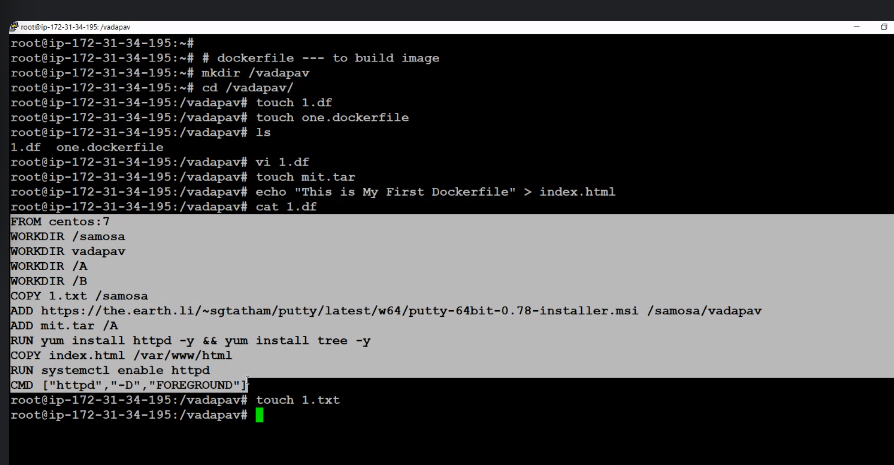
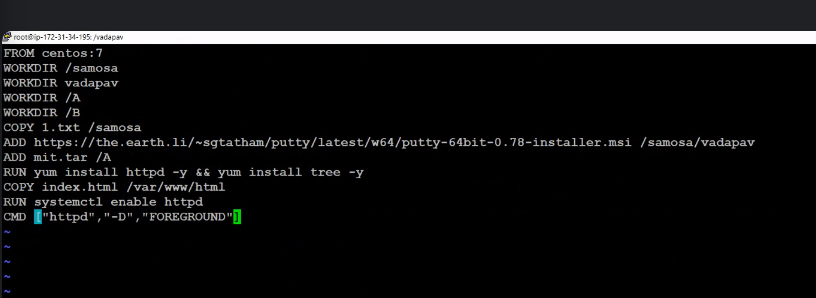
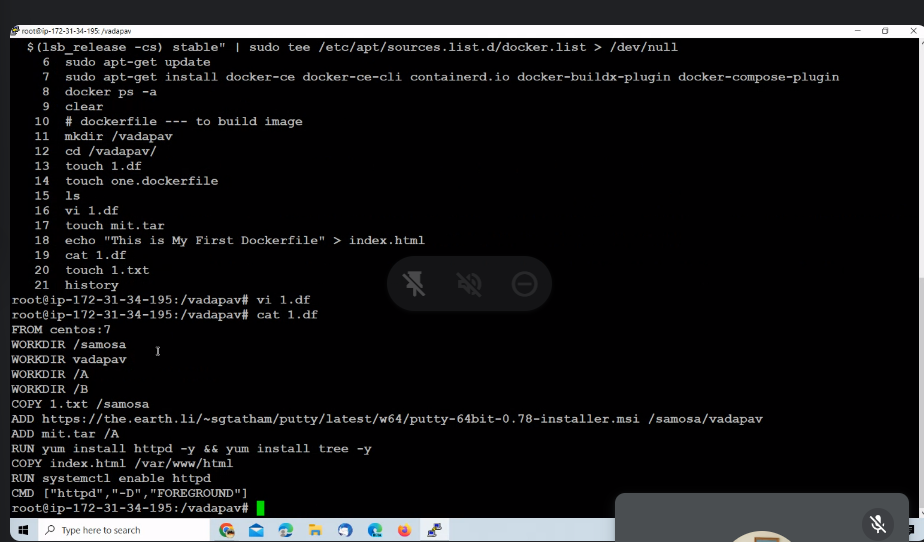
images are tempests

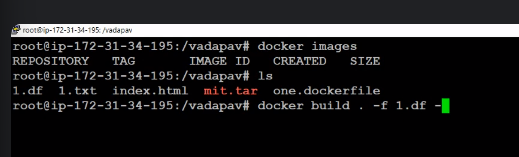
images are created by the docker file



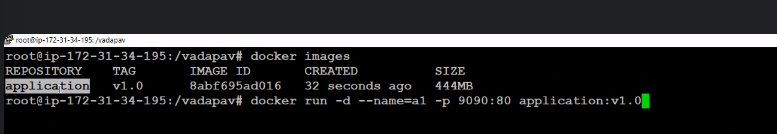
Console:

