to do this at this point.

4. Running Containers

Pull the latest Ubuntu image from Docker Hub. You should know how to do this.

Run the image using **docker run ubuntu**

Run **docker ps -a** to see what happened. Notice that the default command was **bash** but there it exited immediately because there was no terminal session associated with it.

```
D:\Docker>docker run ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
e96e057aae67: Pull complete
Digest: sha256:4b1d0c4a2d2aaf63b37111f34eb9fa89fa1bf53dd6e4ca954d47caebca4005c2
Status: Downloaded newer image for ubuntu:latest

D:\Docker>docker ps -a
CONTAINER ID    IMAGE       COMMAND     CREATED         STATUS                      PORTS       NAMES
412aa267e69d    ubuntu      "bash"      7 seconds ago   Exited (0) 6 seconds ago                epic_gold-
berg
```

Repeat the last step and notice that you now have two exited containers. Notice that each has a name generated by Docker since you didn't provide one when you started the container

```
D:\Docker>docker run ubuntu

D:\Docker>docker ps -a
CONTAINER ID    IMAGE     COMMAND     CREATED            STATUS                      PORTS      NAMES
3408c0e88c46      ubuntu        "bash"        15  seconds  ago      Exited  (0)  14  seconds  ago
reverent_wozniak
412aa267e69d      ubuntu        "bash"        2  minutes  ago        Exited  (0)  2  minutes  ago
epic_goldberg
```

Remove all the stopped containers with the **docker container prune** command

```
D:\Docker>docker container prune
WARNING! This will remove all stopped containers.
Are you sure you want to continue? [y/N] y
Deleted Containers:
3408c0e88c465e9cea7debb15f0b023dfe741e52abf719ba59a4a5361002d7a4
412aa267e69dee1d73a8665b63e36b094dead4fe7fe26c48f474eba8baf21da1

Total reclaimed space: 0B

D:\Docker> docker container ls -a
CONTAINER ID    IMAGE      COMMAND    CREATED    STATUS      PORTS       NAMES
```

5. Running containers interactively

Now run an Ubuntu image interactively with **docker run -it ubuntu** You don't need to specify the command to run since **bash** is the default command the container runs when it starts. This should open a shell in the running container. Execute several bash commands to confirm you are in the containers

Once you are finished, exit the shell with **exit** or **control-D**. As soon as the shell exits, the container stops. Confirm that it is stopped.

```
D:\Docker>docker run -it --name shelly ubuntu

root@3369c57141df:/# cat /etc/issue
Ubuntu 22.04.1 LTS \n \l

root@3369c57141df:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media
mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var

root@3369c57141df:/# exit
exit

D:\Docker>docker ps -a
CONTAINER ID    IMAGE     COMMAND     CREATED        STATUS                    PORTS      NAMES
3369c57141df    ubuntu    "bash"      30 seconds ago  Exited (0) 6 seconds ago              shelly
```

Remove all the containers and images that you used in this lab.

End Lab