

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

Name:- Vansh Bhatt

Sap ID: 500125395

Batch:- DevOps B1

To:- Hitesh Sharma Sir

1. Setup

You will need:

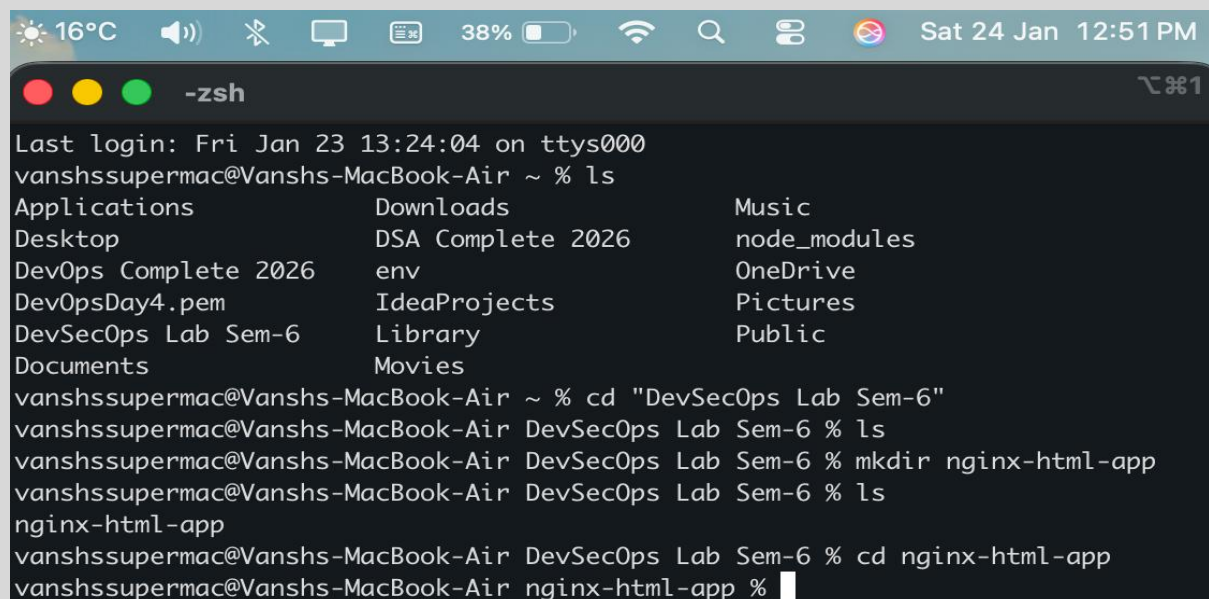
- Docker installed on your machine.
- A simple HTML file for the app.

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
```

A screenshot of a macOS terminal window. The title bar shows the system status with weather (16°C), volume, network, and battery (38%) icons, along with the date and time (Sat 24 Jan 12:51 PM). The terminal window has a dark background and shows the following commands and output:

```
-zsh
Last login: Fri Jan 23 13:24:04 on ttys000
vanshssupermac@Vanshs-MacBook-Air ~ % ls
Applications      Downloads         Music
Desktop           DSA Complete 2026 node_modules
DevOps Complete 2026  env              OneDrive
DevOpsDay4.pem   IdeaProjects     Pictures
DevSecOps Lab Sem-6  Library         Public
Documents         Movies

vanshssupermac@Vanshs-MacBook-Air ~ % cd "DevSecOps Lab Sem-6"
vanshssupermac@Vanshs-MacBook-Air DevSecOps Lab Sem-6 % ls
nginx-html-app

vanshssupermac@Vanshs-MacBook-Air DevSecOps Lab Sem-6 % mkdir nginx-html-app
vanshssupermac@Vanshs-MacBook-Air DevSecOps Lab Sem-6 % ls
nginx-html-app