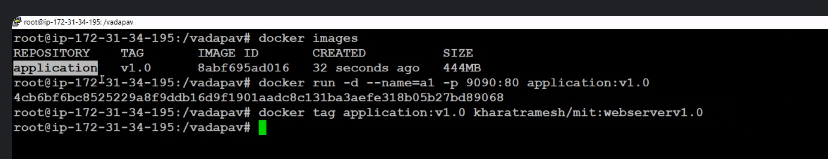


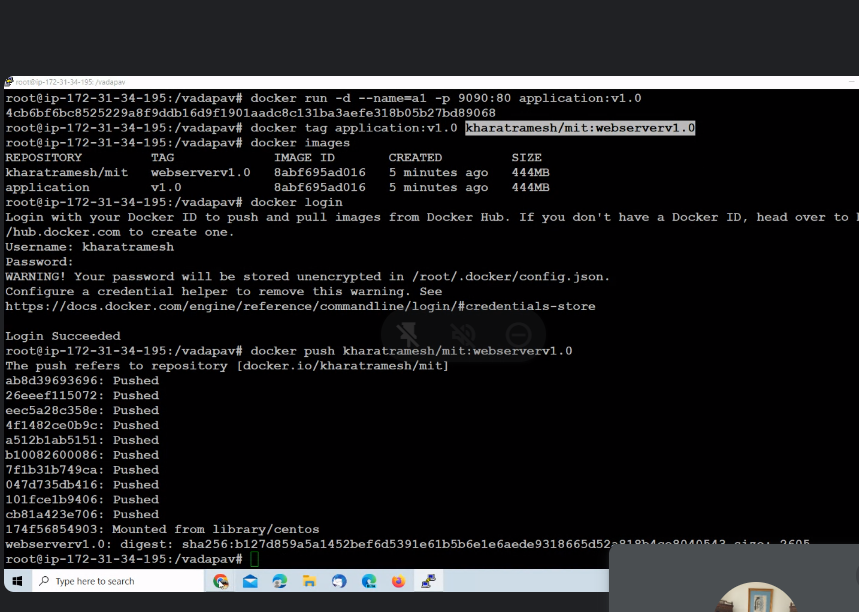


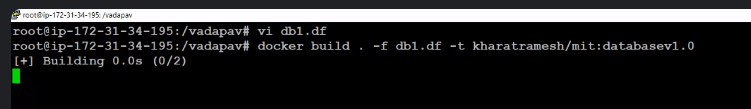








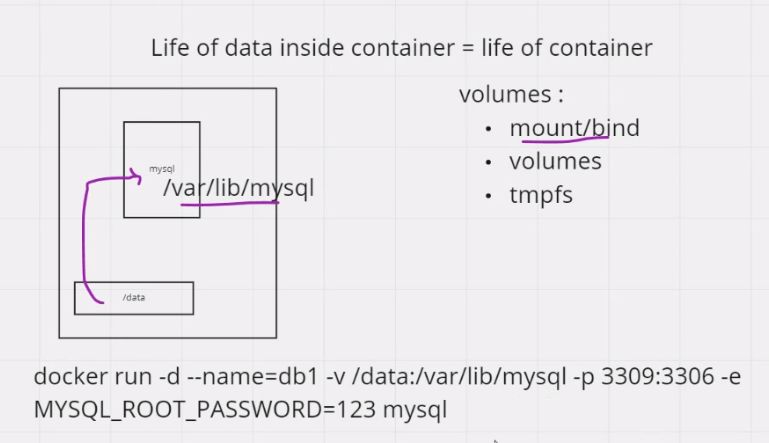


