**Graded Assignment on Dockerizing a Plain HTML Page With Nginx**

1. Basic HTML page
   1. SSH into one of the EC2 instance
   2. Update the system using “sudo apt update”
   3. Make a project folder “dockerize-nginx” and switch to that.
   4. Create basic index.html file.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Hello Docker</title>

</head>

<body>

    <h1>Hello, Docker!</h1>

</body>

</html>

1. Nginx Configuration
   1. Create a “nginx.conf” with below content and configure it to listen on port 80.

events {}

http {

    server {

        listen 80;

        location / {

            root /usr/share/nginx/html;

            index index.html;

        }

    }

}

1. Dockerfile
   1. Create a “Dockerfile” to define the Docker Image

# Use the official Nginx base image

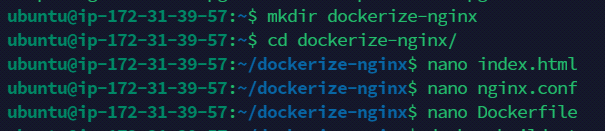
FROM nginx:latest

# Copy the custom nginx configuration file

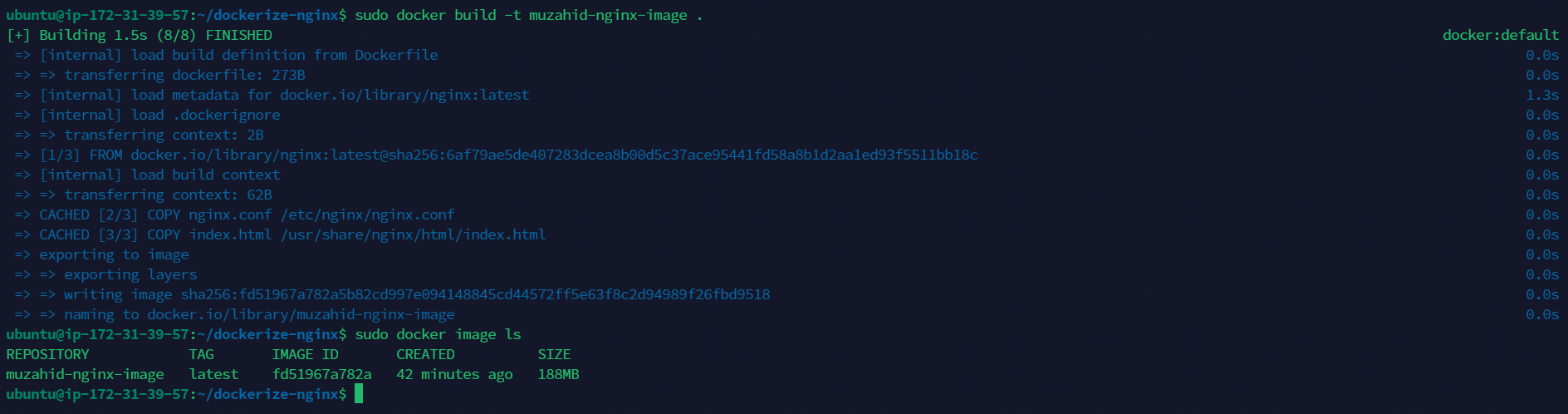
COPY nginx.conf /etc/nginx/nginx.conf

# Copy the HTML file to the Nginx HTML location

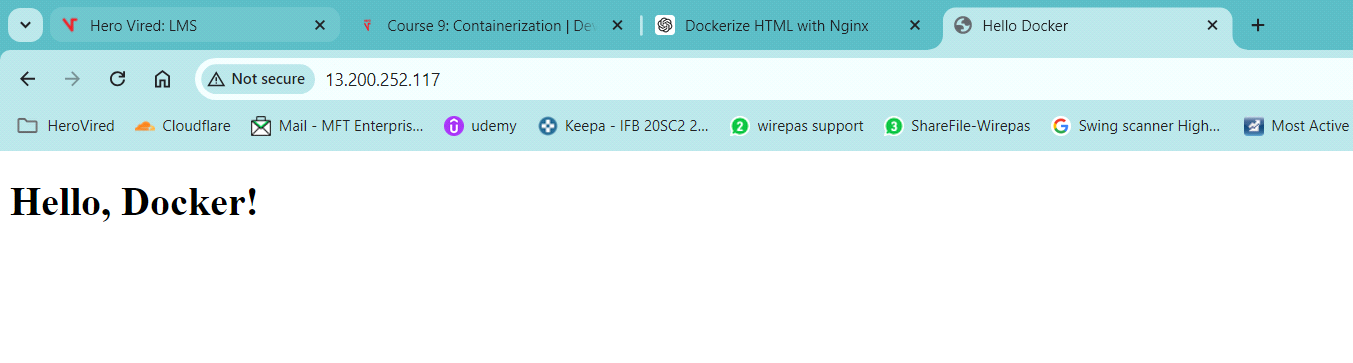
COPY index.html /usr/share/nginx/html/index.html



1. Building the Docker Image
   1. Run the below command to install docker
      1. sudo apt-get install apt-transport-https ca-certificates curl software-properties-common
      2. curl -fsSL <https://download.docker.com/linux/ubuntu/gpg> | sudo apt-key add –
      3. sudo add-apt-repository "deb [arch=amd64] <https://download.docker.com/linux/ubuntu> $(lsb\_release -cs) stable"
      4. sudo apt-get update
      5. sudo apt-get install docker-ce
   2. Build docker image using Dockerfile
      1. sudo docker build -t muzahid-nginx-image .
   3. Check created Docker Image
      1. sudo docker image ls



* 1. Optionally, we can use the below command to run the docker image and start Nginx on port 80.
     1. sudo docker run -d -p 80:80 my-nginx-image
  2. Check the docker container
     1. sudo docker ps
  3. check web page running on port 80



1. Push the image on ECR
   1. Install “AWS CLI” on ubuntu and configure it using AWS Access Key
      1. sudo apt install awscli
      2. sudo aws configure
   2. Go to AWS Amazon Elastic Container Registry and create a public repository.

A screenshot of a computer

Description automatically generated

* 1. Click on repository and check the command to push the image to this repo.
  2. Retrieve an authentication token and authenticate your Docker client to your registry. Use the AWS CLI:
     1. sudo aws ecr-public get-login-password --region us-east-1 | docker login --username AWS --password-stdin public.ecr.aws/f8g8h5d4
  3. Build your Docker image using the following command.
     1. sudo docker build -t muzahid\_custom\_nginx\_repo .
  4. After the build completes, tag your image so you can push the image to this repository:
     1. sudo docker tag muzahid\_custom\_nginx\_repo:latest public.ecr.aws/f8g8h5d4/muzahid\_custom\_nginx\_repo:latest
  5. Run the following command to push this image to your newly created AWS repository:
     1. sudo docker push public.ecr.aws/f8g8h5d4/muzahid\_custom\_nginx\_repo:latest