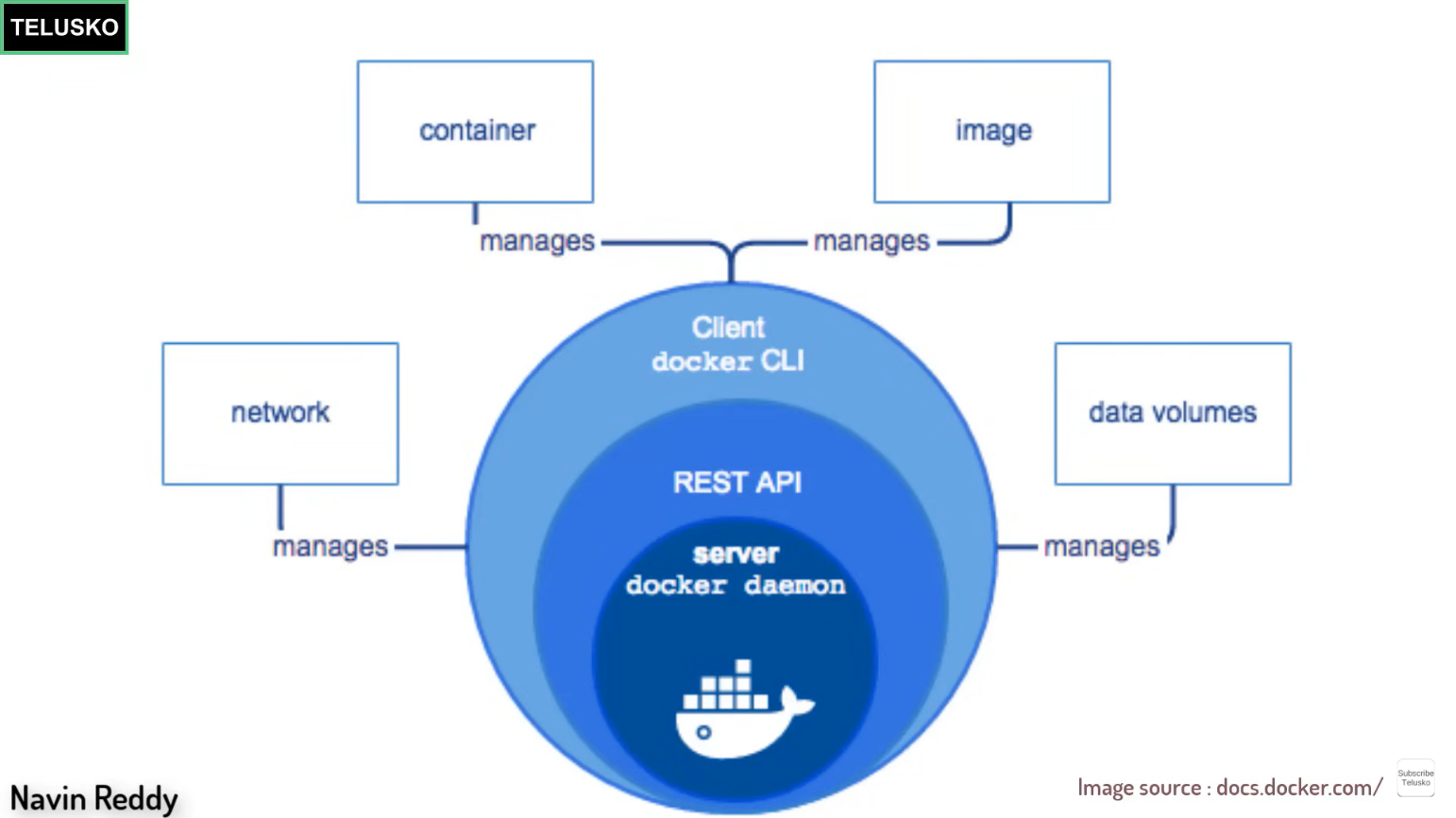
