Lab Exercise 4- Building a Docker Image for an HTML App Using Nginx

Name Viraj bhidola

Sap 500121825

B2 Devops

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

```
PS C:\Users\ASUS> mkdir nginx-html-app


    Directory: C:\Users\ASUS


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         2/2/2026   7:02 PM                nginx-html-app


PS C:\Users\ASUS> cd nginx-html-app
PS C:\Users\ASUS\nginx-html-app> |
```

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

touch index.html

```
PS C:\Users\ASUS\nginx-html-app> touch index.html
PS C:\Users\ASUS\nginx-html-app> |
```

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>
```
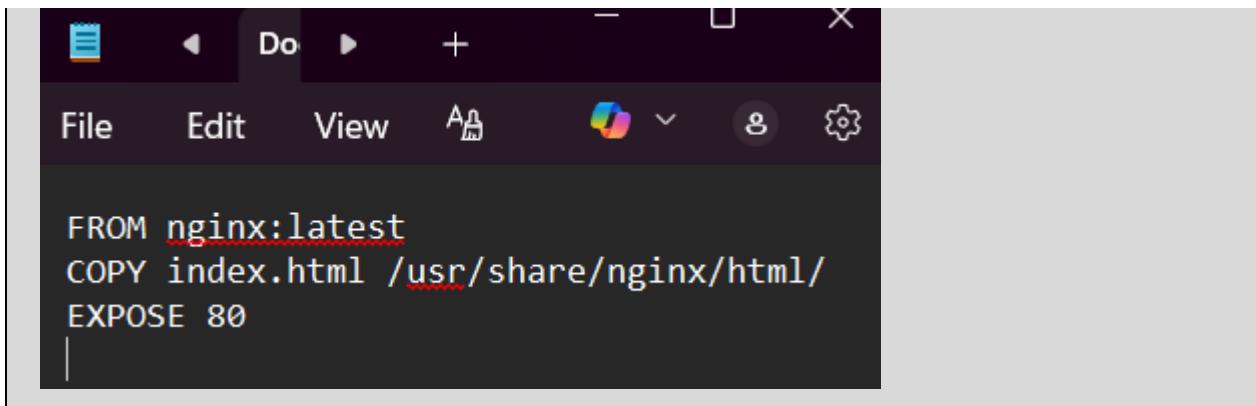
3. Step 2: Create a Dockerfile

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
```

Edit the Dockerfile and add the following content:

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

```
FROM nginx:latest
COPY index.html /usr/share/nginx/html/
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 .
```



```
PS C:\Users\ASUS\nginx-html-app> docker build -t nginx-html-app .
[+] Building 0.6s (7/7) FINISHED                          docker:desktop-linux
 => [internal] load build definition from Dockerfile                      0.0s
 => => transferring dockerfile: 107B                                      0.0s
 => [internal] load metadata for docker.io/library/nginx:latest          0.1s
 => [internal] load .dockerignore                                        0.0s
 => => transferring context: 2B                                          0.0s
 => [internal] load build context                                        0.0s
 => => transferring context: 31B                                         0.0s
 => [1/2] FROM docker.io/library/nginx:latest@sha256:c881927c4077710a    0.1s
 => => resolve docker.io/library/nginx:latest@sha256:c881927c4077710a    0.0s
 => [2/2] COPY index.html /usr/share/nginx/html/                         0.0s
 => exporting to image                                                    0.2s
 => => exporting layers                                                   0.1s
 => => exporting manifest sha256:a9e3ef1bea9a66ece2c695d150605af0eded    0.0s
 => => exporting config sha256:877d6987a1cf975682bb1bd0181348cabb4b43    0.0s
 => => exporting attestation manifest sha256:b72885d65f21246dc295a1a7    0.0s
 => => exporting manifest list sha256:f47870f63bd78817073ebc67448b591    0.0s
 => => naming to docker.io/library/nginx-html-app:latest                 0.0s
 => => unpacking to docker.io/library/nginx-html-app:latest              0.0s
PS C:\Users\ASUS\nginx-html-app>
```

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 8006:80 nginx-html-app
```

```
S C:\Users\ASUS\nginx-html-app> docker run -d -p 8006:80 nginx-html-app
cbcef3f90039345d4ca36c6b0d800fd1732c83744b0806fb4706d3db0fcbce2
S C:\Users\ASUS\nginx-html-app>
```

This command runs the container in detached mode (-d) and maps port 8006 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:8006. 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

docker stop <container-id>

docker rm <container-id>
```

```
PS C:\Users\ASUS\nginx-html-app> docker ps  # to see running containers
CONTAINER ID   IMAGE           COMMAND                  CREATED         STATUS         PORTS
                                NAMES
fcbcef3f9003   nginx-html-app  "/docker-entrypoint..."  19 seconds ago  Up 18 seconds  0.0.0.0:8006
->80/tcp, [::]:8006->80/tcp   angry_booth
PS C:\Users\ASUS\nginx-html-app> docker stop fcbcef3f9003
fcbcef3f9003
PS C:\Users\ASUS\nginx-html-app> docker rm fcbcef3f9003
fcbcef3f9003
PS C:\Users\ASUS\nginx-html-app>
```