vanshssupermac@Vanshs-MacBook-Air DevSecOps Lab Sem-6 % cd nginx-html-app
vanshssupermac@Vanshs-MacBook-Air nginx-html-app %
```

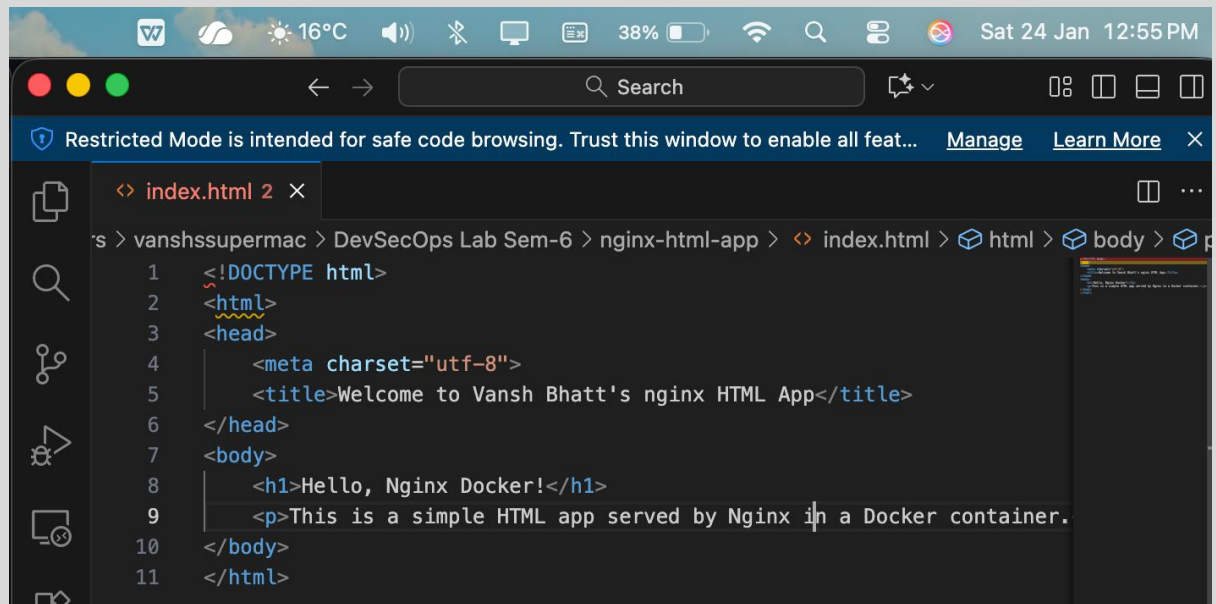
Inside the nginx-html-app directory, create the HTML file.