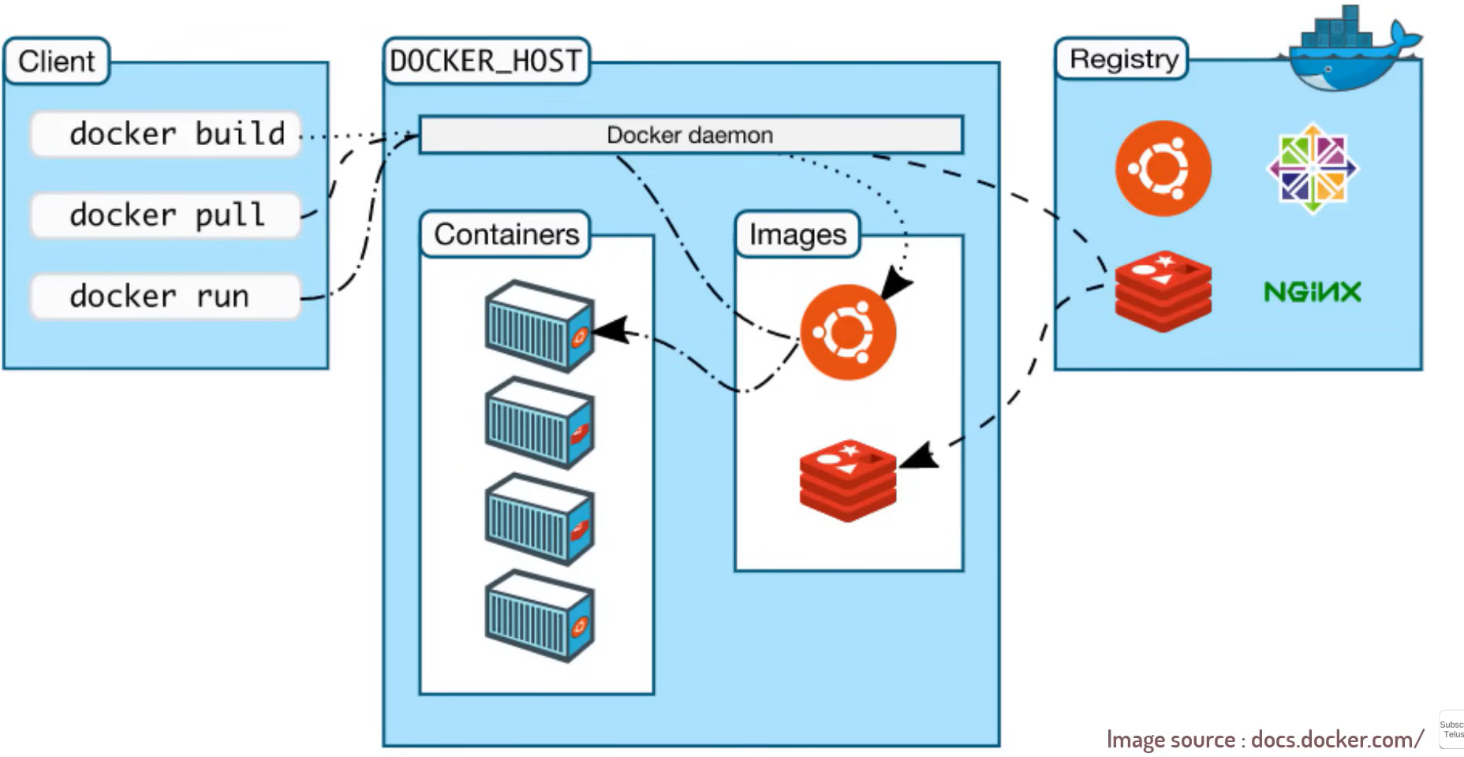
**Docker**

