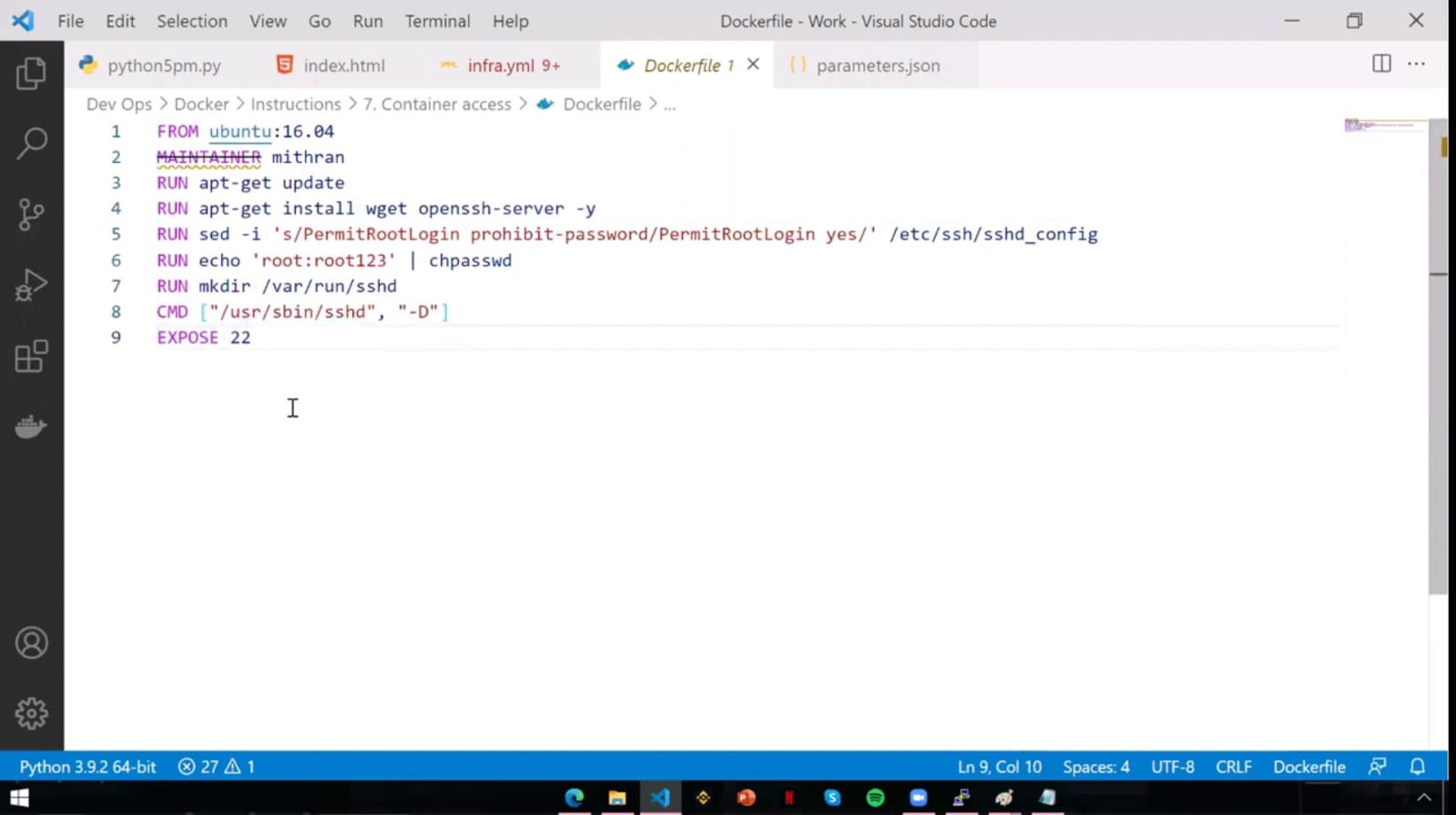
DOCKER



It is used to create a container for giving access to the another user

Expose is used to open the ssh port number

--------------------------------------

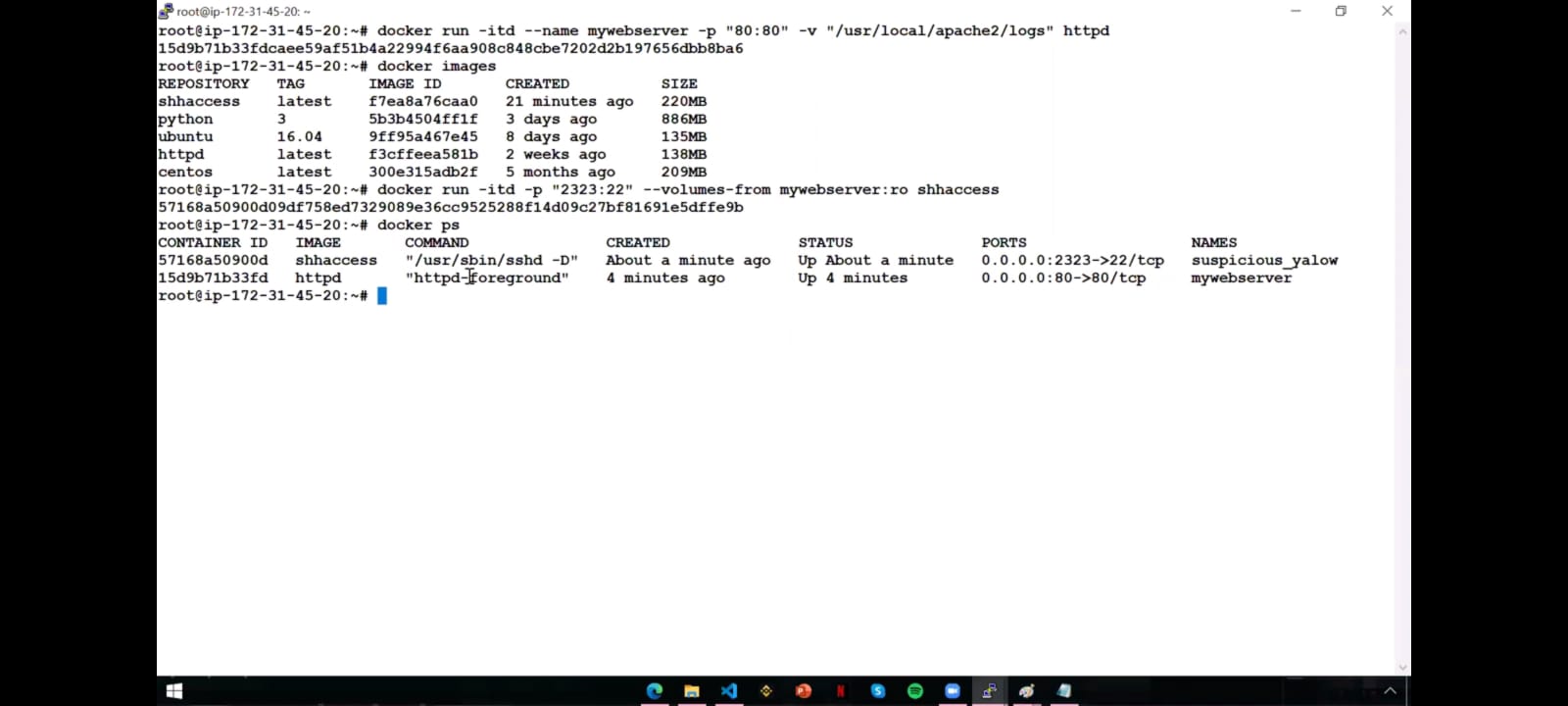
After created a docker file we have to create a image using docker file and create a container using -p (2323:22)portforwarding....