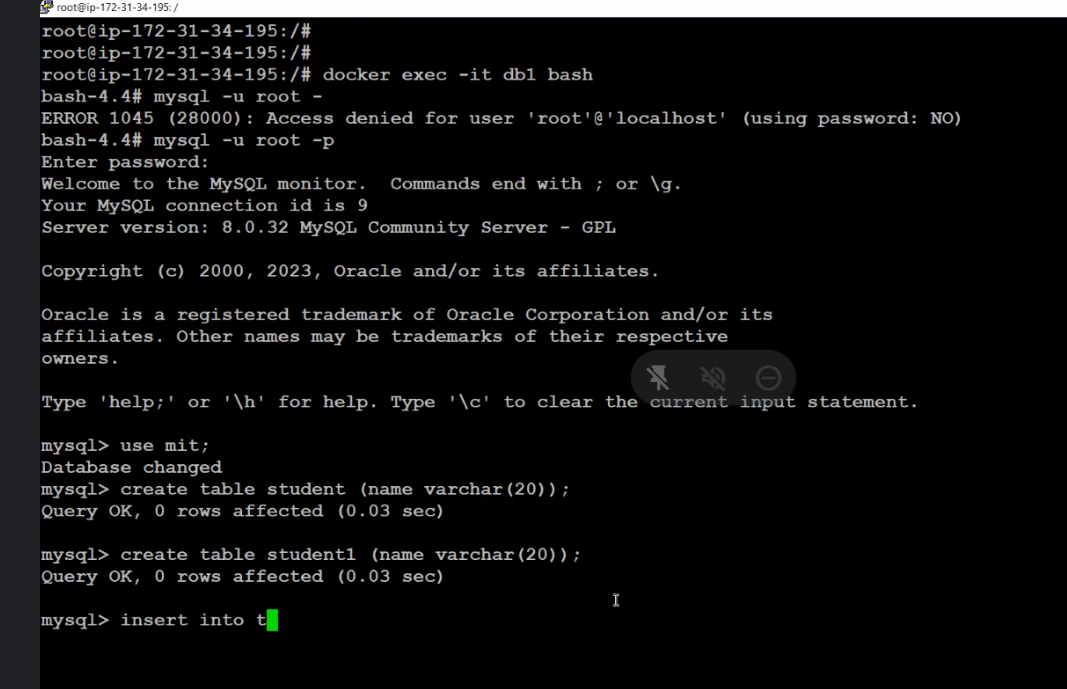


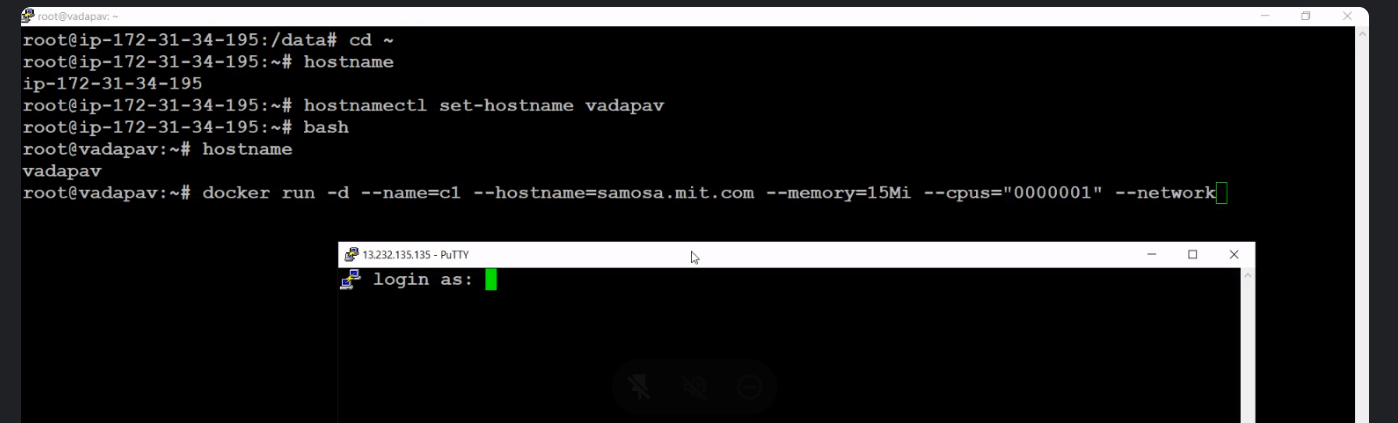
Life of data inside container = life of container

Docker volumes:

