# **Docker**

**Image** : build time object

**Container**: runtime object

It’s best to have 1 image per container

* docker *command* [option] –help

Displays help about the command or command option

* docker image build -t marcusquigley/gsd:ctr --no-cache .

Creates a docker image called ctr for my repository. It’s a container. Create a tag and don’t use any cached info. Finally the . states that it uses all files/folders in current path. Docker then reads the docker file and runs thru the commands

* docker image ls

Displays all docker images on machine

* docker image push marcusquigley/gsd:ctr

Pushes image to docker repo

* docker images –a

Lists all images

* docker rmi *imageid :* removes image (container must be deleted first)
* docker ps – :lists running containers
* docker ps –a :lists all containers
* docker rm *containerid :* removes a stopped container
* docker container prune: removes all stopped containers

To containerize and host an app: [**NOTE need to have docker file in current path**]

1. docker image build -t marcusquigley/gsd:first-ctr --no-cache **.**
2. docker image push marcusquigley/gsd:first-ctr

Run a containerized app

* docker container run -d --name web -p 8000:8080 marcusquigley/gsd:first-ctr

This runs container with tag of first-ctr in gsd repo. All calls to port 8000 in docker gets forwarded to 8080 in app inside. The container is called web

* docker container stop *containername :* stops the specified container
* docker container start *containername :* starts the specified container
* docker container ls : displays all running containers
* docker container ls -a: displays all containers