docker –version => version check of installed docker

docker images => show all images

docker ps -a => all containers

docker run hello-world => if hello-world image is not found in my local , this command pull hello- world image pull docker hub and make it a container and run it

docker build -t name of image . => create an image based on docker file (-t to tag /named the image (can give name I want) , .(dot) for directory of dockerfile )

docker run --name hello-docker-container -p 8080:80 hello-docker:1.0.0

Give container name for a docker image

Image name cmnd

Docker run cmnd for a image

Define port cmnd. Where 8080 in our local machine and 80 is container port. Listing fro 80 and maped to 8080

docker stop 78e

Container Id

Docker stop cmnd for a container

docker start 78e

Container Id

Docker start cmnd for a container

docker restart 78e

Container Id

Docker restart cmnd for a container

docker rm 1be = > remove one container

Container Id

docker ps -a -q => cmnd for get all container id

docker rm $(docker ps -a -q) => remove all container

docker rmi 168 => remove one image

image Id

docker images -q => get all images id

docker rmi $(docker images -q) => remove all images

**docker buid command with options:**

**docker build -f D:\Self\Programming\Docker\dockerfile --label "zakaria'sImage" --label "Builtfors3innovate" -q --rm=false -t new-image:latest -t new-image:1.0.0 -t hello-world:lastest .**

* **-f D:\Self\Programming\Docker\dockerfile => file location of docker file**
* **--label "zakaria'sImage" --label "Builtfors3innovate" => Set metadata for an image**
* **-q => Suppress the build output and print image ID on success**
* **--rm=false => Remove intermediate containers after a successful build (by default true)**
* **-t new-image:latest -t new-image:1.0.0 -t hello-world:lastest => Name and optionally a tag in the 'name:tag' format**
* **. (dot)** => **directory of dockerfile**