Jayanthi.M

Assignment

Assignment 16/10

Write a Dockerfile that includes all major Dockerfile instructions and prints a message when the container starts.

Make the application accessible from outside the container.

Use both \*bind mounts\* and \*Docker volumes\* to store data, then verify which data remains after deleting the container.

**Write a Dockerfile that includes all major Dockerfile instructions and prints a message when the container starts.**

To create a Dockerfile we make use of below command

**vi Dockerfile**

FROM ubuntu:22.04

LABEL maintainer="jayanthi1200@gmail.com" version="1.0"

ENV APP\_NAME="PersistenceDemo"

ARG GREETING="Hello from Docker!"

WORKDIR /app

COPY ./app /app

ADD ./extra /app/extra

RUN apt-get update && apt-get install -y curl && \

mkdir -p /data\_volume && \

echo "Setup complete"

EXPOSE 8080

VOLUME /data\_volume

RUN useradd -ms /bin/bash appuser

USER appuser

ENTRYPOINT ["bash", "-c", "echo Container started: ${GREETING} && python3 -m http.server 8080"]

**A screenshot of a computer

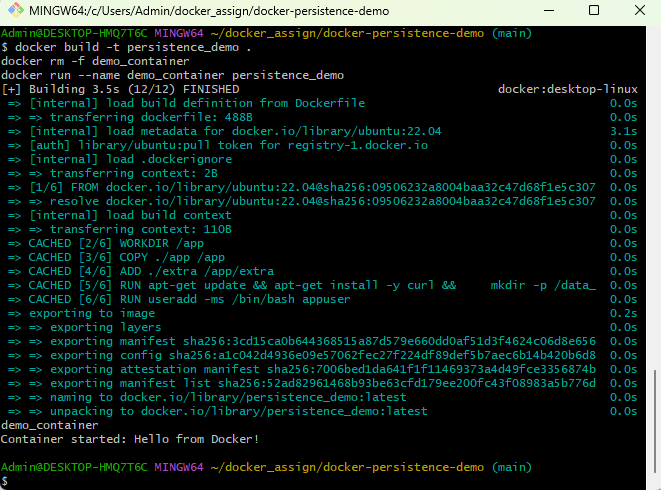
AI-generated content may be incorrect.**

**To Run the Dockerfile**

**docker build -t persistence\_demo .**

**docker rm -f demo\_container**

**docker run --name demo\_container persistence\_demo**

****

**Make the application accessible from outside the container.**

First we need to build the Dockerfile using below cmd

**docker build -t persistence\_demo .**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**docker rm -f demo\_container**

**A screenshot of a computer

AI-generated content may be incorrect.**

**docker run -d -p 8080:8080 --name demo\_container persistence\_demo**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a chat

AI-generated content may be incorrect.**

**Use both \*bind mounts\* and \*Docker volumes\* to store data, then verify which data remains after deleting the container.**

We need to add some content to bind\_data/bind\_file.txt

**echo "Hello from bind mount" > bind\_data/bind\_file.txt**

**A screen shot of a computer code

AI-generated content may be incorrect.**

To Run container with bind mount + volume

**docker build -t persistence\_demo .**

**docker rm -f demo\_container**

**A screen shot of a computer

AI-generated content may be incorrect.**

**docker run -d -p 8080:8080 \**

**-v $(pwd)/bind\_data:/app/bind\_data \**

**-v data\_volume:/data\_volume \**

**--name demo\_container \**

**persistence\_demo**

**A screenshot of a computer

AI-generated content may be incorrect.**

To go inside a container we need to use below cmd

To Run the container with bindmount and volumn

**docker rm -f demo\_container**

**docker run -d -p 8080:8080 \**

**-v ~/docker-persistence-demo/bind\_data:/bind\_data \**

**-v data\_volume:/data\_volume \**

**--name demo\_container \**

**--user root \**

**persistence\_demo**

To access container

**docker exec -it demo\_container bash**

Check Bind Mount

**ls /bind\_data**

**cat /bind\_data/bind\_file.txt**

**# Output: Hello bind mount**

Check Docker Volume

**echo "Hello volume" > /data\_volume/volume\_file.txt**

**ls /data\_volume**

**cat /data\_volume/volume\_file.txt**

**# Output: Hello volume**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A computer screen with white text

AI-generated content may be incorrect.**

**# Volume files**

**docker rm -f demo\_container**

**docker volume rm data\_volume # remove the named volume**

**docker run -d -p 8080:8080 \**

**-v ~/docker-persistence-demo/bind\_data:/bind\_data \**

**-v data\_volume:/data\_volume \**

**--name demo\_container \**

**--user root \**

**persistence\_demo**

**docker exec -it demo\_container bash**

**echo "Hello volume" > /data\_volume/volume\_file.txt**

**ls /data\_volume**

**cat /data\_volume/volume\_file.txt**

**A screenshot of a computer

AI-generated content may be incorrect.**

Verify persistence after container deletion.

To Remove Container

**docker rm -f demo\_container**

To check Check Bind mount files still exist on host:

**cat ~/docker-persistence-demo/bind\_data/bind\_file.txt**

To check Host volume files still exist (if using data\_volume/ bind mount):

**cat ~/docker-persistence-demo/data\_volume/volume\_file.txt**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**