1. Docker is an ecosystem where we can create containers.



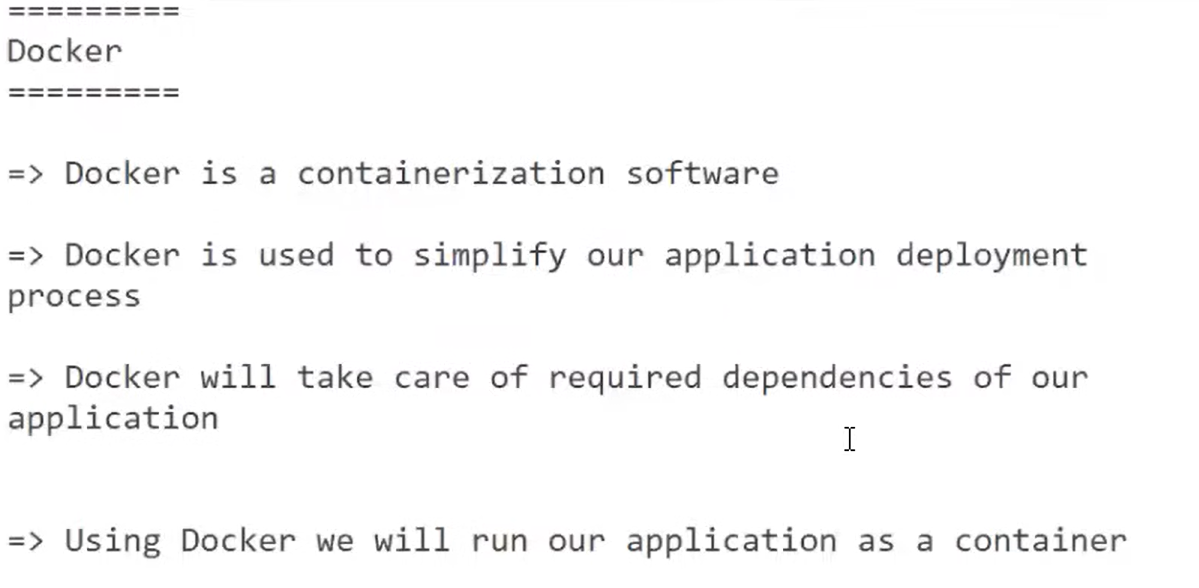
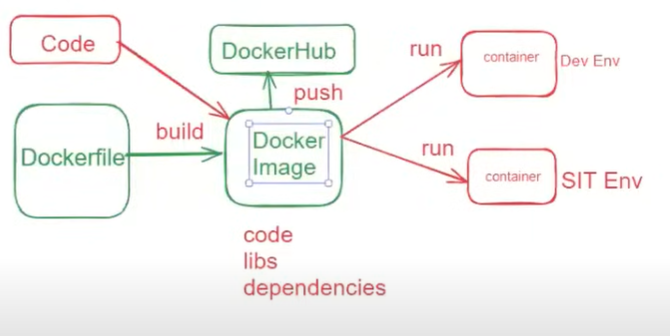
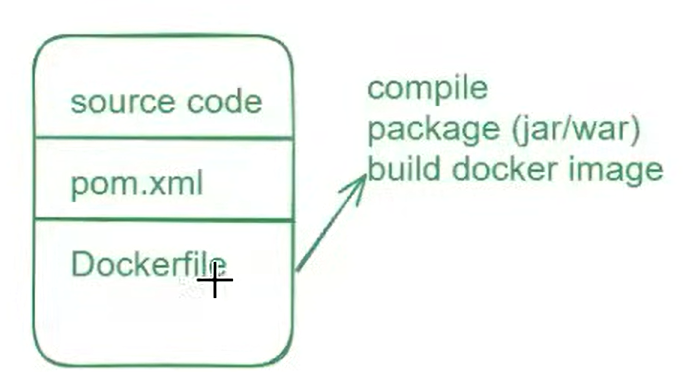
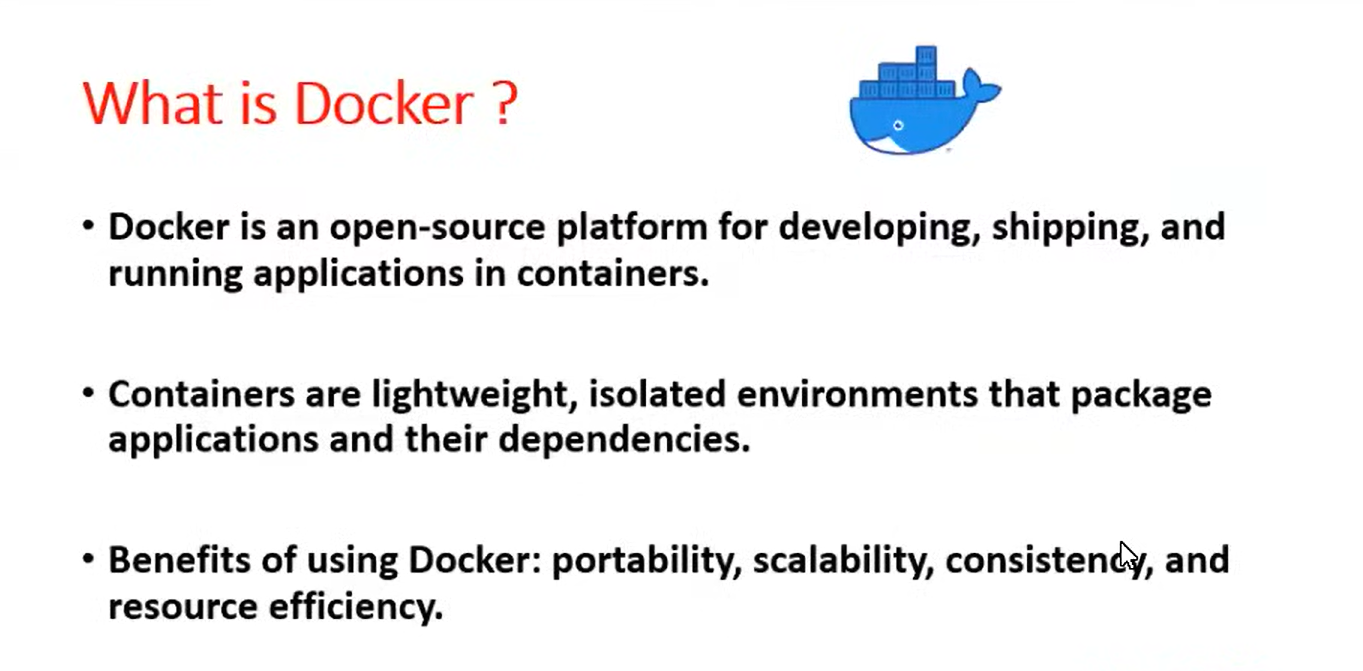
1. Initially docker was made as monolithic application.
2. Now docker is a microservices application.
3. Docker-cli – it takes request from user and send the request to docker server i.e. docker daemon.
4. Client connects with the server by help of rest-api.
5. Container can run the application and store the data.
6. Images are the image of applications to run on the container.



