1. What is Docker?

Docker is a set of Platform as a service tools that use Os-level virtualization to deliver software in packages as containers.

1. What are Primary name spaces ?

pid - process

mnt - all mounted folders

net - all physical ethernet card - eth0 , ens33

UTS - Unix Time sharing - hostname/ domain name

IPC - memory sharing - Interprocess Communication

1. Which is your docker image repository ?

**Harbor**

ECR

Docker Hub

1. Image writing codes (Dockerfile) ?

1. Image build and run commands ?

docker build -t web1:v1 /docker1

docker run -it -d –name webcon -p 8080:80 web1:v1

1. How to check docker Ram and processor ?

docker stats

1. Container Life cycle commands ?

1. run - to run the docker container

docker run -it -d –name webcon -p 8080:80 webserver1:v1

1. start

docker start containerid

1. stop

docker stop containerid

1. restart

docker restart containerid

1. pause - to stop the traffic to the container

docker pause containerid

1. unpause

docker unpause containerid

1. rm

docker rm containerid

1. rm -f

docker rm -f containerid

1. exec (to login inside the container)

docker exec -it <containerid> /bin/bash

exit (to exit the container)

1. logs and inspect

docker logs <conaintername/id >

docker inspect <conaintername/id >

8. how to limit cpu and processor to a container ?

docker run -it -d --name dbserver2 -e MYSQL\_ROOT\_PASSWORD='admin123' **--cpus 0.5 --memory 200mb**  -p 3306:3306 mysql:5.7

9. Construct a 2 tier app ?

* java container
* mongo db container

10. Docker networking ?

* Bridge -(make the internal communication between containers)
* host - (directly connect the container to host IP)
* null - No IP

1. What is restart policy ?

* If container crashes the restart policy will help to bring back the container until it comes in healthy state!

1. Docker Storage ?

* Local
* Bind
* NFS (remote storage)

1. Docker Compose ?

* Using docker compose we can create and link multiple containers using yaml file.

for example:

web:

  name: test

  network: newnet

  image: centos:7

  ports:

     ingress: 9090

     container: 80

1. docker commit ? (like AMI)

     Taking snapshot of the container for any changes or upgrade.

docker commit <containername> app:backup1 (it will create an image from the container)

1. Docker registry ?

      Creating our own registry to maintain the docker images

docker run -d -p 5000:5000 --name registry registry:2

1. Docker swarm: (replica) ?

It’s an orchestration tool!

       Instead we are using **kubernetes**.

17 Difference between CMD AND ENTRYPOINT?

**CMD**  => sets default parameters that can be overridden when a container is running.

**ENTRYPOINT** => sets default parameters that cannot be overridden when a container is running.

   CMD  => CMD commands are ignored when there are parameters stated within the docker run command.

ENTRYPOINT => Instructions are not ignored and are instead treated as arguments