

## Containers

`docker ps` : listing containers  
    `-a` : list all, running and stopped

`docker run <image>` : run an image  
    `-i` : interactive mode  
    `-t` : expose TTY  
    `-p` : expose port `ext:int` eg. `3000:80`  
    `-d` : detached mode  
    `-v <abs-lpath>:<cpath>` : create bind mount  
    `-v <name>:<cpath>` : create named volume  
    `-v <cpath>` : create anonymous volume  
    `-rm` : automatically remove when exits  
    `-name <string>` : give container a name

`docker rm <container>` : remove container

`docker stop <container>` : stop running container

`docker start <container>` : start stopped container

`docker attach <container>` : attach to the running container

## Dockerfile

`FROM <base-image>` : eg. `node`

`WORKDIR /app` : Commands will be executed relatively to this dir

`COPY <src> <src>: src (<dest> (/app))`

`RUN npm install` : runs when image is created

`EXPOSE 80` : exposes port 80

`VOLUME [<path>, ...]` : create anonymous volume

`CMD [<node>, <server.js>]` : runs when container is started. Should always be the last

## Images

`docker build <src>` : build docker image; `src(.)` - usually the `Dockerfile` is in the root  
    `-t <name>:<tag>` : tag an image

`docker images` : list all images

`docker rmi <image-id>` : remove image

`docker image inspect <image-id>` : details of the image

`docker image prune` : remove image  
    `-a` : remove all images

## DockerHub & Sharing

`docker push <image-name>` : Push image to DockerHub

`docker pull <image-name>` : Pull image from DockerHub

`docker login` : Login into DockerHub

`docker logout` : Logout from DockerHub

## Volumes

`docker volume ls` : List volumes

`docker volume rm` : Remove volume

## Other

`docker logs <container>` : fetch the logs of the container  
    `-f` : follow

`docker cp <src> <dest>` : copy files to / from the container; `src/dest (<container-name>:<path> | <path>)`