1. Docker daemon accepts the commands from client and perform the operation.
2. It look out for the container and manages the images.



1. Containerd – it is responsible to manage the life cycle of containers.
2. Runc – it simply creates a container.
3. To find the images , there are some registries such as docker hub etc.
4. So when you use the container you can store some data. When you stop the container the data is present. But when you destroy the container the data is lost.
5. So we can use volumes to store data.



**Install docker on Amazon Linux VM**

sudo yum update -y

sudo yum install docker -y

sudo service docker start

sudo usermod -aG docker ec2-user

exit

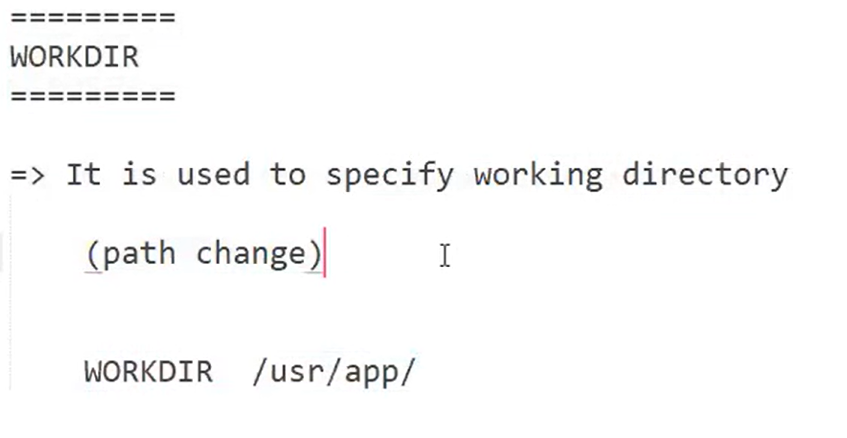
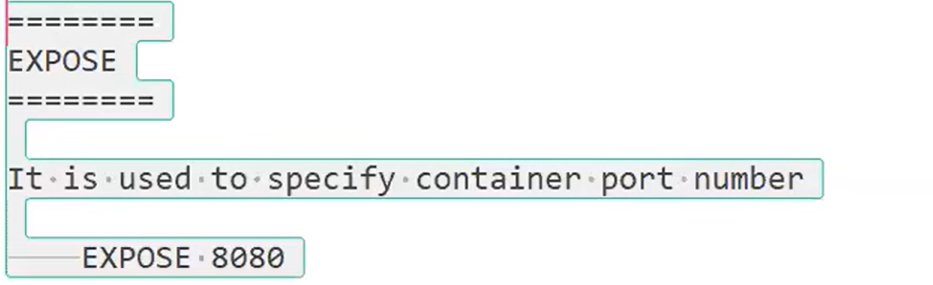
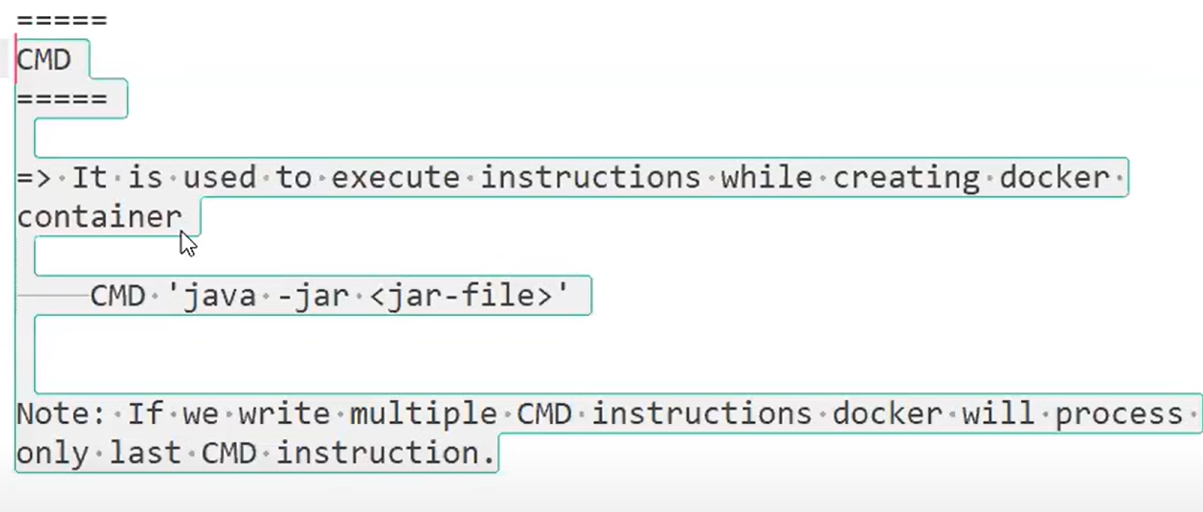
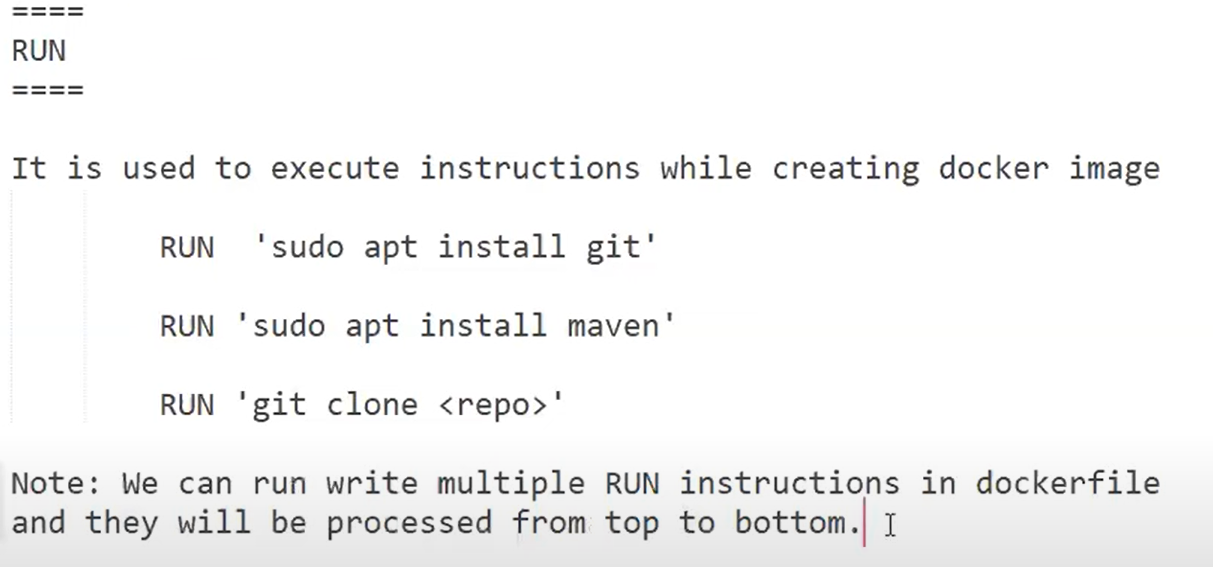
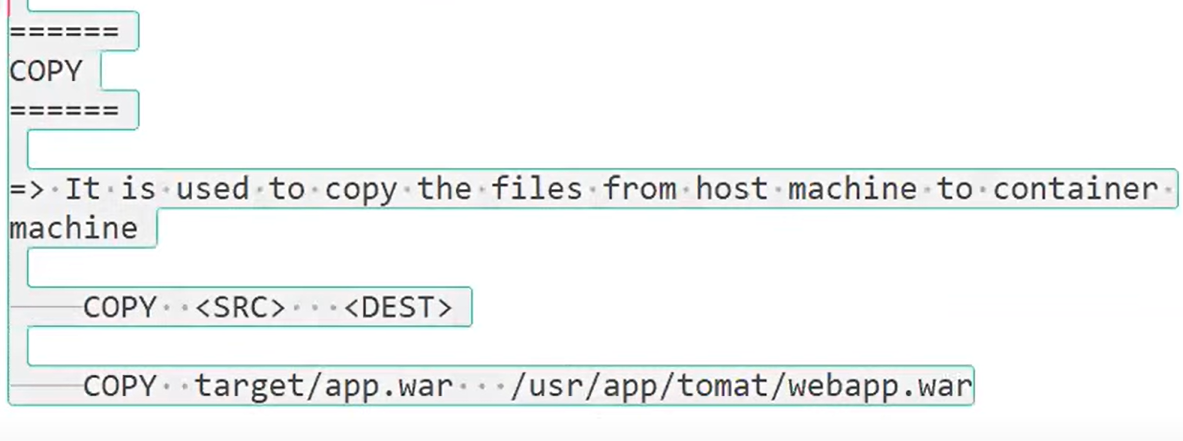
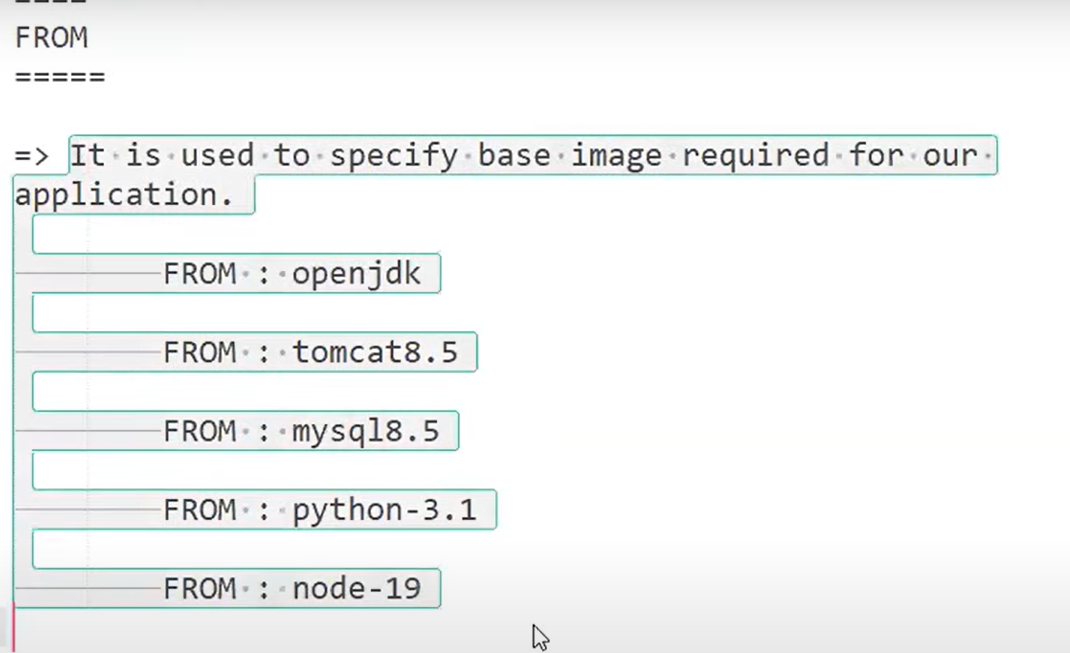
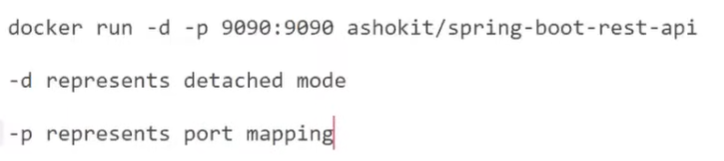
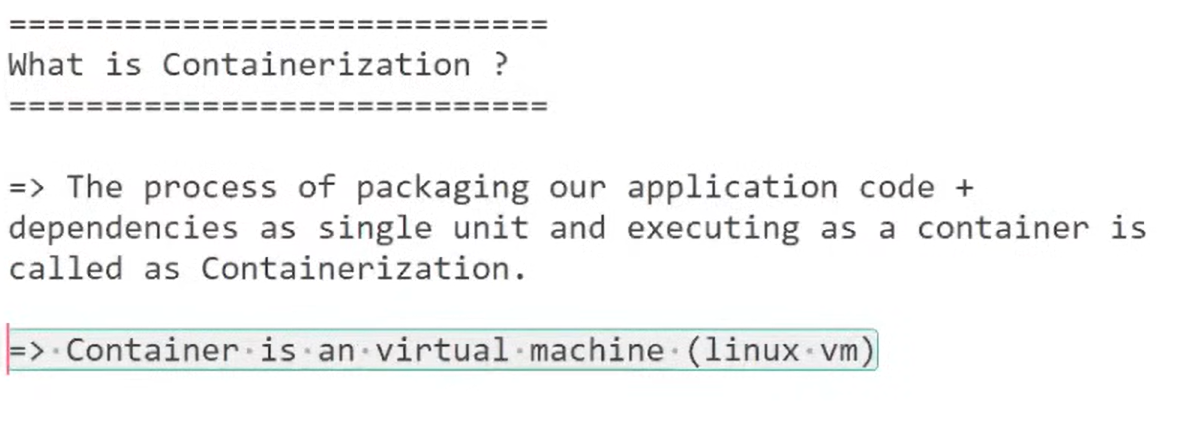
**Install docker on Ubantu Linux VM**