* mount/bind
* Volumns
* tmpfs



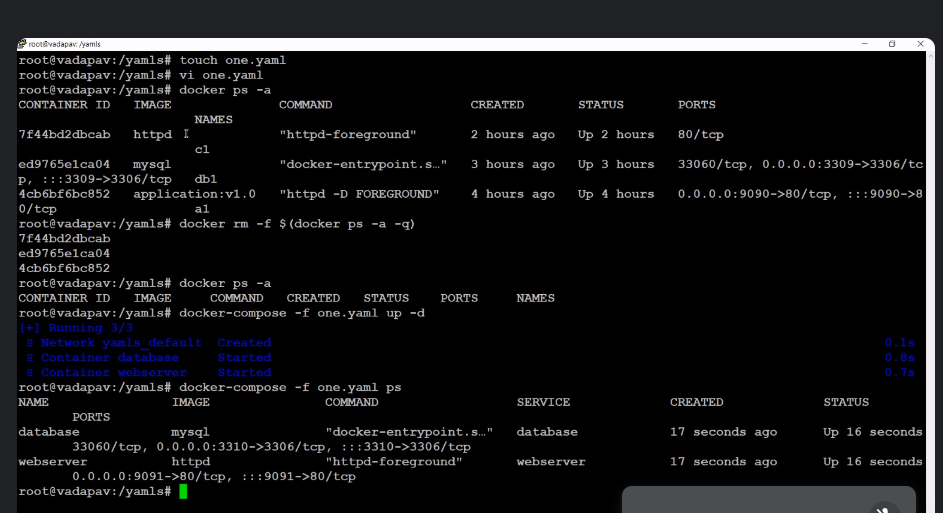


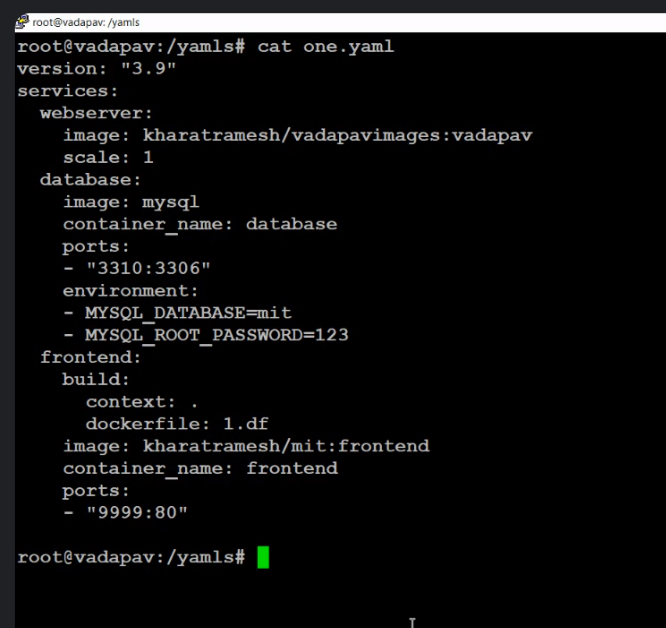




====================================================================







version: "3.9"

services:

webserver:

image: httpd

container\_name: webserver

ports:

- "9096:80"

database:

image: mysql

container\_name: database

Ports: .

- "3316:3306"

environment:

- MYSQL\_DATABASE=mit

- MYSQL\_ROOT\_PASSWORD=123

frontend:

build:

context:

