

# Module-6: Containerization using Docker

---

Demo Document - 1

edureka!

**edureka!**

© Brain4ce Education Solutions Pvt. Ltd.

## DEMO-1: Docker CLI Cheat Sheet

### Docker Containers

```
docker search # search the images in docker hub
docker pull   # pull image from docker hub
docker images # list the images in local
docker run (interactive / detached ) # run a container from an image
docker ps     # list the running containers
docker ps -a  # list running + exited containers
docker inspect # inspect a container
docker stop|start|restart # stop/start/restart a container
docker rm / docker rm -f # remove the containers from the server
docker exec   # get inside a running container
```

### Docker Images

```
docker images # list images in local
docker commit ( manual ) # create image from container changes
docker build ( automated ) # create image using a dockerfile
docker history # view the layers of an image
docker inspect # inspect an image
docker rmi     # remove image from local
```

### image repository

```
docker hub (public) # docker image repository
docker registry container (private) # docker image repository open source
DTR (private) # docker image repository licensed tool from docker org
    docker login # login to image repo from command line
    docker tag   # create alias name for an image in local
    docker push  # push images to docker image repository
```

### Docker Volumes

```
docker volume ls # list volumes in local
docker volume create # create a volume
docker inspect volume # inspect the volume
```

### Docker COMPOSE:

```
case 1: deploy multiple containers from a single image
    docker-compose -f docker-compose.yml up --scale web=4 -d ; docker-compose down
case 2: deploy multiple containers from multiple images
    docker-compose -f docker-compose.yml -p webapps up -d --scale web=2 --scale app=2
```

### Docker Networking:

```
docker network ls # list the default & custom networks on a docker host
    none|host|bridge|overlay|docker_gwbridge
docker network create -d <driver> <network name> # create a custom network of
```

#### bridge/overlay

```
docker run -d --net host --name cont1 alpine # attach a container to host network
docker run -d --net none --name cont2 alpine # attach a container to none network
docker run -d --net ravinet --name cont3 alpine # attach a container to custom
```

#### bridge/overlay network

### Docker Swarm

```
docker swarm init ## initialize the swarm mode (swarm manager)
docker swarm join ## join a node to manager as worker / manager
docker swarm join-token worker/manager ## generate worker/manager token
docker swarm leave ## run on worker nodes to leave the node from swarm

docker node ls ## list the node part of swarm
docker node inspect ## inspect a node
docker node rm <nodename> ## run on manager node to remove a node from swarm
```

```
node    docker node ps          ## List tasks running on one or more nodes, defaults to current  
  
        docker service create  ## create service in docker swarm cluster  
        docker service ls      ## list the running services  
        docker service ps <service name> ## list the containers running inside a service  
        docker service rm      ## remove the service
```

edureka!