After created an created an container give the public ip and 2323 port number to the user...

Not pem file because in docker creation itself we created password authentication and created an username and password....

Give the username and password to that user...

Now the user can only acces that container...



Scenerio : It is used to give permission to a particular folder on docker container

In first command it will create an container and save the volumes on (var/lib/docker/volumes/<volume\_id>/\_data) rather than host folder

In second command it will use VOLUME-FROM command and it will sync with log files which have created before... And it have only read access

How to user access this container?

Which have we created a image before, he use that same image to give access to the user(shhaccess)

docker run --name mywebserver -p "80:80" -v "/usr/local/apache2/logs" httpd

Creates and starts a new container named mywebserver using the httpd image.

Maps port 80 on the host to port 80 in the container.

Mounts a volume from /usr/local/apache2/logs for logging.

2. docker images

Lists all locally available Docker images with details like repository, tag, image ID, creation time, and size.

3. docker ps

Displays all running containers, showing details like container ID, image used, commands, creation time, status, ports, and container names.

4. docker run -p "2323:22" --volumes-from mywebserver:ro shhaccess

Starts a new container from the shhaccess image.

Maps port 22 in the container to port 2323 on the host.

Shares volumes from the mywebserver container in read-only mode (--volumes-from ...:ro).

So volume-from is used to get the only volumes from previous volume

Is it mount from the already running httpd or it will create new httpd and mount from new container?

Does not mount a volume to an already running container.

Instead, it creates a new container from the httpd image and mounts the specified volume during the creation of this container.

----------------------------------------------------------------------------

Docker run mysql

When we run this command it will download but can't run...

We don't know why..

How to see?

Docker logs mysql

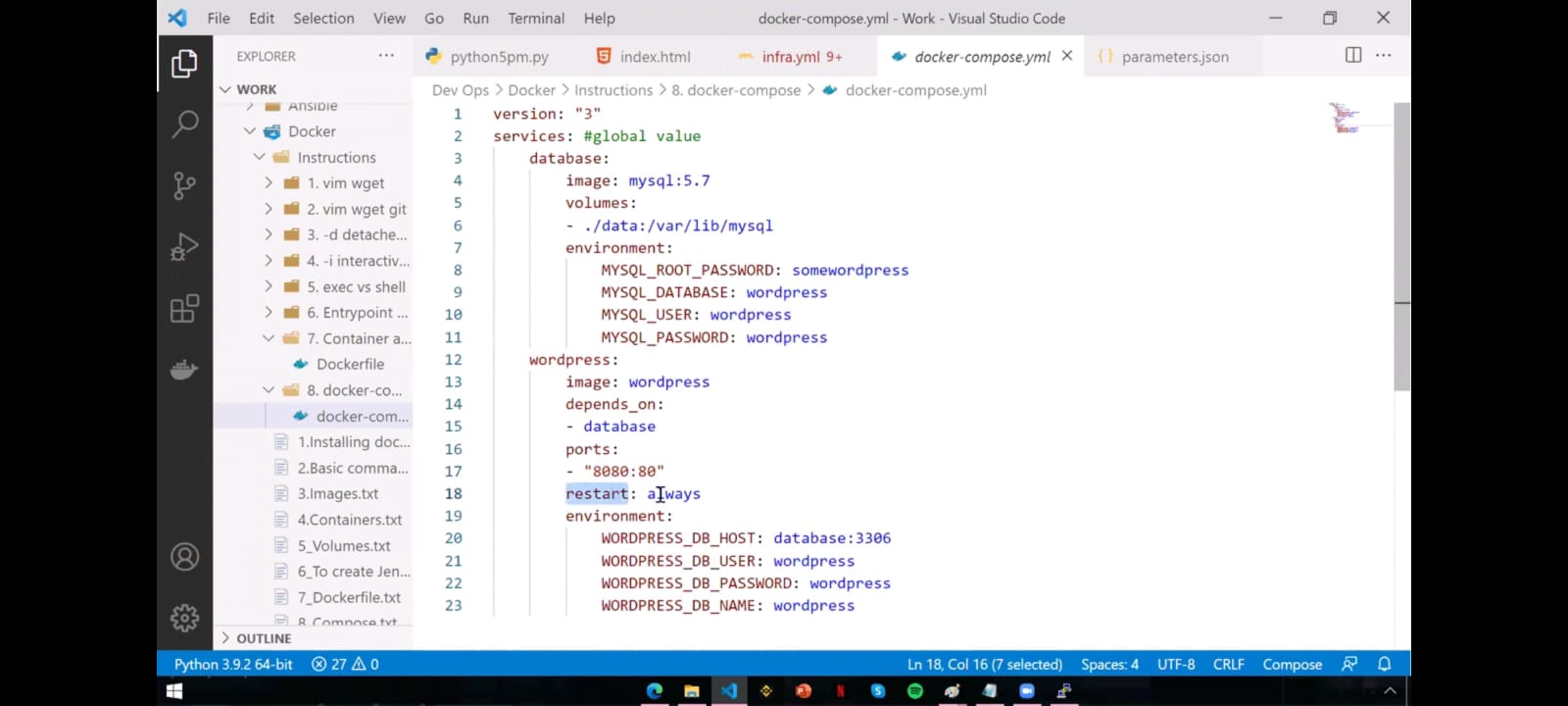
It shows to set the password for db..

To set the pw,

docker run -itd -e MYSQL\_ROOT\_PASSWORD = Nith123 mysql

When we run this command it will run

----------------------------------------------------------------------------



Docker compose yaml file to create mysql and wordpress

Above volume is it get sql data from mysql and save it into /var/lib/mysql

------------------------------------------

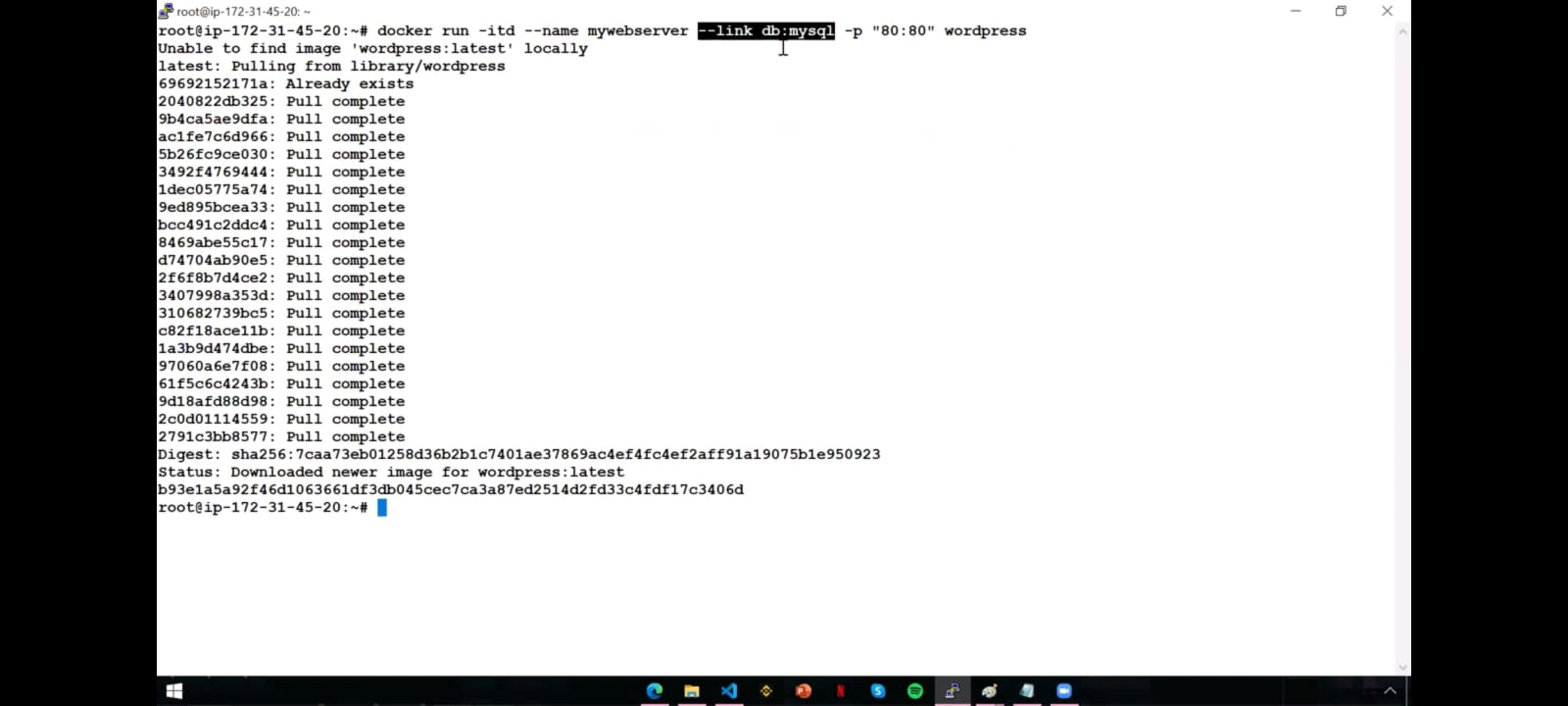
How to save yaml file on our machine?

