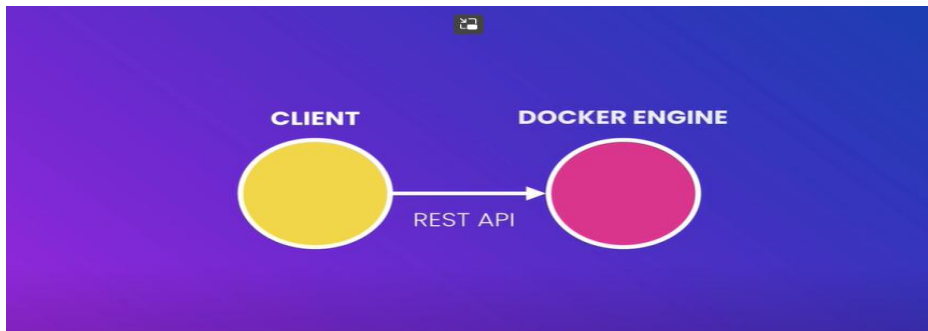


Docker

Docker

- What is Docker? Docker is a platform for building, running applications and shipping application for productions.
- What Docker can solve:
 - One or more files is/are missing
 - Software version mismatch
 - Dependencies of different frameworks
 - Different configuration settings
- Container
 - is an isolated environment for running an application.
 - It is similar to virtual machine but it is more light weight.
 - Use OS of the host.
 - Needs less hardware resources.
 - Different between docker container and virtual machine is that virtual machine is an abstraction of a machine (physical hardware). Virtual machines are using Hypervisors (VirtualBox, VMware, Hyper-v (Windows only))
- Docker Architecture
 - Docker uses a client server architecture so it has a client component that talks to a server component using restful api, the server also called docker engine sits on a background and building and running docker containers.
 - But technically a container is just a process like processes running on your computer.
 - Containers don't contain a full-blown operating system, instead all containers on a host share the operating system of the host. More accurately all these containers share the kernel of the host. Kernel is the core of an operating system.
 - Kernel manages applications and hardware resources like memory and cpu.
 - Every OS has its own kernel or engine and these kernels have different apis and that's why we cannot run a windows application on linux because under the hood this application needs to talk to the kernel of the underlying operating system so that means on linux machine we can run linux containers because these containers need linux.



- Image
 - A cut-down OS
 - A runtime environment (eg Node)
 - Application files
 - Third-party libraries
 - Environment variables
 - We tell docker to start a container using an image. Because container is just a process.
- Dockerfile
 - It is a process to package our app.
 - It is an instruction (eg. Make a simple nodejs app to print "Hello Docker")
 - ◆ Start with an OS
 - ◆ Install Node
 - ◆ Copy app files
 - ◆ Run node app.js
 - Dockerfile Example:

```
FROM node:alpine
COPY . /app
WORKDIR /app
CMD node app.js

# FROM = You need to specify your base image here. Maybe an OS or the
# NodeJS framework
# if we use node as base image. It will automatically built on top of
# linux
# COPY - it accepts two arguments source => dest
# WORKDIR - where we specify the current directory inside of a docker
# container
# CMD - it accepts arguments to process as commands
# Type "docker build -t hello-docker ."
# After it is build on a docker you can run it using "docker run hello-
# world"
```

- Remove docker containers
 - `docker container rm -f $(docker container ls -aq)`

- Remove docker images
 - `docker image rm -f $(docker image ls -aq)`
- Docker Compose
 - We use docker-compose when we are working on a multiple containers

```
: The Ultimate Docker Course
price: 149
is_published: true
tags:
  - software
  - devops
author:
  first_name: Mosh
  last_name: Hamedani

# How do we represent array and object in a yaml?
# example of array in a yaml is
#   tags:
#     - first_item
#     - second_item
# example of object in a yaml file
# author:
#   first_name: Mosh
#   last_name: Hamedani
```

■
Docker Compose -

Docker commands

Docker command examples: