Lab Exercise 5- Building a Docker Image for an HTML App Using Nginx

1. Setup

You will need:

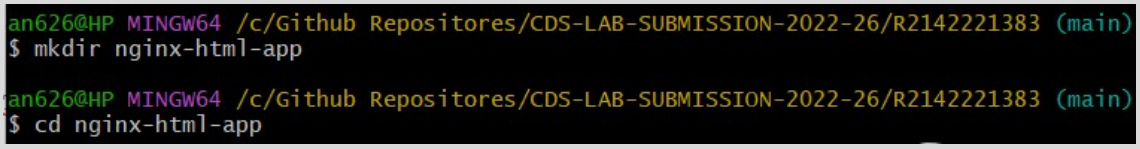
- Docker installed on your machine.
- A simple HTML file for the app.

2. Step 1: Create the HTML File

Create a directory for your HTML app and place an index.html file in it.

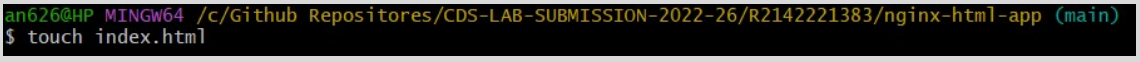
```
mkdir nginx-html-app

cd nginx-html-app
```



Inside the nginx-html-app directory, create the HTML file.

```
touch index.html
```



Edit the index.html file with the following content (or any custom HTML content you want):

```
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to My Nginx HTML App</title>
</head>
<body>
  <h1>Hello, Nginx Docker!</h1>
  <p>This is a simple HTML app served by Nginx in a Docker container.</p>
</body>
```

</html>

MINGW64:/c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app

```
<!DOCTYPE html>
<html>
<head>
    <title>Welcome to My Nginx HTML App</title>
</head>
<body>
    <h1>Hello, Nginx Docker!</h1>
    <p>This is a simple HTML app served by Nginx in a Docker container.</p>
</body>
</html>
```

3. Step 2: Create a Dockerfile

In the same directory, create a Dockerfile. This file will define how to build the Docker image using Nginx as the base image.

touch Dockerfile

```
an626@HP MINGW64 /c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app (main)
$ touch Dockerfile
```

Edit the Dockerfile and add the following content:

FROM nginx:latest

COPY index.html /usr/share/nginx/html/

EXPOSE 80

Dockerfile > FROM

```
1   FROM nginx:latest
2   COPY index.html /usr/share/nginx/html/
3   EXPOSE 80
```

4. Step 3: Build the Docker Image

Now that you have the Dockerfile and index.html, it's time to build the Docker image. Run the following command to build the image, giving it a tag (e.g., nginx-html-app):

docker build -t nginx-html-app .

```
an626@HP MINGW64 /c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app (main)
$ docker build -t nginx-html-app .
[+] Building 0.4s (7/7) FINISHED                                                                    docker:default
 => [internal] load build definition from Dockerfile                                                         0.0s
 => => transferring dockerfile: 105B                                                                         0.0s
 => [internal] load .dockerignore                                                                            0.1s
 => => transferring context: 2B                                                                              0.0s
 => [internal] load metadata for docker.io/library/nginx:latest                                              0.0s
 => [internal] load build context                                                                            0.1s
 => => transferring context: 259B                                                                            0.0s
 => [1/2] FROM docker.io/library/nginx:latest                                                                0.1s
 => [2/2] COPY index.html /usr/share/nginx/html/                                                             0.1s
 => exporting to image                                                                                       0.0s
 => => exporting layers                                                                                      0.0s
 => => writing image sha256:2b18e83e8b0933c28128d78d2279c5bea0ea88aae39d5272ece3b68099dfc800                 0.0s
 => => naming to docker.io/library/nginx-html-app                                                            0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
```

Docker will use the Nginx base image, copy your index.html into the appropriate directory, and build the image.

5. Step 4: Run the Docker Container

After building the image, you can run the container with the following command:

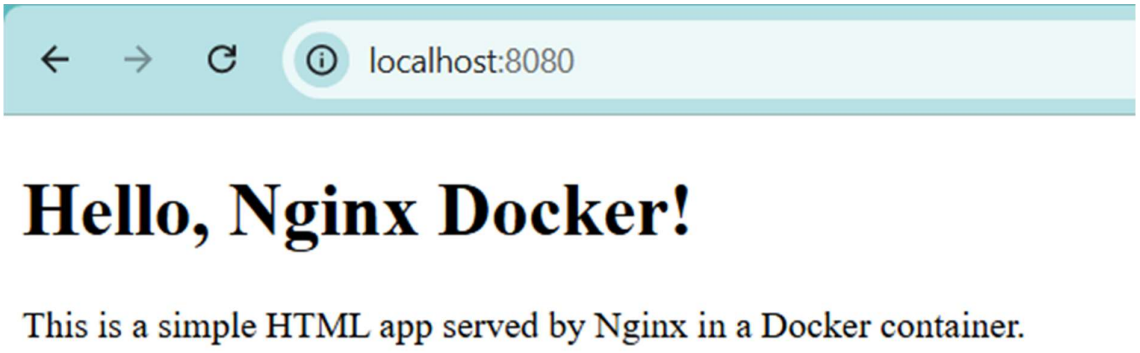
docker run -d -p 8080:80 nginx-html-app

```
an626@HP MINGW64 /c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app (main)
$ docker run -d -p 8080:80 nginx-html-app
e3fd1f3616a3fc7739684a21649410c6a1793bd6b69ef995a513d4286c5631c1
```

This command runs the container in detached mode (-d) and maps port 8080 on your host machine to port 80 inside the container, where Nginx is serving your HTML app.

6. Step 5: Verify

Open a browser and go to http://localhost:8080. You should see your HTML page with the message “Hello, Nginx Docker!”.



7. Step 6: Stop and Remove the Container

Once you're done, you can stop and remove the container:

docker ps # to see running containers

```
an626@HP MINGW64 /c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app (main)
$ docker ps
CONTAINER ID   IMAGE            COMMAND                  CREATED         STATUS         PORTS                   NAMES
e3fd1f3616a3   nginx-html-app   "/docker-entrypoint.…"   2 minutes ago   Up 2 minutes   0.0.0.0:8080->80/tcp   friendly_tesla
```

docker stop <container-id>

```
an626@HP MINGW64 /c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app (main)
$ docker stop e3fd1f3616a3fc7739684a21649410c6a1793bd6b69ef995a513d4286c5631c1
e3fd1f3616a3fc7739684a21649410c6a1793bd6b69ef995a513d4286c5631c1
```

docker rm <container-id>

```
an626@HP MINGW64 /c/Github Repositores/CDS-LAB-SUBMISSION-2022-26/R2142221383/nginx-html-app (main)
$ docker rm e3fd1f3616a3
e3fd1f3616a3
```