-Docker-compose.yaml

After saved on our local machine How to run it?

docker-compose up -d

----------------------------------------------------------------------------



Docker link

Docker link is used to link the containers

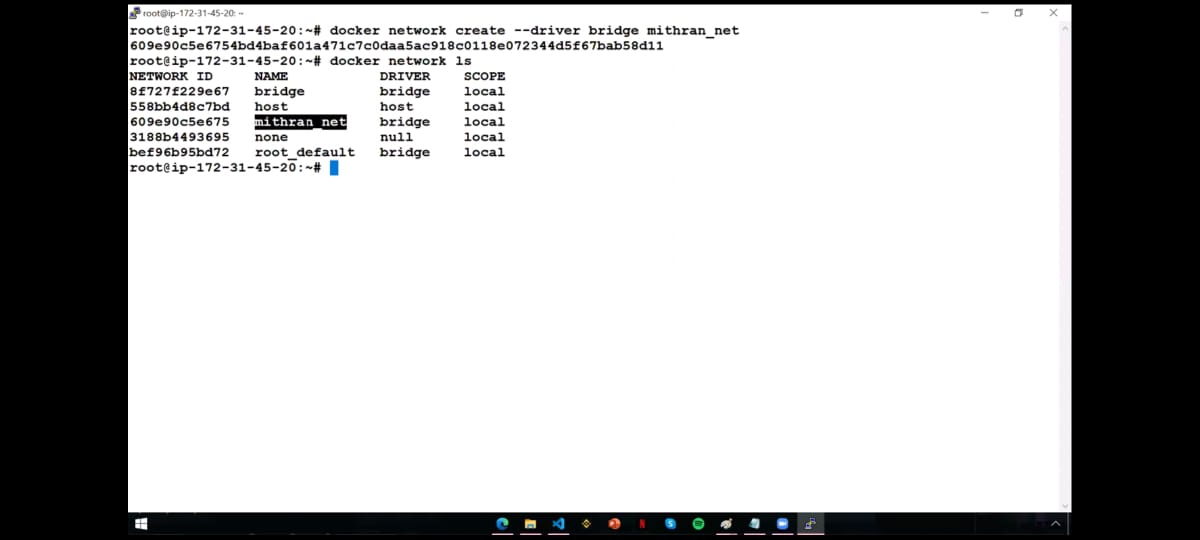
Above scenerio: It will create a WordPress and link with mysql because wordpress need mysql...

--link db --> db is the name what we have give to mysql while creating

mysql --> is the image what we going to use

While link two containers OS is no matter

----------------------------------------------------------------------------



Docker network

Docker network is a just a connection between containers

Three types of docker networks

\* Bridge --> It is a connection we used regularly(Isolated containers)

