***What is container***

=container provides isolated environment to run an application.

= container contains everything needed to run an application –code, dependencies, version and libraries.

• Containers:

• Share the host OS kernel.

• Lightweight and fast to start.

• Suitable for micro services and modern cloud applications.

• Virtual Machines (VMs):

• Include a full OS with its own kernel.

• More resource-intensive and slower to start.

• Useful for running multiple different os on a single host

***What is Dockers***

Docker is open-source platform that allows you to build, test and deploy applications quickly by container.

Docker is a tool it allows to create a container, manage container, scale container.

Docker enables you to separate your applications from your infrastructure so you can deliver software quickly.

***Why use Docker***

Using Docker can help you ship your code faster, gives you control over your applications.

You can deploy applications on containers that make it easier for them to be deployed, scaled, perform rollbacks and identify issues.

1. Docker Engine: The runtime that executes and manages containers on a host operating system.

It consists of the Docker daemon (dockerd), which manages Docker objects (images, containers, networks, and volumes), and the Docker CLI, which allows users to interact with the Docker daemon.

2. Docker client: is the primary way to interact with Docker. Docker client can communicate with multiple daemon

3. Docker Daemon: It listen of Docker API request and manages docker object such as images, containers, and network. A daemon can also communicate with other daemons to manage Docker services.

It is responsible for building, running, and managing Docker containers.

4. Docker registries: Docker registry stores Docker images. Docker Hub is a public registry that anyone can use, and Docker looks for images on Docker Hub by default.

Q What is docker file?

Docker can build images automatically by reading the instructions from a Dockerfile.

A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image.

Comands of docker

docker run <image name>..........to create container with image

docker ps ........to see running containers

docker ps -a ....to see all the containers

docker run -d <image name> ....... create container in the background.

docker exec -it contianerID ...........login into the container

docker run -p hostport:container port <imagename> ........ mapp port.

docker rm containeid ....... to remove container.

docker start container

docker stop container

docker inspect containeID .........to see all congirue of container

docker images ...........to see images

docker rmi imagename ............to delete images

docker build <dockerfile> ....build a image from dockerfile.

docker commit <containerID> .build image from container

docker volume create <volumename> ........... to create new volume

docker volume ls .........list volumes.

docker volume inspect <volume id> ……..to see volume details.