**Create Dockerfile and Customized index.html**

* **Directory:** dockerws

**Create a Dockerfile:**

FROM nginx

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

**Create a simple index.html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>StackSimplify</title>

<style>

body {

font-family: Arial, sans-serif;

text-align: center;

padding: 50px;

background-color: rgb(197, 144, 144);

}

h1 { font-size: 50px; }

h2 { font-size: 40px; }

h3 { font-size: 30px; }

p { font-size: 20px; }

</style>

</head>

<body>

<h1>Welcome to DevOps Docker Session</h1>

<h2>Docker Image BUILD, RUN, TAG and PUSH to Docker Hub</h2>

<p>Learn technology through practical, real-world demos.</p>

<p>Application Version: v1</p>

</body>

</html>

**Step 6: Build Docker Image and Run It**

# Change to the directory containing your Dockerfile

* cd dockerws

# Build the Docker image

docker build -t <IMAGE\_NAME>:<TAG> .

# Example:

docker build -t mynginx-custom:v1 .

# List Docker images

docker images

# Run the Docker container and verify

docker run --name <CONTAINER-NAME> -p <HOST\_PORT>:<CONTAINER\_PORT> -d <IMAGE\_NAME>:<TAG>

# Example:

docker run --name mynginx1 -p 8090:80 -d mynginx-custom:v1

# Access the application in your browser

http://localhost:8090

**Step 7: Tag and Push the Docker Image to Docker Hub**

# List Docker images

docker images

# Tag the Docker image

docker tag mynginx-custom:v1 YOUR\_DOCKER\_USERNAME/mynginx-custom:v1

# Example with 'stacksimplify':

docker tag mynginx-custom:v1 < YOUR\_DOCKER\_USERNAME >/mynginx-custom:v1

# Push the Docker image to Docker Hub

docker push YOUR\_DOCKER\_USERNAME/mynginx-custom:v1

# Example with 'stacksimplify':

docker push < YOUR\_DOCKER\_USERNAME >/mynginx-custom:v1

# IMPORTANT NOTE:

# Replace YOUR\_DOCKER\_USERNAME with your actual Docker Hub username.

# Search for 'nginx' images

docker search nginx

# Limit the search results to 5

docker search nginx --limit 5

# Filter search results by stars (e.g., images with at least 50 stars)

docker search --filter=stars=50 nginx

# Filter for official images only

docker search --filter=is-official=true nginx

<https://github.com/opencontainers/image-spec/blob/main/annotations.md>

Modify the existing image

# Use nginx:alpine-slim as base Docker Image

FROM nginx:alpine-slim

# Custom Labels

LABEL maintainer="Priyanka"

LABEL version="1.0"

LABEL description="A simple Nginx Application"

# OCI Labels

LABEL org.opencontainers.image.authors="Priyanka"

LABEL org.opencontainers.image.title="Nginx Alpine Slim Application"

LABEL org.opencontainers.image.description="A lightweight Nginx application built on Alpine."

LABEL org.opencontainers.image.version="1.0"

LABEL org.opencontainers.image.revision="1234567890abcdef"

LABEL org.opencontainers.image.created="2024-11-04T08:30:00Z"

LABEL org.opencontainers.image.url="”

LABEL org.opencontainers.image.source=" "

LABEL org.opencontainers.image.documentation="”

LABEL org.opencontainers.image.vendor="DevOps Institute"

LABEL org.opencontainers.image.licenses="Apache-2.0"

# Using COPY to copy a local file

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