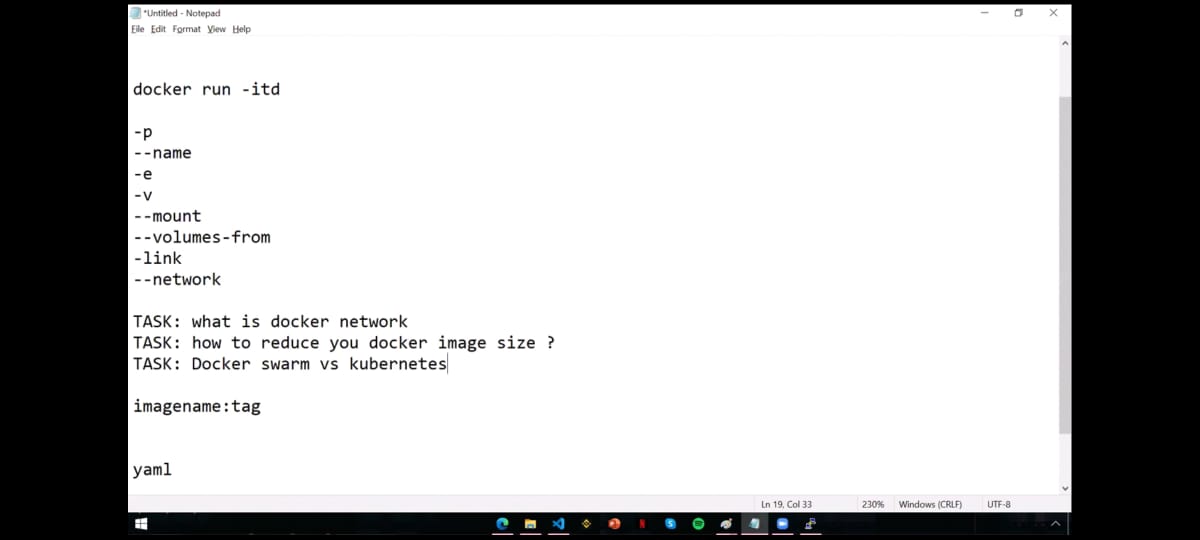
\* Hosts-->

\* None --> It is we not going to use any type of network

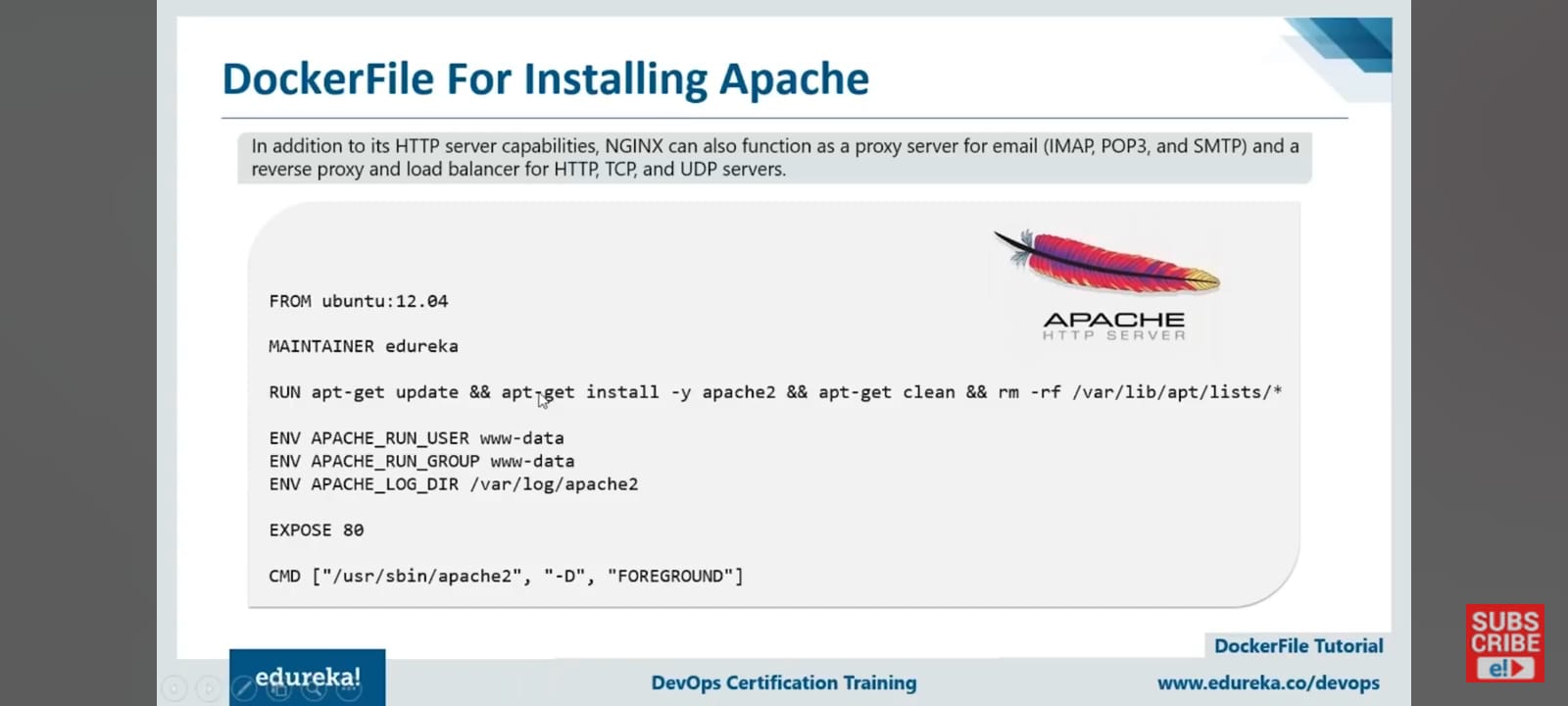
How to create new network?

That is shown above and these three are default we cant able to change

----------------------------------------------------------------------------



TASKs⚠⚠



In Dockerfile,

CMD ["catalina.sh", "run"]

\* catalina.sh: This is the script provided by Tomcat to control the server.

\* run: This argument runs Tomcat in the foreground, keeping the container active.

For Apache web server,

CMD [ "usr/sbin/apache2", "-D", "FOREGROUND" ]

Dockerfile:

Defines how to build a single container image.

Includes setup instructions (e.g., installing dependencies, copying files).

Example: Used to create a custom image for your application.

2. Docker Compose:

Manages multiple containers locally and their relationships.

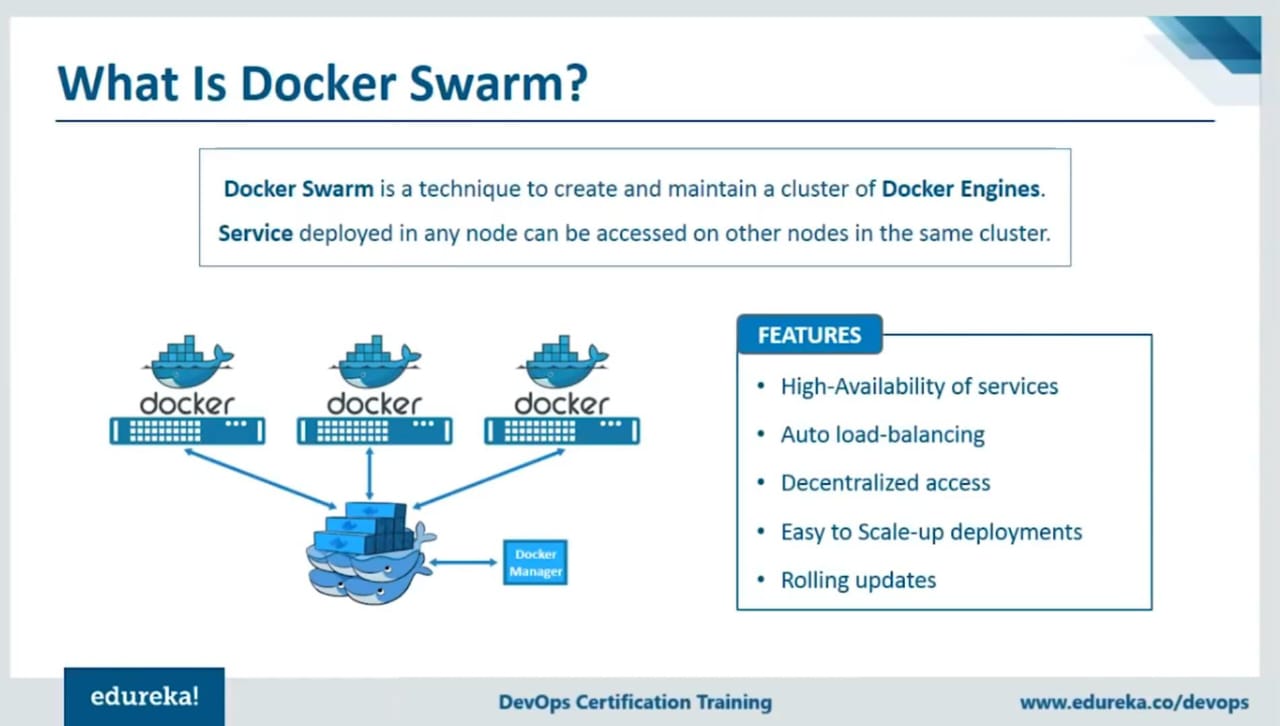
Defined in a docker-compose.yml file.

Example: Runs an app, its database, and other services locally.

3. Docker Swarm:

A tool that helps you run and manage containers on multiple servers working together as a team (called a cluster).

