Module-3: Docker – I Assignment - 5

Assignment Submitted By:-Hitesh Chauhan
Course Offered: -Advanced Cloud Computing and Devops
Assignment By: -Intellipaat
Trainer: -Kumar
Date Of Submission: -10/02/2025

Tasks To Be Performed:

1. Create a sample HTML file
2. Use the Dockerfile from the previous task
3. Replace this sample HTML file inside the Docker container with the default page

SOLUTION

1. Create a sample HTML file

First Of We need to create the html file and name is index.html So How to create this index.file.

Go to linux Server and type the command **touch**

index.html

Then You need to write html code in this index.html.

So this is the command

Sudo nano index.html

After paste this below code and save this file and press ctrl + s key then press key ctrl + x button so you exit the command from nano.

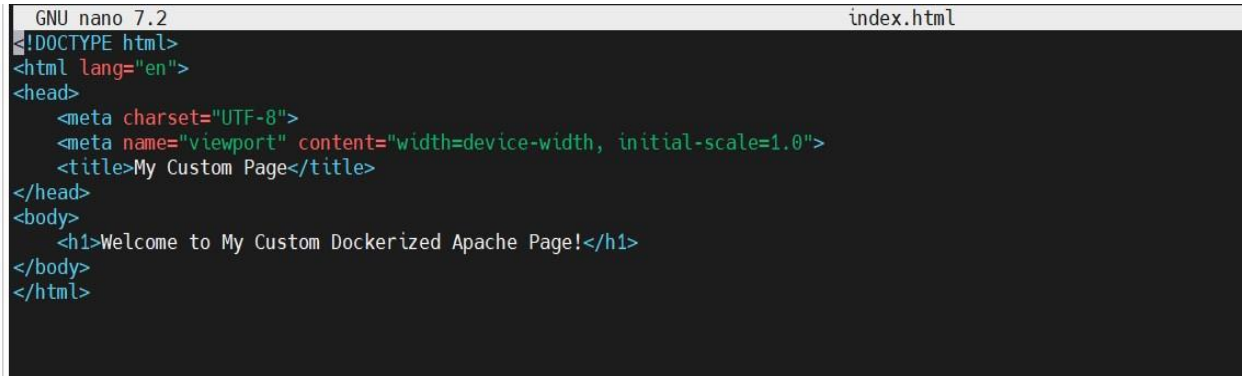
Step 1: Create a Sample HTML File

Create a file called **index.html** in your working directory:

==========

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>My Custom Web Page</title>
</head>
<body>
  <h1>Hello, Docker! This is my custom Apache page.</h1>
</body>
</html>
```

==========



Step 2: Create or Update Your Dockerfile

After creating index.html file you need to write a Dockerfile for install apache and replace with our index.html.

Modify or create your **Dockerfile** to use Ubuntu and Apache, and replace the default HTML file then you need add one line in COPY command for this **COPY index.html /var/www/html/index.html**

==========

```
# Use Ubuntu as the base image
FROM ubuntu:latest
# Install Apache2
RUN apt update && apt install -y apache2 && apt clean
# Copy custom index.html to Apache's default web directory
COPY index.html /var/www/html/index.html
# Expose port 80
EXPOSE 80
# Start Apache server in the foreground
CMD ["apachectl", "-D", "FOREGROUND"]
```

==========

```
 GNU nano 7.2                                                    Dockerfile *

# Use Ubuntu as the base image
FROM ubuntu:latest

# Install Apache2
RUN apt update && apt install -y apache2 && apt clean

# Copy custom index.html to Apache's default web directory
COPY index.html /var/www/html/index.html

# Expose port 80
EXPOSE 80

# Start Apache server in the foreground
CMD ["apachectl", "-D", "FOREGROUND"]
```

Then save this file as a DockerFile.

In this case generally created two files

1.Index.html

2.DockerFile

After We need to build the image from this docker file

So this is the command **docker Build -t my-apache-server**