```
touch index.html
```

```
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % touch index.html
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % ls
index.html
vanshssupermac@Vanshs-MacBook-Air nginx-html-app %
```

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

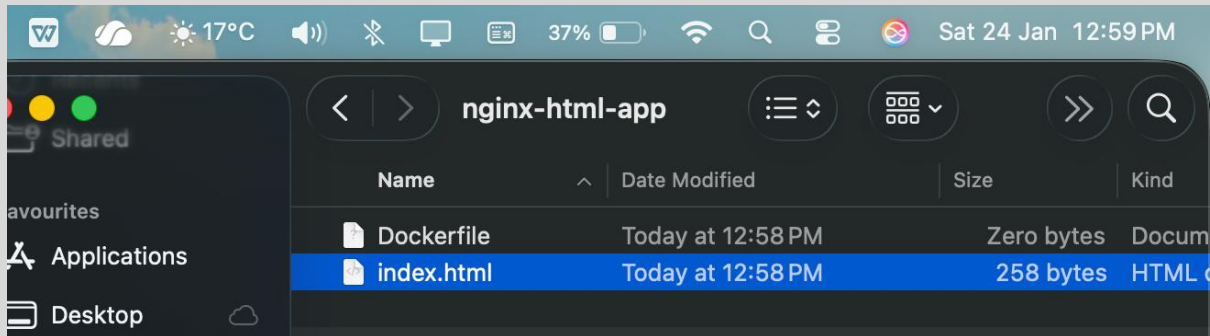
```
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to My Nginx HTML App</title>
</head>
<body>
  <h1>Hello, Nginx Docker!</h1>
  <p>This is a simple HTML app served by Nginx in a Docker container.</p>
</body>
</html>
```



3. Step 2: Create a Dockerfile

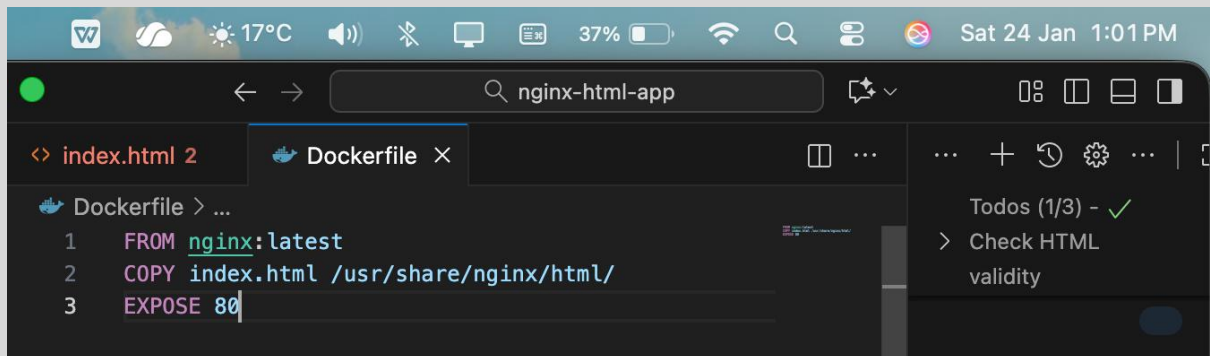
In the same directory, create a Dockerfile. This file will define how to build the Docker image using Nginx as the base image.

touch Dockerfile



Edit the Dockerfile and add the following content:

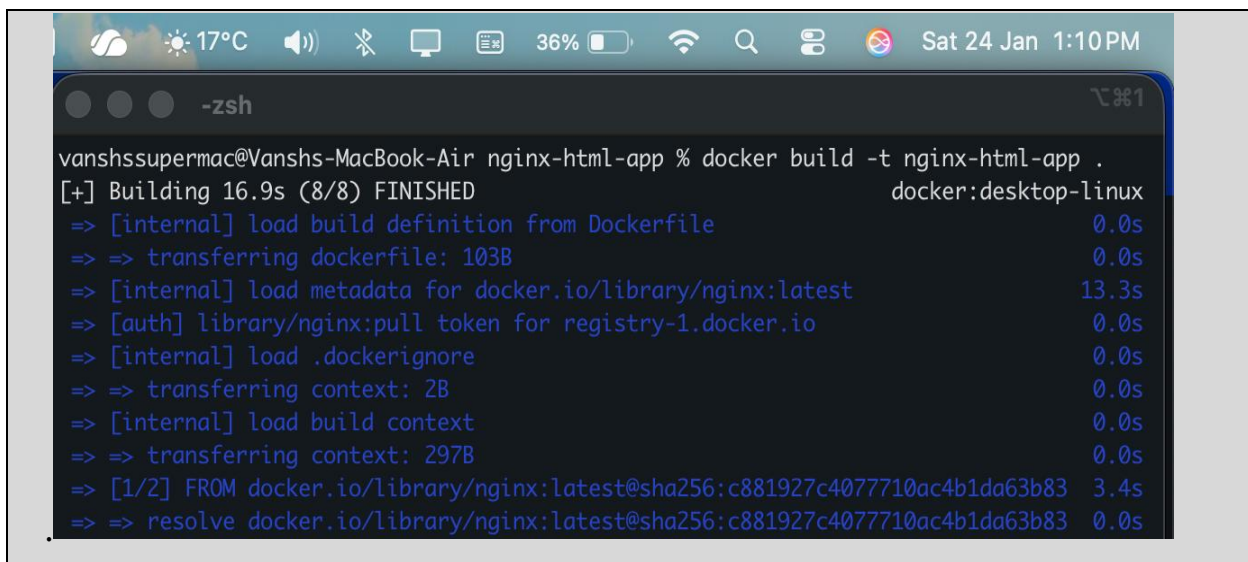
```
FROM nginx:latest
COPY index.html /usr/share/nginx/html/
EXPOSE 80
```



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
```