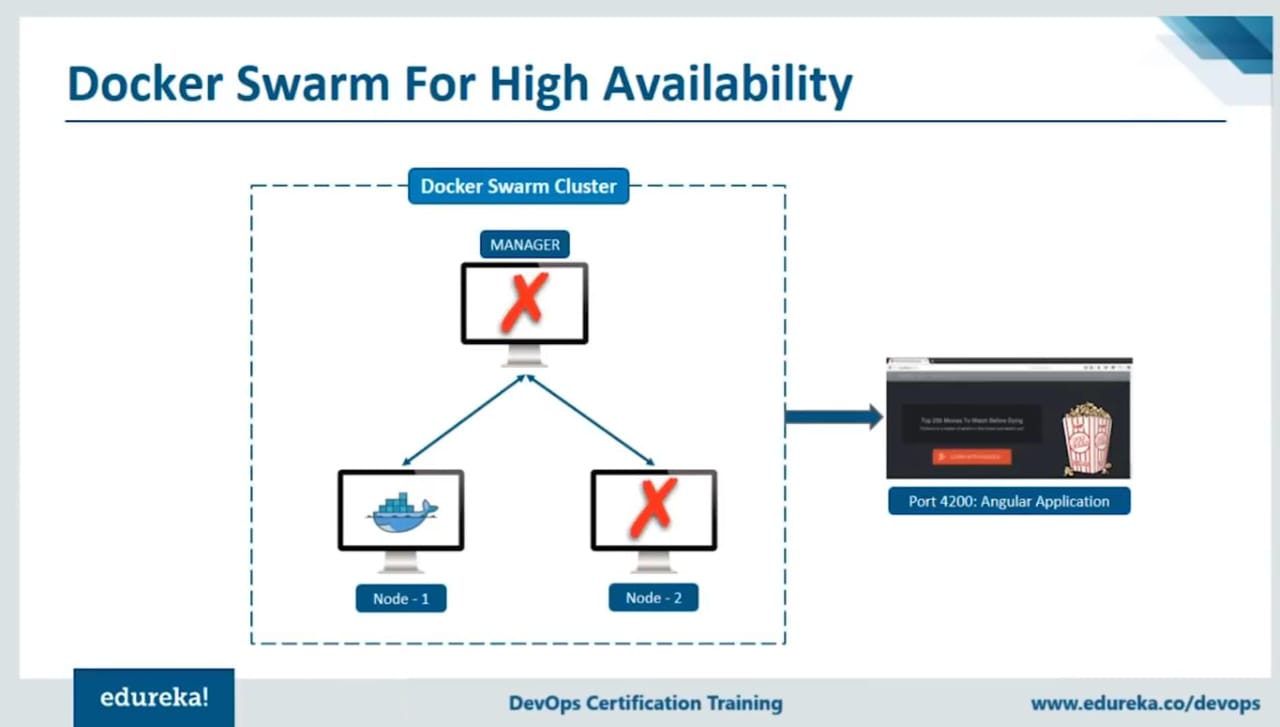
Simple Words: It makes sure your app runs smoothly across many computers, shares the workload, and stays online even if one server fails.



Docker swarm is act like control node and manage node...

In docker swarm, It manages the load balancing when using docker swarm for deploy application and it leads to high availability

If one node us down, the nodes will take over that requests from the user and docker swarm is agentless



In swarm, Even docker control node goes down... The manage down will not off the connection and it still works because that system also have application code and port number will open. So, it works

Docker services

\* docker service create --name "Ang-app" -p 4200:4200 httpd

It is used to create container on the control node

\* docker service create --name "Ang-app" -p 4200:4200 --mode global httpd

It is used to create a container on control node and also manages nodes which have connected to this control node

\* docker service ps --> To see service containers

\* docker service rm httpd --> It will delete container

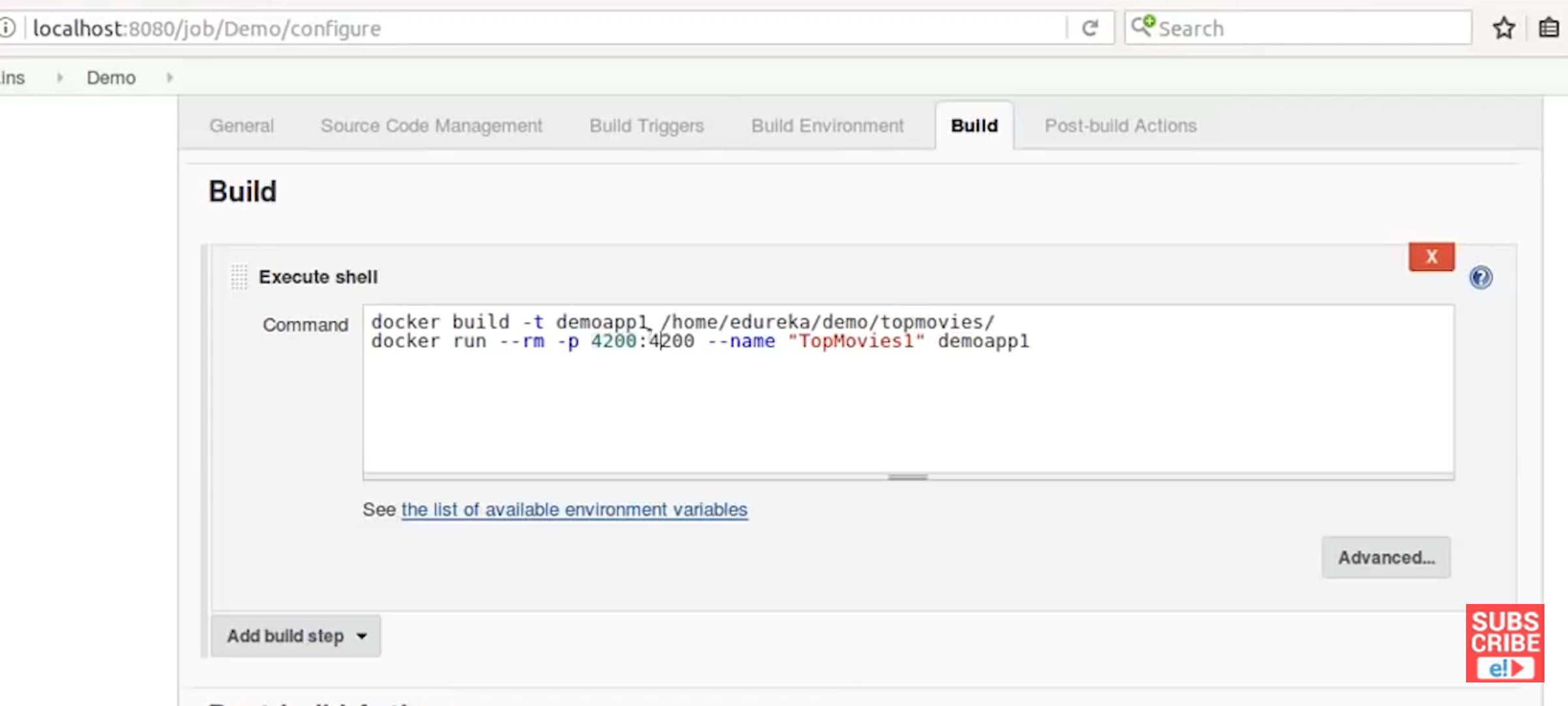
How to deploy to particular num of nodes in docker?

