Deploying a Web Application On Docker

Objective:
This task involves creating a Docker container to host a simple HTML website, making it accessible on port 18085 of the host machine, and persisting data to a mounted volume.

Scenario:
Your company is transitioning to containerized deployments. This task provides hands-on experience with Docker, including image creation, containerization, and volume mounting.

Constraints:
- You can use httpd image provided by docker.
- You can take a precreated templates form the internet or you can build your own website if you are good with httpd.

1. Check docker version or install docker on your machine

```
root@ubuntu:~# docker --version
Docker version 27.3.1, build ce12230
```

2. Create a directory go into that directory using simple CD command and create a file name index.html

```
root@ubuntu:~# mkdir ~/html-website
root@ubuntu:~# cd ~/html-website
root@ubuntu:~/html-website# vi index.html
```

3. Build your own html

```
<!DOCTYPE html>
<html lang="en">
        <head>
                <meta charset="UTF-8">
                    <meta name="viewport" content="width=device-width, initial-scale=1.0">
                        <title>My Dockerized WebsiteWebsite</h1>
                            <p>This website is served from a Docker container running Apache HTTP Server.</p>
                            </body>
</html>
<!DOCTYPE html>
<html lh, initial-scale=1.0">
            <title>My Dockerized Website</title>
        </head>
        <body>
                <h1>Welcome to My Website</h1>
                    <p>This website is served from a Docker container running Apache HTTP Server.</p>
        </body>
</html>

~
~
~
~
~
```

4. Create a Volume for Persistent Storage ● Use Docker to create a volume to store website data persistently
   - Pull the httpd Docker Image ● Pull the latest httpd image from Docker Hub

```
root@ubuntu:~/html-website# docker run -dit --name my-website \
>   -p 18085:80 \
>   -v ~/html-website:/usr/local/apache2/htdocs \
>   httpd:latest
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
2d429b9e73a6: Pull complete
d675ed392a91: Pull complete
4f4fb700ef54: Pull complete
3ed0d9182dde: Pull complete
0062038102c9: Pull complete
334a67c7f78b: Pull complete
Digest: sha256:6bdbdf5ac16ac3d6ef543a693fd5dfafae2428b4b0cdc52a480166603a069136
Status: Downloaded newer image for httpd:latest
323fbc6304523988e9f9939cba51df3aec02cc65d666e59447fae1e74a66462d
```

5. Run the Container with Volume Mount ● Run the container, mapping the volume and exposing the desired port

```
root@ubuntu:~/html-website# docker ps
CONTAINER ID   IMAGE          COMMAND              CREATED          STATUS          PORTS                                      NAMES
323fbc630452   httpd:latest   "httpd-foreground"   52 seconds ago   Up 33 seconds   0.0.0.0:18085->80/tcp, [::]:18085->80/tcp   my-website
```

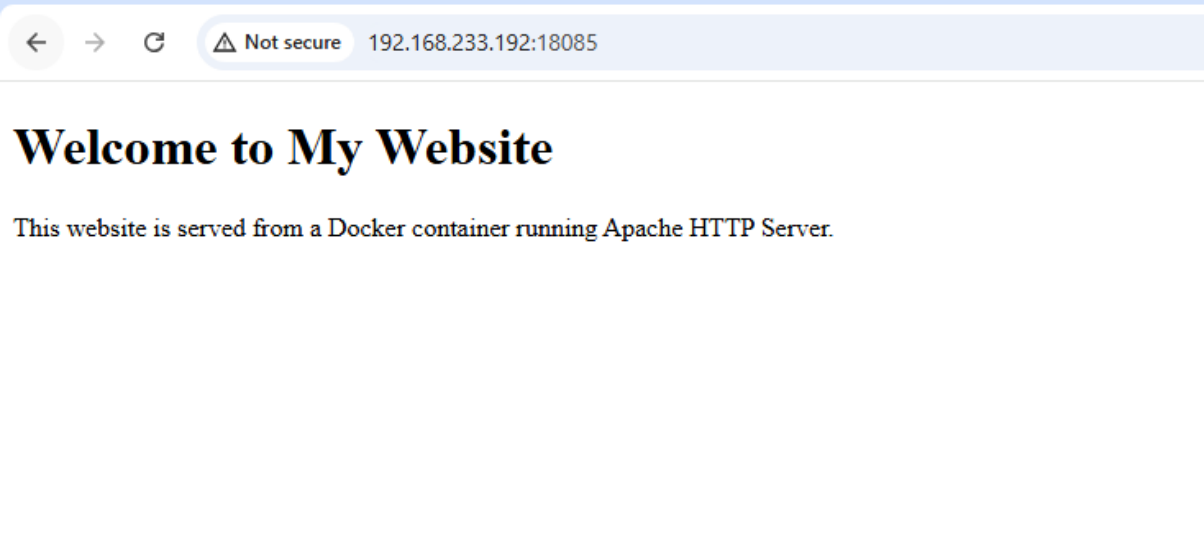
```
root@ubuntu:~/html-website# docker images
REPOSITORY   TAG       IMAGE ID       CREATED        SIZE
httpd        latest    dad6ca1caf78   4 months ago   148MB
```

6. Verify the Setup ● Open a web browser and navigate to: “http://:18085”

```
root@ubuntu:~/html-website# root@ubuntu:~/html-website# curl localhost:18085
<!DOCTYPE html>
<html lang="en">
        <head>
                    <meta charset="UTF-8">
                        <meta name="viewport" content="width=device-width, initial-scale=1.0">
                            <title>My Dockerized WebsiteWebsite</h1>
                                <p>This website is served from a Docker container running Apache HTTP Server.</p>
                                </body>
</html>
<!DOCTYPE html>
<html lh, initial-scale=1.0">
            <title>My Dockerized Website</title>
        </head>
        <body>
                    <h1>Welcome to My Website</h1>
                        <p>This website is served from a Docker container running Apache HTTP Server.</p>
        </body>
</html>

root@ubuntu:~/html-website#
```

Also open your web browser



- Prasad Rajendra Pansare