

Docker Commands



IMAGES :

List all Local images

`docker images`

Delete an image

`docker rmi <image_name>`

Remove unused images

`docker image prune`

Build an image from a Dockerfile

`docker build -t <image_name>:<version> .` //version is optional

`docker build -t <image_name>:<version> . -no-cache` //build without cache

CONTAINER :

List all Local containers (running & stopped)

`docker ps -a`

List all running containers

`docker ps`

Create & run a new container

`docker run <image_name>`

//if image not available locally, it'll be downloaded from DockerHub

Run container in background

`docker run -d <image_name>`

Run container with custom name

`docker run - -name <container_name> <image_name>`



Port Binding in container

`docker run -p<host_port>:<container_port> <image_name>`

Set environment variables in a container

`docker run -e <var_name>=<var_value> <container_name> (or <container_id>)`

Start or Stop an existing container

`docker start|stop <container_name> (or <container_id>)`

Inspect a running container

`docker inspect <container_name> (or <container_id>)`

Delete a container

`docker rm <container_name> (or <container_id>)`

TROUBLESHOOT :

Fetch logs of a container

`docker logs <container_name> (or <container_id>)`

Open shell inside running container

`docker exec -it <container_name> /bin/bash`

`docker exec -it <container_name> sh`

DOCKER HUB :

Pull an image from DockerHub

`docker pull <image_name>`

Publish an image to DockerHub

`docker push <username>/<image_name>`



Login into DockerHub

`docker login -u <image_name>`

Or

`docker login`

//also, `docker logout` to remove credentials

Search for an image on DockerHub

`docker search <image_name>`

VOLUMES :

List all Volumes

`docker volume ls`

Create new Named volume

`docker volume create <volume_name>`

Delete a Named volume

`docker volume rm <volume_name>`

Mount Named volume with running container

`docker run - -volume <volume_name>:<mount_path>`

//or using `- -mount`

`docker run - -mount type=volume,src=<volume_name>,dest=<mount_path>`

Mount Anonymous volume with running container

`docker run - -volume <mount_path>`

To create a Bind Mount

`docker run - -volume <host_path>:<container_path>`

//or using `- -mount`

`docker run - -mount type=bind,src=<host_path>,dest=<container_path>`

Remove unused local volumes

`docker volume prune` //for anonymous volumes



NETWORK :

List all networks

`docker network ls`

Create a network

`docker network create <network_name>`

Remove a network

`docker network rm <network_name>`

Remove all unused networks

`docker network prune`