Docker

- `docker login` log in docker repo
- `docker system info` check docker information
- `docker --version` or `docker -v` check docker version
- `docker port [container]` set container port

Image

- `docker image ls` check download images

    - `-a` show all images
    - `--digests` show digests
    - `--filter dangling=true` filter
    - `--format "{{.Repository}}:{{.Tag}}:{{.Size}}"` see info in specific format
    - `-q` only show image id

- `docker image pull [image]:[version]` pull image from repo or dockerhub

- `docker search [image]` search image

- `docker image inspect [image]` see more info about image

- `docker image rm [image]` or `docker rmi [image]` remove image

- `docker image build -t [dockerfile]` build new image by dockerfile

- `docker image push [new_image]` push your image into dockerhub

- `docker image history [image]` see image history

- `docker image rm $(docker image ls -q)` clear all images

- `docker image prune` deletes all dangling images

  - `-a` delelte dangling images and unused images

- `docker tag [image]:[tag] [repo]:[tag]` tag image

Container

- `docker container ls` check created containers

  - `-a` show all containers

- `docker container run -it [image]` create new container

  - `--restart` restart the container
  - `--name [name]` define the name of container
  - `-p 80:8080` give access port
  - `-d` run but not attach in container
  - `-v [local_path]:[container_path]` use volume to map with the container directory and local directory
  - `--network [network]` set WLAN

- `docker attach [container]` go into container

- `docker exec -it [container] bash` go into container interface

- `docker start [container]` start container

- `docker container stop [container]` stop container

- `docker container rm [container]` remove container

- `docker commit [container] [new_image:tag]` commit container as a new image

  - `-a [author]` author
  - `-m [message]` message

- `docker container prune` remove all stopped containers

- `docker rm -f $(docker ps -aq)` clear all containers

- `docker container exec -it [container] [command]` execute command in selected container

- `docker rename [old_name] [new name]` rename the container

- `docker cp [container]:[file_path] [local]` copy file between local and container
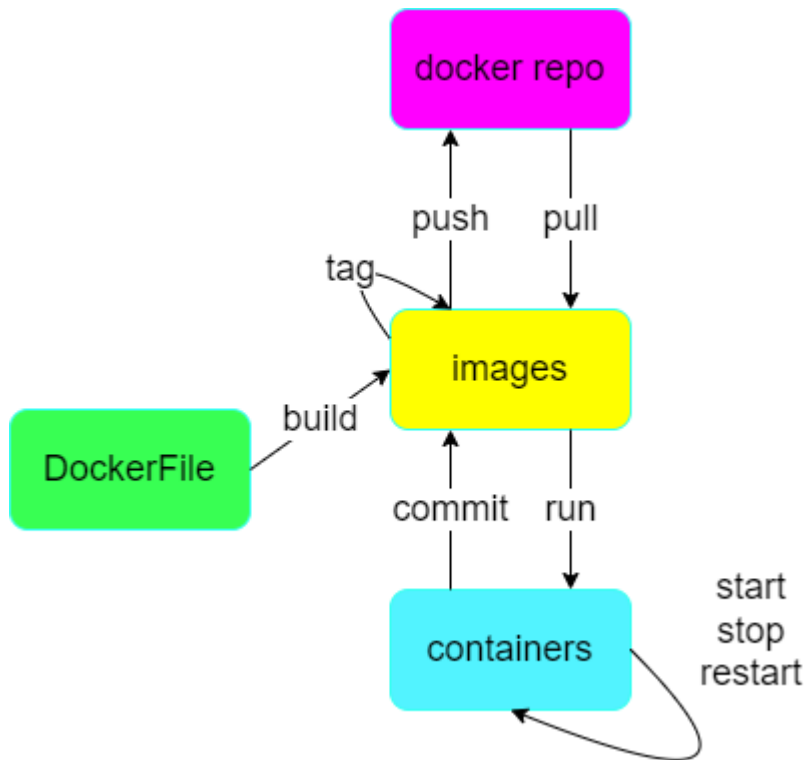
Volume

- `docker volume ls` see the volume
- `docker volume create [volume]` create new volume
- `docker volume inspect [volume]` see volume info
- `docker volume rm [volume]` remove volume
- `docker volume prune` remove volume not used in container

network

- `docker network ls` see the network configuration
- `docker network create [name]` create new network(base on bridge)
- `docker network inspect [name]` see network
- `docker network connect [name] [container]` connect container into network

- `docker network disconnect [name] [container]` disconnect container from network
- `docker network rm [name]` remove network