Docker - quick command lookup - Day 1

|  |  |
| --- | --- |
| **Command** | **Description** |
| docker version | Version of docker |
| docker info | A lot details |
| docker | List of commands |
| docker container ls | List of running containers |
| docker container run nginx | Run a container |
| docker container run --publish 80:80 nginx | Open port 80 |
| docker container run --publish 80:80 --detach nginx | detached mode |
| docker container run -p 80:80 nginx | -p is same as --publish |
| docker container run -p 80:80 -d nginx | -d is Same as --detached |
| docker ps | Old way of listing containers |
| docker container stop <container ID> | Stop the running container |
| docker container start <container id> | Start the container |
| docker container ls -a | List all the containers |
| docker container run -p 80:80 -d -name web1 nginx | Custom name to a container |
| docker login | Log in to docker hub |
| docker logout |  |
| docker container run -ti nginx bash | Start a container in interactive mode and attach a tty |
| docker container exec -it nginx bash | Connect interactively to a running container |
| docker image ls | List all the images |
| docker image history <image id> | How is the image built |
| docker image inspect <image id> | Details about an image |
| docker image pull image:tag | Pull a particular image |
| docker image tag customimage vijaytrainer/customimage | Tag an existing image. defaults to latest |
| docker image push vijaytrainer/customimage | Push the image |
| docker system prune> | Delete all the stopped containers and dangling images |
| docker container rm <container id> | Delete a containercd |
| docker login | Log in to docker hub |
| docker logout |  |
| docker container port <container id> | Show the ports |
| docker network ls | List all the networks |
| docker network inspect | Shows details about the network and all containers running on them |
| docker network connect network\_id container\_id |  |
| docker network disconnect network\_id container\_id |  |
| docker network create my\_own\_net | Creates a virtual network to which containers can attach |
| docker container run -d --network my\_own\_net nginx | Puts the docker on a network |
| docker container logs web1 | Show the log |
| docker container inspect <container id> | Details of a container |
| docker container stats <container id> | Performance stats of a container |
| docker inspect <container id> | Old way |