

## 1. What is a Dockerfile?

**Answer:** A Dockerfile is a text file that contains a set of instructions on how to build a Docker image. It typically includes commands for setting up the environment, installing dependencies, copying files, and specifying the entry point.

Dockerfile

Copy code

FROM node:14

WORKDIR /app

COPY . .

RUN npm install

CMD ["node", "app.js"]

## 2. What is the difference between docker run and docker build?

**Answer:**

docker build: This command is used to build a Docker image from a Dockerfile. It creates a new image based on the instructions in the Dockerfile.

docker build -t myapp .

docker run: This command runs a container from a specified image. It can start a container, attach volumes, or set environment variables.

docker run -d -p 8080:80 myapp

## 3. Explain the concept of Docker volumes.

**Answer:** Docker volumes are used to persist data generated by and used by Docker containers. Unlike data stored inside the container, which is lost when the container is deleted, volumes are stored on the host machine and can be shared between containers.

Volumes are managed by Docker and can be created using the docker volume command.

docker volume create my\_volume

docker run -d -v my\_volume:/app/data myapp

#### 4. What is Docker Swarm?

**Answer:** Docker Swarm is Docker's native clustering and orchestration tool that allows you to manage a cluster of Docker nodes (machines) and run containers in a distributed environment. Swarm enables high availability, load balancing, service discovery, and scalability for Docker containers.

In a Swarm, containers are deployed across multiple nodes in a cluster, ensuring fault tolerance and scalability.

#### 5. How can you stop and remove a Docker container?

**Answer:** To stop a running container, use:

```
docker stop <container_name_or_id>
```

To remove a stopped container, use:

```
docker rm <container_name_or_id>
```

You can stop and remove a container in one step using:

```
docker rm -f <container_name_or_id>
```

#### 6. What is a multi-stage build in Docker?

**Answer:** A multi-stage build allows you to use multiple FROM statements in a Dockerfile to create intermediate images, each with a specific purpose.

Example:

Dockerfile

Copy code

```
# Stage 1: Build Stage
```

```
FROM node:14 AS build
```

```
WORKDIR /app
```

```
COPY . .
```

```
RUN npm install
```

# Stage 2: Production Stage

FROM node:14

WORKDIR /app

COPY --from=build /app .

CMD ["node", "app.js"]