\* docker service create --name "ang-app" -p 4200:4200 --replicas 2 httpd

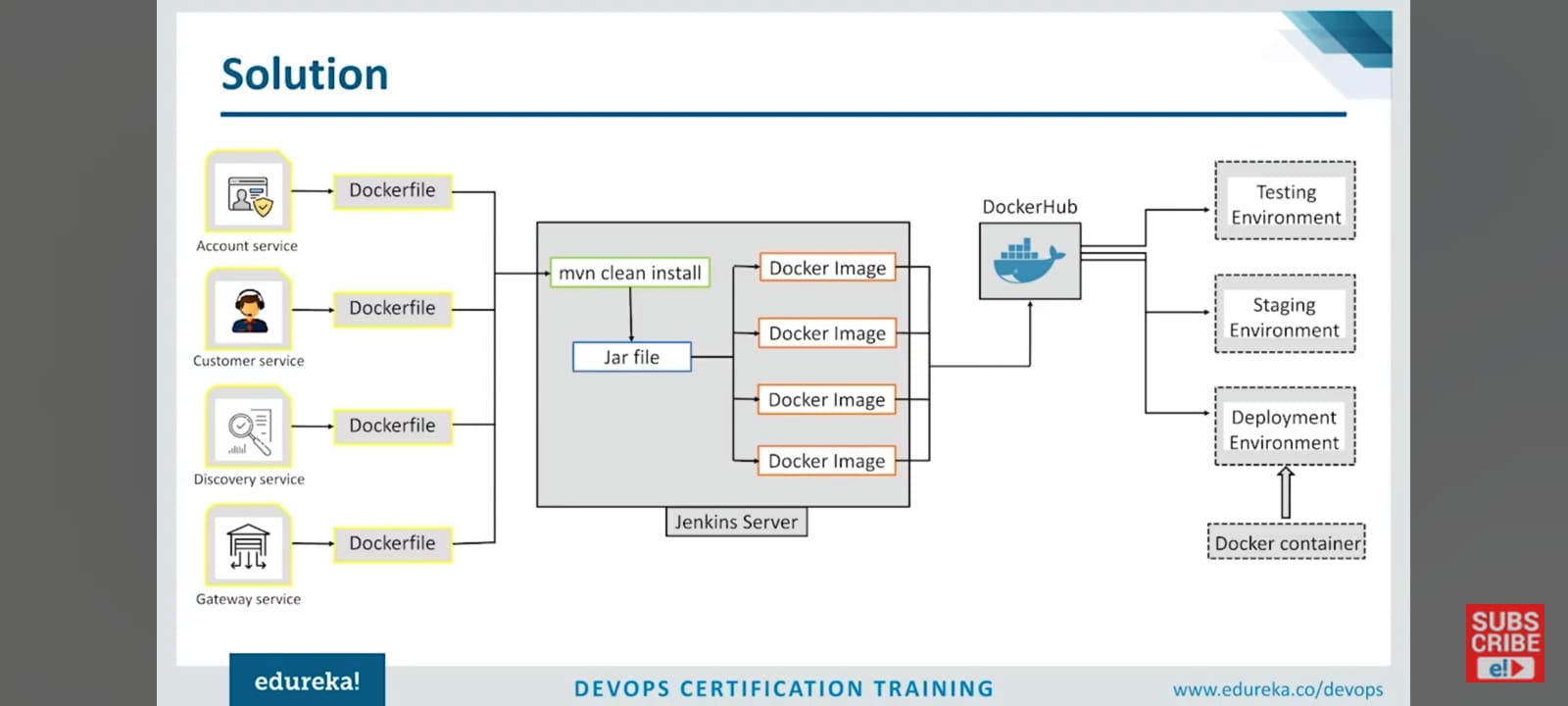
It will deploy on only 2 nodes even there are are multiple nodes avilable

\* docker service scale ang-app(name)=5

It will scale upto 5 containers in connected nodes by creating multiple containers if there is only 2 node available



Real time scenerio: When using docker in jenkins, we have to write everything on dockerfile and put on github... Only docker image build and run container command only have to write on Jenkins execute shell



It is the flowchar for deploy microservices into docker containers using jenkins

How to do it?

In linux, we have to store the codes on that particular folder

(Ex: Product catlog on product catlog folder)

\* use cd command to go on that particular folder(write cd <path/to/folder> on jenkins execute shell)

\* use mvn clean install command to convert into jar file

\* use docker build command to create jar file into docker images(docker build -t <Imagename/Tag> </path/to/that file>(use path or simply use .) )

\* push to docker hub(if needed)

This is how it works.

docker run --rm httpd cat /etc/os-release

To check which os can be used,

For every image in dockerfile when using from command it takes linux is the default os(eg: jenkins, maven, httpd)

stage('Build Account-Service') {

steps {

dir('account-service') {

sh 'mvn clean install'

}

}

}

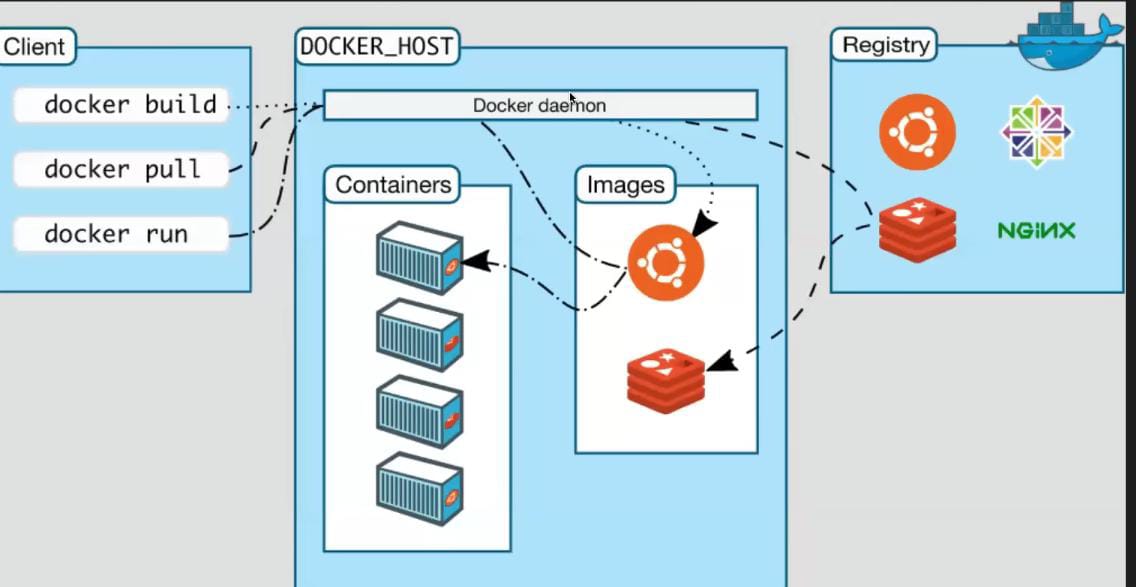
Here, dir('account-service') line is used for it acts like cd command in shell script.so when we want to change directory through Jenkins use dir dor change directory

ADD target/petclinic.war petclinic.war

In this command,

First target/petclinic.war file will be on local machine

Second petclinic.war file will be saved on inside container



Docker architecture

Docker Daemon--> is a intermediate between docker commands and docker interior

It is get a commands from user and apply on docker hub or inside images to build a container

-------------------------------------

# First stage: Build Java application

FROM maven:3.8.5-openjdk-17 AS build

WORKDIR /app

COPY . .

# Build the Java application

RUN mvn clean package -DskipTests

# Second stage: Run the Java application

FROM openjdk:17-jdk-slim

WORKDIR /app

# Copy the compiled JAR from the build stage

COPY --from=build /app/target/\*.jar app.jar

# Run the Java application

ENTRYPOINT ["java", "-jar", "app.jar"]

------------------------------------

\* It is the java web application for multi stage build

\* First stage for Development side and it contains all dependencies

\* second stage only contain java environment and app.jar file which have contents for website and it contains less storage because it doesnt have dependencies

Entrypoint command

The ENTRYPOINT command in a Dockerfile is used to specify the main process that should run when a container starts.

Use ENTRYPOINT when you want your container to always execute a specific command, making it act like a dedicated application.

Dockerfile:

FROM ubuntu:latest

ENTRYPOINT ["echo", "Hello from Docker!"]

