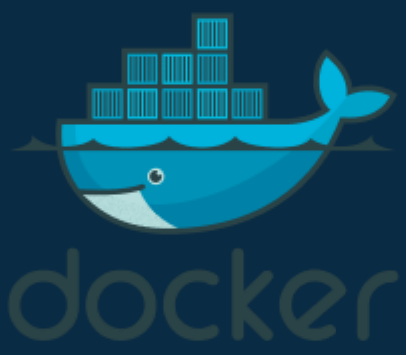
Notes

Images

Containers

Registry

Volumes



Compose

Dockerfile

Daemon

Client

Docker image

A read only template with instructions for creating a Docker container. In other words, a **docker image** is a docker object which is a template for creating containers.

`Dockerfile`

Docker file is a blue print for creating docker images.

Each instruction on a dockerfile creates a layer in the image.

If you change the dockerfile and rebuild the image, only the changed layers are rebuilt. This makes images lightweight, small and fast.

It is always named `Dockerfile` without an extention.

Docker Container

A container is a runnable instance of an image.

It is a normal operating system process except that this process is isolated and has its own file system, its own networking, and its own isolated process tree separate from the host.

Containers are lightweight and contain everything needed to run the application, and it becomes the unit for distributing and testing your application.


docker build
docker run
Dockerfile
Image
Container

Docker daemon

Docker daemon - `dockerd` listenes to Docker API requests and manages Docker objects like **images, containers, networks and volumes**.

Docker client

Docker client `docker` is how users interact with docker.

Docker client sends commands such as `docker run` to `dockerd` to be carried out.

Docker client can communicate with more than one docker daemon.

Docker registry

A Docker registry stores Docker images.

example: Docker Hub

Docker compose

Docker compose is a yaml file to run multiple containers.

It can be defined outside the application code.

Docker volume

Docker volume is a folder in host file system mounted to the virtual file system of docker container.

Data get automaticaly replicated in the host file system.

What are docker objects?

Docker images, containers, networks, volumes, plugins are all different docker objects.

When you use Docker, you are creating and using images, containers, networks, volumes, plugins, and other objects. [Docker overview | Docker Documentation]

Docker Image vs Docker Container

Docker images are a type of docker object. A docker image, once created cannot be changed (immutable). This makes it a read-only object and it contains the source code, dependencies and everything else needed to run the application.

Docker containers are a different kind of docker object. Basically it is made up of the read-only image and a writable layer on top of it.

Docker Image Layers

A docker image consists of several different layers. These layers are called intermediate layers of an image.

The layers exist as files that can be found in the docker host at `/var/lib/docker/aufs/diff`

Once the image is built, you can view all the layers that make up the image with the following command.

```
docker history IMAGE/IMAGE_ID
```

eg: `docker history mysql:8.0`

For more content like this Follow me on

LinkedIn