

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.

```
C:\Users\OM VATS>mkdir nginx-html-app  
C:\Users\OM VATS>cd nginx-html-app
```

```
C:\Users\OM VATS\nnginx-html-app>echo. > index.html  
C:\Users\OM VATS\nnginx-html-app>notepad index.html
```

```
<!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.

```
C:\Users\OM VATS\nnginx-html-app>echo. > Dockerfile  
C:\Users\OM VATS\nnginx-html-app>notepad Dockerfile
```

```
File Edit View

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

```
C:\Users\OM VATS\nnginx-html-app>docker build -t nginx-html-app .
[+] Building 0.7s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 110B
=> [internal] load metadata for docker.io/library/nginx:latest
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 271B
=> [1/2] FROM docker.io/library/nginx:latest
=> [2/2] COPY index.html /usr/share/nginx/html/
=> exporting to image
=> => exporting layers
=> => writing image sha256:3641cc3706422474e6f0128ee3d66b1211d7da9f6e6b5aecdb48f75b8c21e2ab
=> => naming to docker.io/library/nginx-html-app

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/4y3h4gyner002slx73dmevxee

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview

C:\Users\OM VATS\nnginx-html-app>docker run -d -p 8080:80 nginx-html-app
74f374a5972572ef6fa01b33d3b717de449d53ddb07ac00b1646660d933eb088
```

5. Step 4: Run the Docker Container

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

```
C:\Users\OM VATS\nnginx-html-app>docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
74f374a59725   nginx-html-app "/docker-entrypoint..." About a minute ago Up 43 seconds   0.0.0.0:8080->80/tcp    nervous_shirley
```

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!”.



Hello, Nginx Docker!

This is a simple HTML app served by Nginx in a Docker container.

7. Step 6: Stop and Remove the Container

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

```
C:\Users\OM VATS\nginx-html-app>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                NAMES
74f374a59725   nginx-html-app  "/docker-entrypoint..." About a minute ago  Up 43 seconds  0.0.0.0:8080->80/tcp  nervous_shirley

C:\Users\OM VATS\nginx-html-app>docker stop 74f374a5972572ef6fa01b33d3b717de449d53ddb07ac00b1646660d933eb088
74f374a5972572ef6fa01b33d3b717de449d53ddb07ac00b1646660d933eb088

C:\Users\OM VATS\nginx-html-app>docker rm 74f374a5972572ef6fa01b33d3b717de449d53ddb07ac00b1646660d933eb088
74f374a5972572ef6fa01b33d3b717de449d53ddb07ac00b1646660d933eb088
```