

1 Introduction to containerization and Docker fundamentals, Basic Commands.

Answer: Basic Commands:

- (a) docker --version - Checks the version of the current installed Docker Desktop.
- (b) docker info - Checks the info of the docker images.
- (c) docker login - Login into your Docker Hub account locally through terminal.
- (d) docker logout - Logout from your Docker Hub Account Locally.
- (e) docker system prune - Delete all the available container.

2. Docker installation and basic container operations, Build an image from Dockerfile.

Answer: Build Image From DockerFile:

```
docker images
docker pull alpine
docker build -t alpine
docker run -p 3000:8000 alpine
docker stop 36589455451
docker rm 12564724
```

3. Docker Registry, DockerHub, Create a Multi-Stage Build.

Answer: Multi-Stage Build:

```
DockerFile Locally:
FROM golang:alpine AS build
WORKDIR/app
COPY . .
RUN go build -o main
```

```
FROM alpine
WORKDIR/app
COPY --from = build/app/main
CMD["./main"]
```

4. Create a docker image from multiple methods likes Dockerfile, running containers.

Answer: Docker Image from docker file:

```
Base Image
WORKDIR/app
COPY . .
My.json
CMD["npm","Start"]
```

Docker Image from running container:

```
docker run -it alpine
docker commit 56755527464 my_committed_image
```

5. Push and pull image to Docker hub and ACR.

Answer: Push an image to Docker Hub:

```
docker push my_repo img1
```

Pull an image to Docker Hub:

```
docker pull my_repo
```

6. Create a Custom Docker Bridge Network.

Answer: Docker Bridge Network:

```
docker network create my_custom_net
docker network ls
My_custom_net
```

7. Create a Docker volume and mount it to a container.

Answer: Docker Volume:

```
docker volume create my_vol
docker run -d -v my_vol: / data alpine sleep 1000
docker ls volume
my_vol
```

8. Docker Compose for multi-container applications, Docker security best practices.

Answer: Docker Compose:

`docker -compose.yml`