dockerfile: 1.df

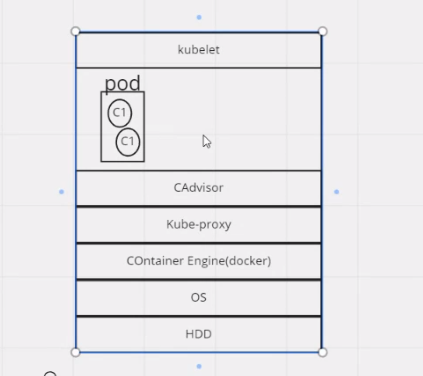
image: aths/project1:frontend

container\_name: frontend

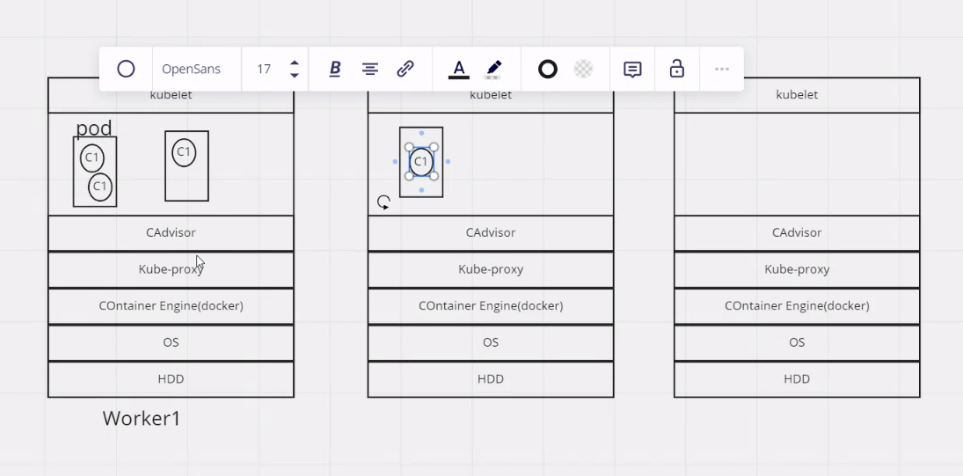
ports:

- "9999:80"





Pod is a logical space assigned by the kubelet to hold one or more container.



Applications runs on worker nodes

Can have 2000 worker nodes in one server/data center

Can have many different set of worker nodes in different environments

To control workers you need a master node.

Master node:-

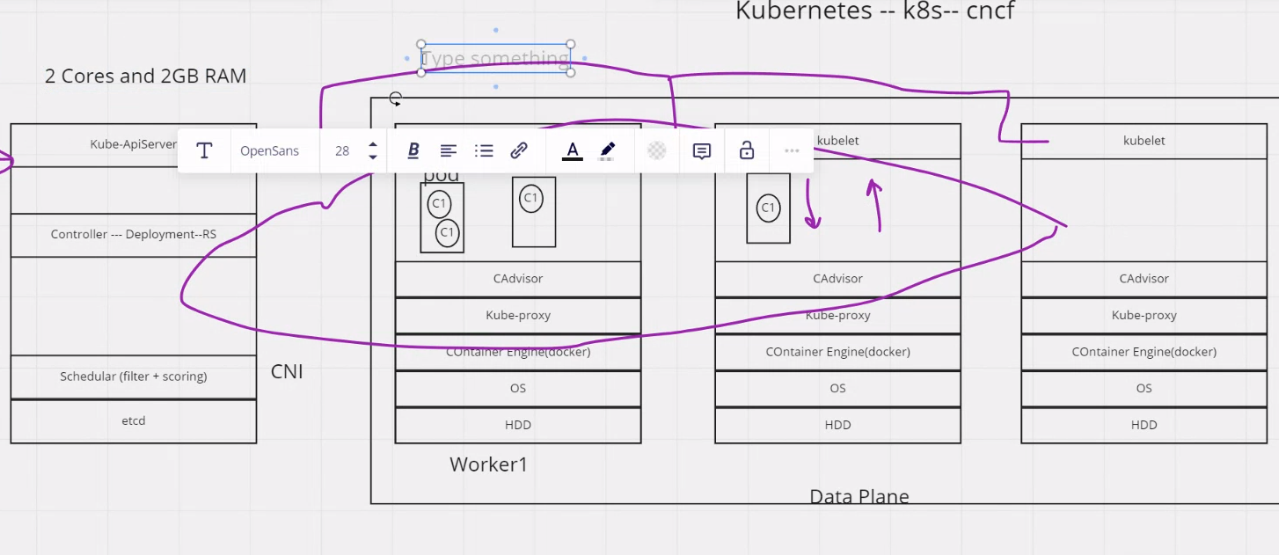
* Kube -APIServer - for communicating
* Etch - it is a database that stores in key and value pairs
* Schedular (filter +scoring) -

Filter - finds the best worker node

Scoring - give a score to each individual worker node

The scheduler gives this information to the kube-API server and kube-API server to

kubelet.



* Controller - Contains object like deployment RS (Replica state)
* CNI(container network interface) - Gives IP address to Pod.

https://kubernetes.io