Contents

1 Basics

- Docker: Set of Platform as a Service (`PaaS`) product delivering software in packages called `containers`
- Architecture:
  - Client-server architecture
  - Docker client talks to the `Docker daemon`, which builds, runs, and distributes the Docker containers
  - Docker client and daemon communicate using `REST API`
  - `Docker Compose` is another client to work with applications consisting of a set of containers
- Docker daemon (`dockerd`)
  - manages `images, containers, networks,` and `volumes`
  - can communicate with other daemons to manage Docker services
- Docker client (`docker`)
  - sends commands, e.g. `docker run` to `dockerd`, which carries them out
  - uses Docker API
  - can communicate with more than one daemon
- Docker registries stores Docker `images`

1.1 Docker objects

1.1.1 Images

- Read-only (immutable) template with instructions for creating a Docker `container`
- Often based on another `image` with additional customization
- Built from an instructional file named `Dockerfile`
- Each instruction in an `Dockerfile` creates a `layer` in the `image`
- Each `layer` represents a set of file system changes that add, remove, or modify files
- Only changed `layers` will be rebuild when changing the `Dockerfile` and rebuilding the `image`

1.1.2 Containers

- Runnable instance of an `image`
- Can be connected to one or more networks
- Isolated from other `containers` and its host machine
- Defined by its `image`
- Include all of the dependencies required for an app to work

1.1.3 Dockerfile

- Provide the instructions needed to build an `image` which is then the recipe for a container
- Are written by the user whereas an `image` is written by the machine based on the `Dockerfile`
- Syntax:

```
FROM <image>:<tag>
RUN <install some depedencies>
CMD <command that is executed on ‘docker container run‘>
```

1.2 Docker CLI basics

- Docker Engine is made up of `CLI` client, a `REST API`, and a `Docker daemon`
- When running `docker container run`, the CLI sends a request through REST API to the Docker daemon which takes care of images, containers and other resources
- List all containers:

```
docker container ls     # list only running containers
docker container ls -a # list all containers
docker container ls -a | grep <expr> # filtering for <expr>

docker ps (-a)          # shorter form
```

- Remove containers:

```
docker container rm <id1> <id2> <id3>
docker container prune # deletes all stopped containers
docker
```

- Remove images:

```
docker image rm <image>
docker image prune # removes unnamed and not used images
```

1.2.1 Most used commands

Command	Description	Shorthand
`docker image ls`	Lists all images	`docker images`
`docker image rm <image>`	Removes all images	`docker rmi`
`docker image pull <image>`	Pulls an image from a docker registry	`docker pull`
`docker container ls -a`	Lists all containers	`docker ps -a`
`docker container run <image>`	Runs a container from an image	`docker run`
`docker container rm <container>`	Removes a container	`docker rm`
`docker container stop <container>`	Stops a container	`docker stop`
`docker container exec <container>`	Executes a command inside a container	`docker exec`