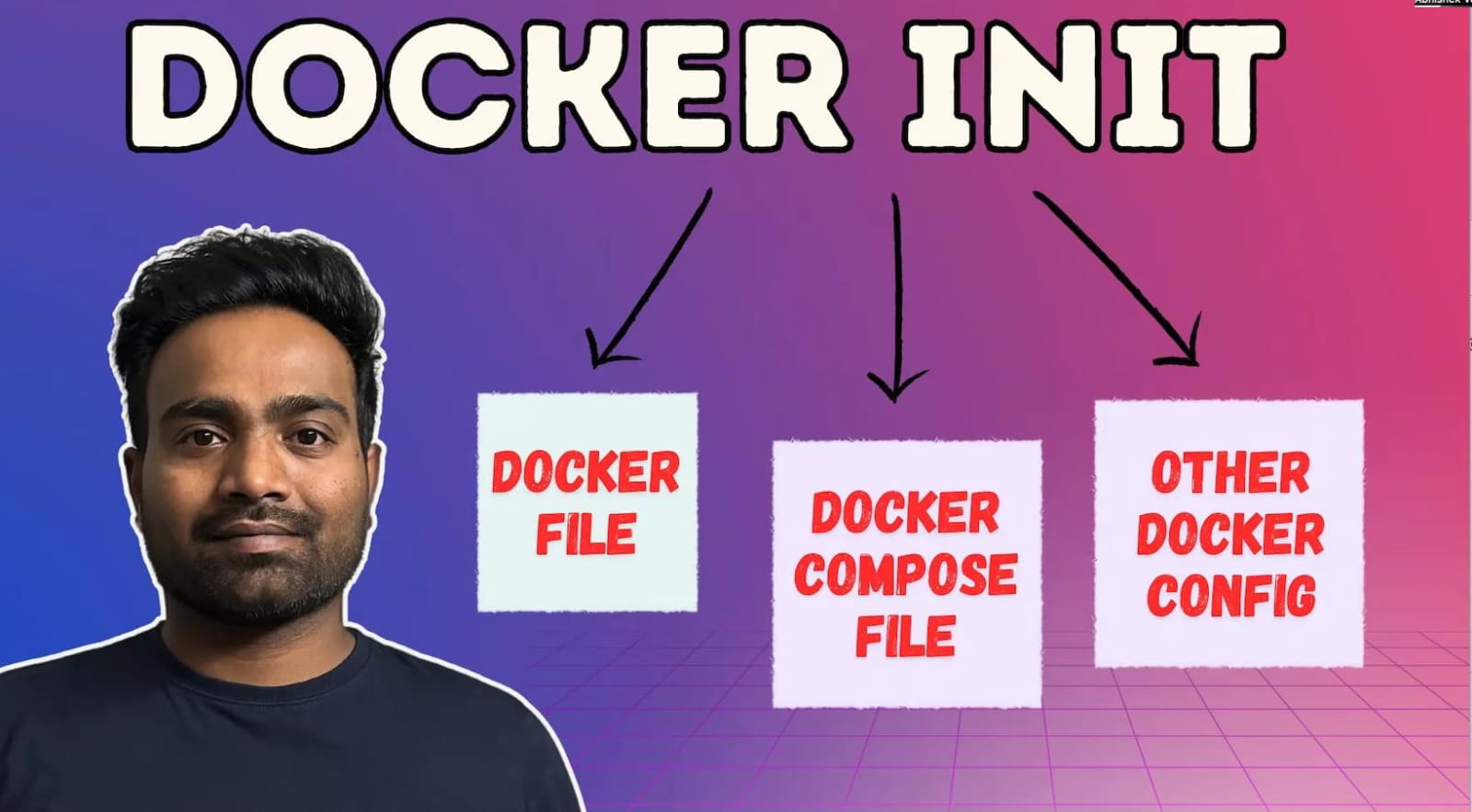
Commands:

docker build -t myimage .

docker run myimage

Output:

Hello from Docker!



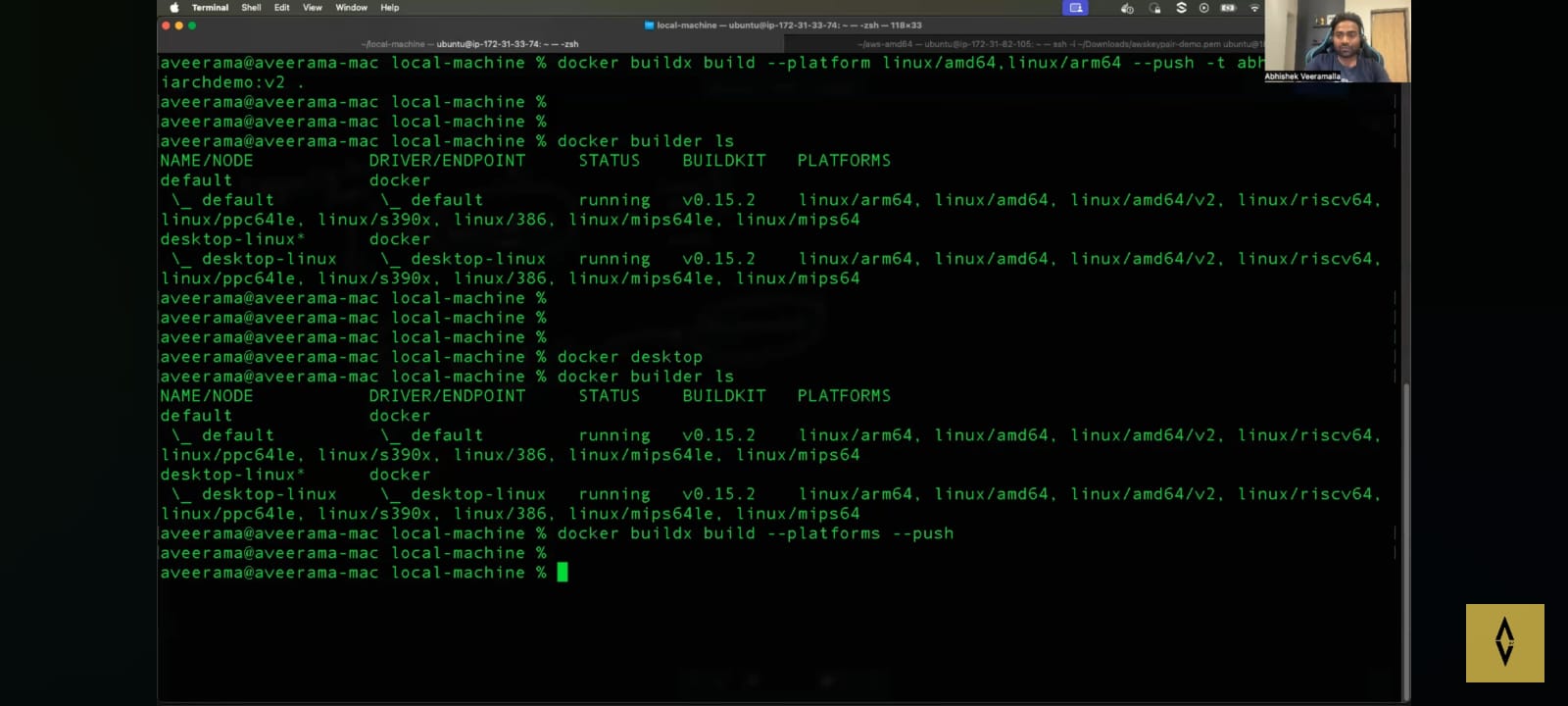
Docker init is a command used to create docketfile, compose file and README.md

How to use it?

Change directory to path where code was and type(Docker init).

It creates a files Automatically with the dependencies

(STILL IT IS IN GROWING STATE, IT GIVES ACCURATE FOR SHORTER LEVEL CODES BUT LARGER CODES WE HAVE TO MAKE QUITE CHANGES)



Docker multiarch platform

Docker multiarch is used for when there is an multiple different processors like(arm64, arm32, X86)..for each processor we want to write multiple dockerfile to create image... To overcome that problem we use docker multiarch platform

Requirements:

\* Docker Desktop (or) Build(if it is ec2)