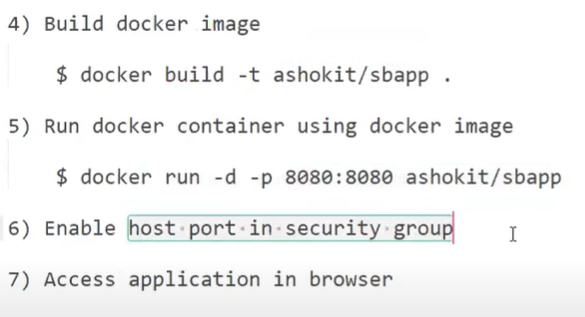
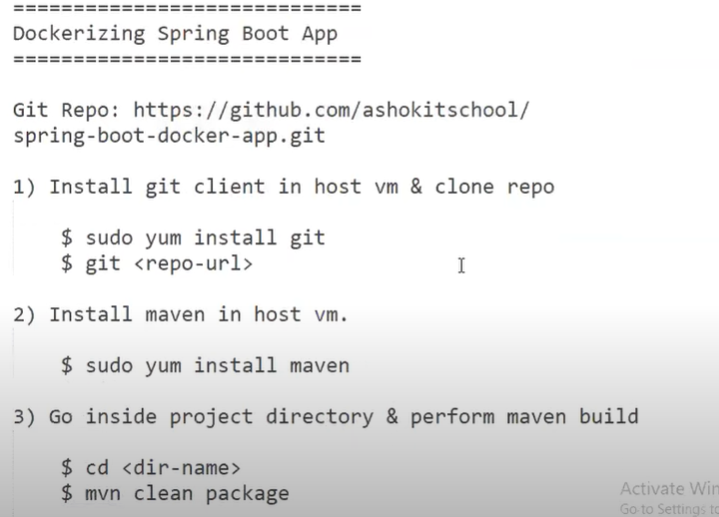
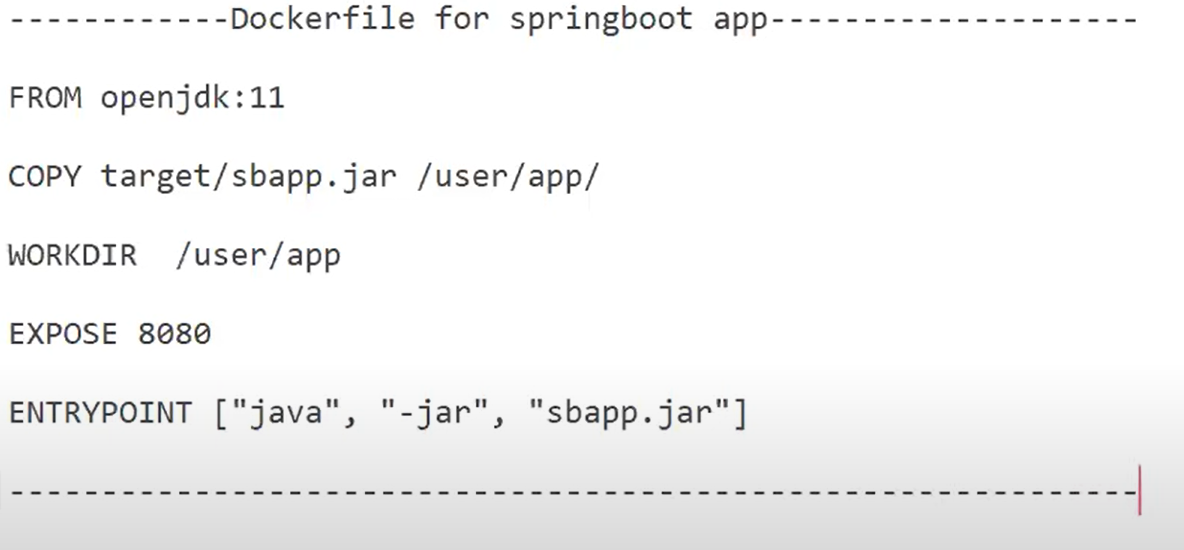
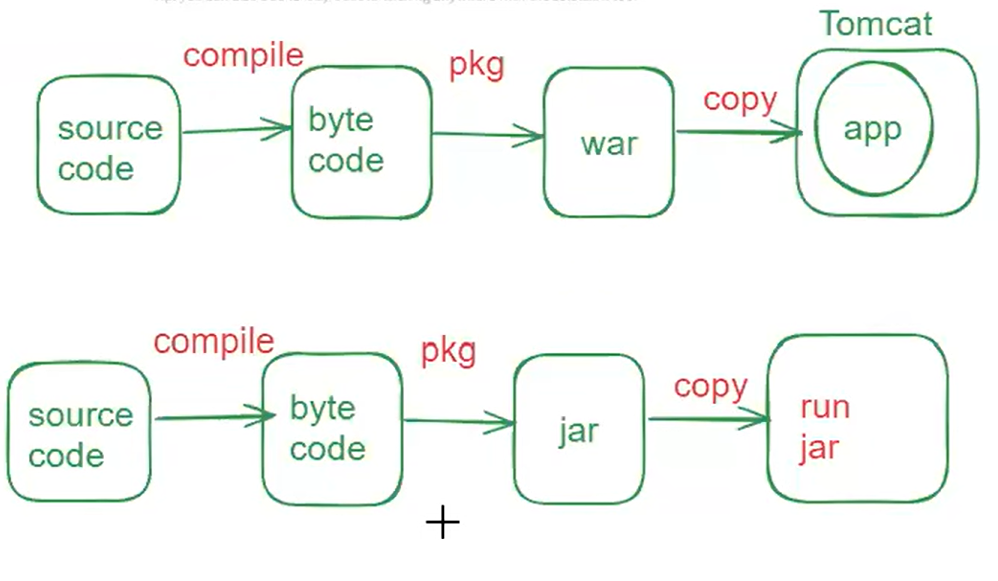
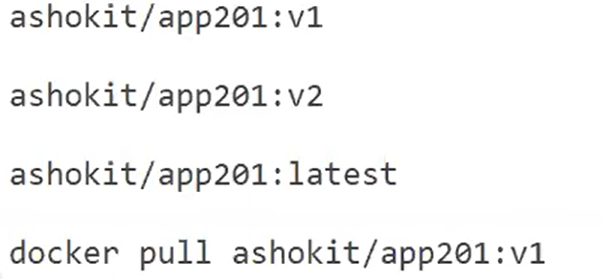
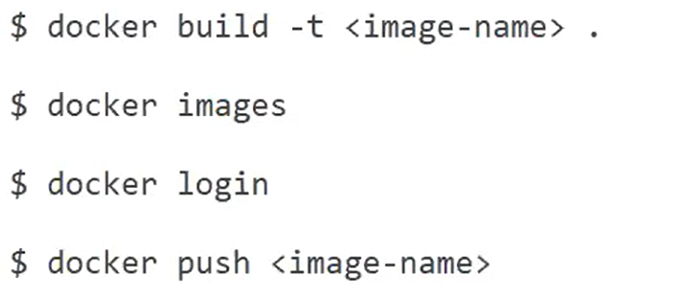
sudo apt update