```
root@ip-172-31-20-100:/home/ubuntu# touch index.html
root@ip-172-31-20-100:/home/ubuntu# sudo nano index.html
root@ip-172-31-20-100:/home/ubuntu# touch Dockerfile
root@ip-172-31-20-100:/home/ubuntu# sudo nano Dockerfile
root@ip-172-31-20-100:/home/ubuntu# docker build -t my-apache-server .
[+] Building 25.7s (8/8) FINISHED                                                                                    docker:default
 => [internal] load build definition from Dockerfile                                                                         0.1s
 => => transferring dockerfile: 370B                                                                                         0.0s
 => [internal] load metadata for docker.io/library/ubuntu:latest                                                             0.5s
 => [internal] load .dockerignore                                                                                            0.0s
 => => transferring context: 2B                                                                                              0.0s
 => [1/3] FROM docker.io/library/ubuntu:latest@sha256:72297848456d5d37d1262630108ab308d3e9ec7ed1c3286a32fe09856619a782       2.3s
 => => resolve docker.io/library/ubuntu:latest@sha256:72297848456d5d37d1262630108ab308d3e9ec7ed1c3286a32fe09856619a782       0.0s
 => => sha256:72297848456d5d37d1262630108ab308d3e9ec7ed1c3286a32fe09856619a782 6.69kB / 6.69kB                               0.0s
 => => sha256:3aff29dffbc200d202546dc6c4f614edc3b109691e7ab4aa23d02b42ba86790 424B / 424B                                    0.0s
 => => sha256:a04dc4851cbcbb42b54d1f52a41f5f9eca6a5fd03748c3f6eb2cbeb238ca99bd 2.30kB / 2.30kB                               0.0s
 => => sha256:5a7813e071bfadf18aaa6ca8318be4824a9b6297b3240f2cc84c1db6f4113040 29.75MB / 29.75MB                             0.6s
 => => extracting sha256:5a7813e071bfadf18aaa6ca8318be4824a9b6297b3240f2cc84c1db6f4113040                                    1.5s
 => [internal] load build context                                                                                            0.0s
 => => transferring context: 304B                                                                                            0.0s
 => [2/3] RUN apt update && apt install -y apache2 && apt clean                                                             21.0s
 => [3/3] COPY index.html /var/www/html/index.html                                                                           0.1s
 => exporting to image                                                                                                       1.6s
 => => exporting layers                                                                                                      1.6s
 => => writing image sha256:a4d4f6aa5c551054a9ca441220e7b58a937005f9d3458c2def3697e29f92bcd2                                 0.0s
 => => naming to docker.io/library/my-apache-server                                                                          0.0s
```

Then After Complete this building image. We

need to run this command

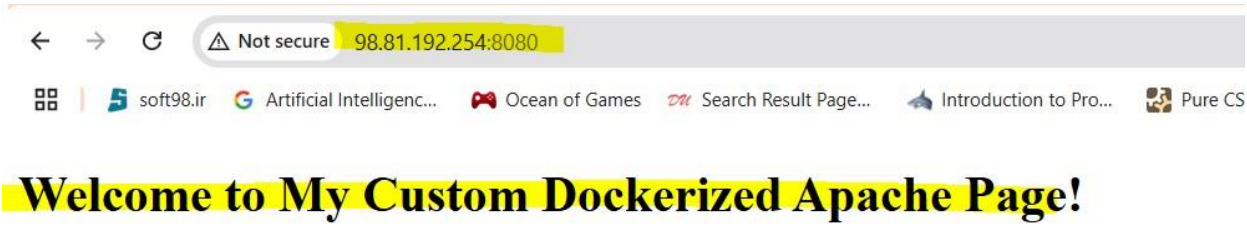
docker run -d -p 8080:80 --name apache-container my-apache-server

After run this command you will see the below container running.

```
root@ip-172-31-20-100:/home/ubuntu# docker ps
CONTAINER ID   IMAGE              COMMAND                  CREATED         STATUS         PORTS                                                  NAMES
8f2884d02383   my-apache-server   "apachectl -D FOREGR…"   7 minutes ago   Up 7 minutes   0.0.0.0:8080->80/tcp, [::]:8080->80/tcp   apache-container
```

3. Replace this sample HTML file inside the Docker container with the default page

Now After All task has been completed so we need to open the webpage from chrome browser.



Now as per the task we have open the browser with customized apache webpage from browser.