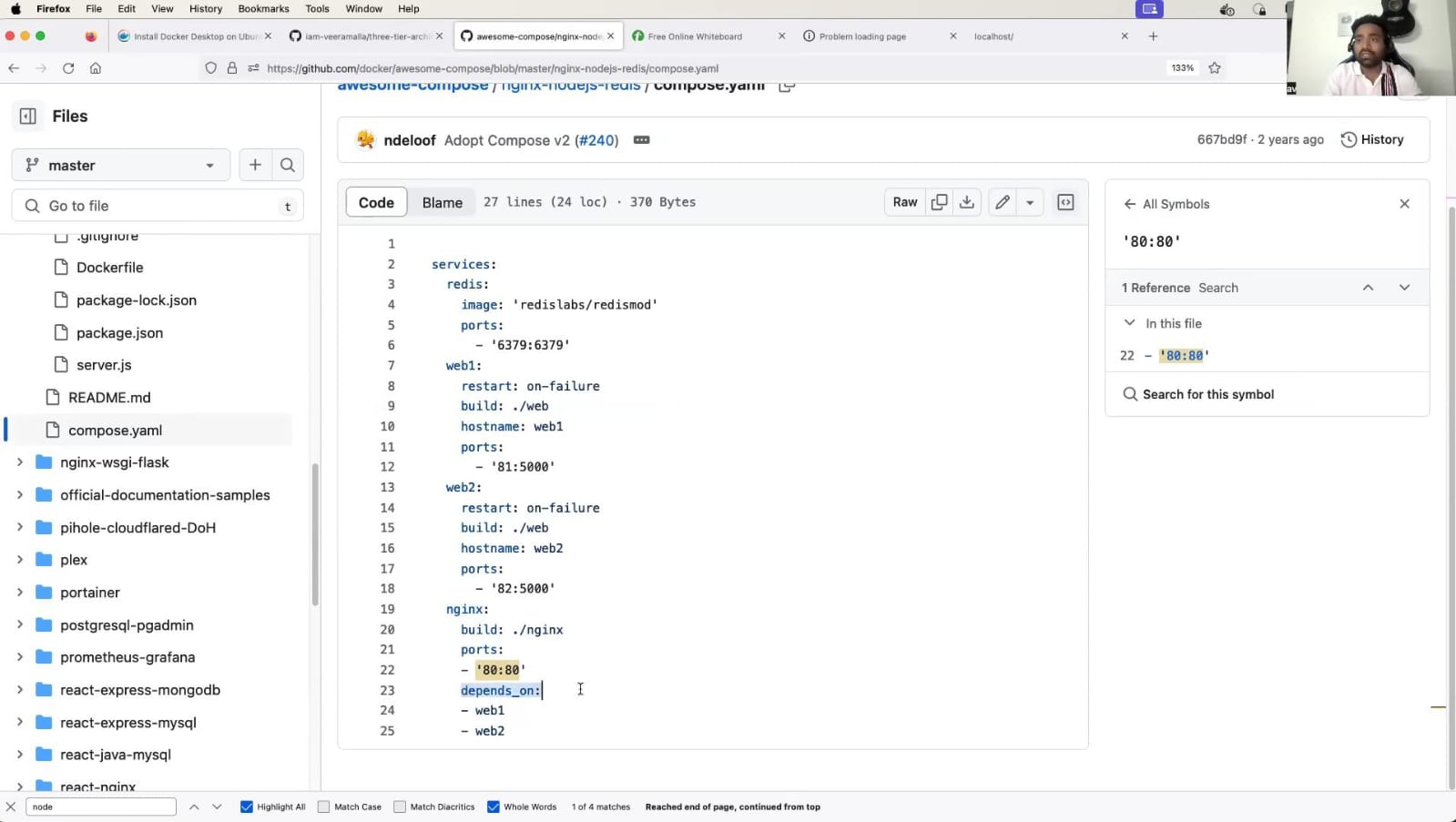
Advantage:

\* We can able to push a docker image include with all processors in a single image name and tag

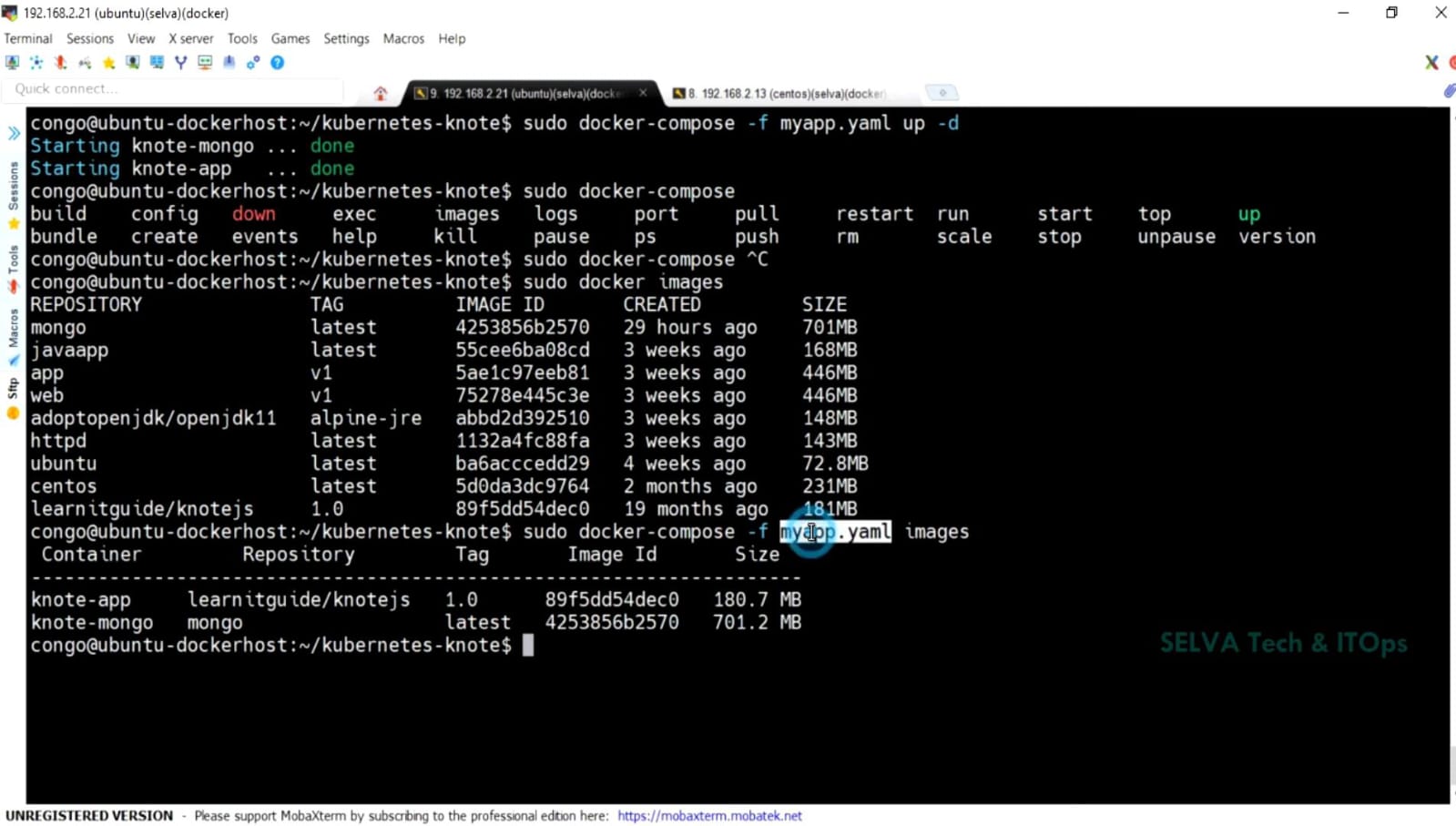
(We will only able to give conditions when we pull to dockerhub)

scratch is the lowest size of image in docker

(https://github.com/docker/awesome-compose)



In service, when we use redis, it is not to create container it is just for use the redis image but not containerize it (like how we use docker multi stage images)



How to see docker compose images?

docker-compose images

How to use other name images instead of docker-compose file?

-f myapp.yaml --> myapp.yaml is a file name and -f is used to use other name files

---------------------------------------------------------------------------------

Docker bake

Docker bake is just like a docker file but wroted on HCL formats