A screenshot of a macOS terminal window titled '-zsh'. The terminal shows the command 'docker build -t nginx-html-app .' being executed. The output indicates the build is finished in 16.9s. It details the steps: loading build definition, transferring Dockerfile (103B), loading metadata for 'nginx:latest' (13.3s), pulling the image from registry-1.docker.io, and resolving the image. The final step shows the image 'nginx:latest@sha256:c881927c4077710ac4b1da63b83' is ready.

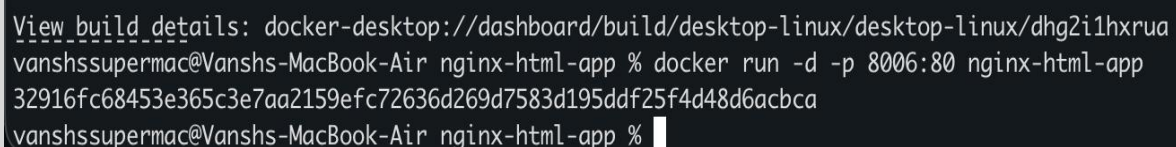
```
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % docker build -t nginx-html-app .  
[+] Building 16.9s (8/8) FINISHED                                docker:desktop-linux  
=> [internal] load build definition from Dockerfile              0.0s  
=> => transferring dockerfile: 103B                             0.0s  
=> [internal] load metadata for docker.io/library/nginx:latest  13.3s  
=> [auth] library/nginx:pull token for registry-1.docker.io    0.0s  
=> [internal] load .dockerignore                                0.0s  
=> => transferring context: 2B                                   0.0s  
=> [internal] load build context                                0.0s  
=> => transferring context: 297B                                 0.0s  
=> [1/2] FROM docker.io/library/nginx:latest@sha256:c881927c4077710ac4b1da63b83 3.4s  
=> => resolve docker.io/library/nginx:latest@sha256:c881927c4077710ac4b1da63b83 0.0s
```

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
```

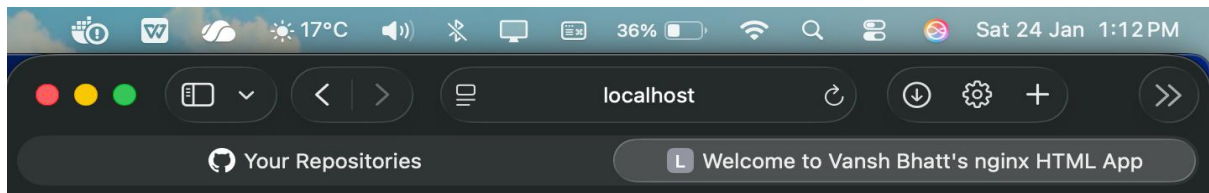
A screenshot of a terminal window showing the command 'docker run -d -p 8006:80 nginx-html-app' being executed. The output shows the container ID '32916fc68453e365c3e7aa2159efc72636d269d7583d195ddf25f4d48d6acbca'. Above the command, there is a link to view build details.

```
View_build_details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/dhg2i1hxrua  
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % docker run -d -p 8006:80 nginx-html-app  
32916fc68453e365c3e7aa2159efc72636d269d7583d195ddf25f4d48d6acbca  
vanshssupermac@Vanshs-MacBook-Air 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!”.



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:

```
docker ps # to see running containers
```

```
docker stop <container-id>
```

```
docker                                rm                                <container-  
id>
```

```
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % docker ps  
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS  
PORTS         NAMES  
32916fc68453   nginx-html-app "/docker-entrypoint...." About a minute ago Up About a minute  
0.0.0.0:8006->80/tcp, [::]:8006->80/tcp busy_khayyam  
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % docker stop 32916fc68453  
32916fc68453  
vanshssupermac@Vanshs-MacBook-Air nginx-html-app % docker rm 32916fc68453
```

Thank You