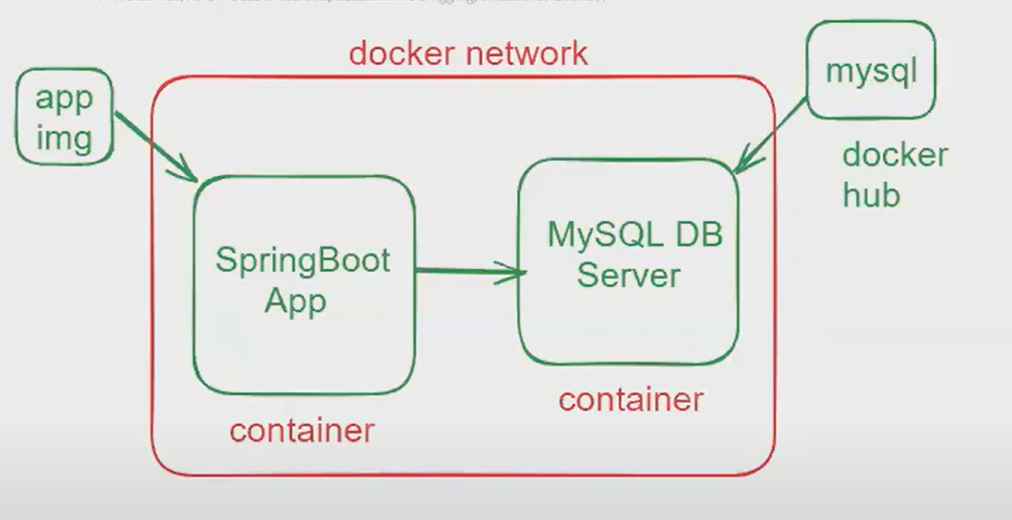
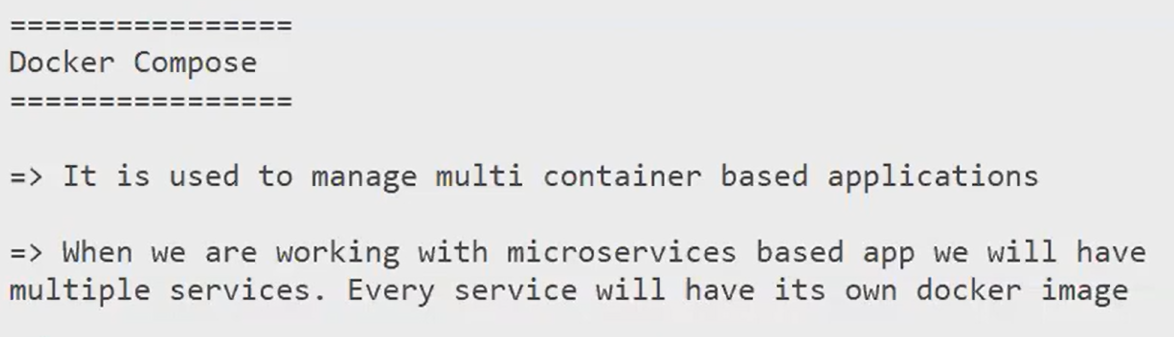
curl -fsSL get.docker.com | /bin/bash

sudo usermod -aG docker ubuntu

exit







**Docker-compose.yml**

version: "3"

services:

application:

image: spring-boot-mysql-app

ports:

- "8080:8080"

networks:

- springboot-db-net

depends\_on:

- mysqldb

volumes:

- /data/springboot-app

mysqldb:

image: mysql:5.7

networks:

- springboot-db-net

environment:

- MYSQL\_ROOT\_PASSWORD=root

- MYSQL\_DATABASE=sbms

volumes:

- /data/mysql

networks:

springboot-db-net: