Phase 5

Total classes : 10

Day 8 : 14 Feb 2024

Docker : Docker is an open source software platform which help use to create, manage, distribute and run the application in Docker container. With help of Docker we can package our application with required dependencies and run that application with help of Docker Images.

Docker is known as Advanced Virtualization

If we want to run any application software or tools or database or product we need system software.

One machine with One OS ie window, linux, unix, Mac.

One machine we can install multi os.

VMWare software

Oracle VM etc

In base machine we can install VM ware software and with help of VMWare software we can run different os like window XP, Mac, Unix.

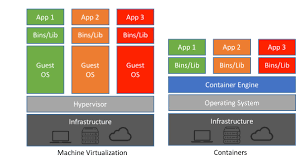
Base machine 16GB RAM 500GB harddisk.

Mac using VM 4 GB 50GB hard disk space need to provide

10 OS. 1gb ram and 10Gb hard disk space.

With help of VM ware software we can do Virtualization. Virtualization help to provide abstract version of OS.

Docker is use to do containerization. Containerization is use to do abstract version of an application.



Container : it is a run time environment.

Docker container : it is a type of container which is responsible to run docker images. Docker container is light weighted container. Docker engine provide run environment for Docker

We can run any application software without system software.

Docker provide internally provide light weighted os in the form of Docker Image. Which is responsible to run the application.

docker --version check the version of docker

docker info to check the information of docker

docker image : Docker image is a ready only template which contains application details with their dependencies which help to run the application with help of containers.

If we want to run any application develop in any language we need to create docker image with help of Docker file. Image contains that application details with their dependencies. So rather than sharing to whole application code you need to share the image with us.

docker images this command is use to find all images present in current machine.

docker pull imageName this command is use to pull the image from Docker hub account. If image present in local machine it will not download else it download by default from Docker hub account.

Docker hub is like a git hub. In git hub we can push or clone and pull any type of files. But in Docker hub is a type of remote repository which help to publish or push or pull private as well as public images.

Using docker hub we can share the images between one team to another team.

hello-world is one the type of public image.

docker run imageName/imageId using this command we can run the image . Once we run the image that application which configured inside that image it up with help of container.

busybox is type of unix utility OS Images.

Hello-world image is use to run the C program ie application software

Busybox is a type of system software images.

docker pull debian

docker run -it debina

docker run -it ubuntu bash

we will take the help of OS image and copy and paste our application like java, python, angular, spring boot with their dependencies and create the image. That image we can run.

We can publish that image in docker hub account other team member can pull that image and they can run that application without installing any required software in their machine

If we want to create docker custom image we need to create file with name as Dockerfile without extension.

1. Creating custom image to display echo message using busybox base image.

Dockerfile

FROM busybox:latest

CMD ["echo","Welcome to Simple Docker Image create by Akash Kale!"]

docker build -t my-busybox . -f Dockerfile

docker images

docker run my-busybox

1. Creating docker image to run the simple Java Program

**Demo.java**

public class Demo {

    public static void main(String[] args) {

        System.out.println("Welcome to Java Program running using Docker Image!");

    }

}

Dockerfile

FROM openjdk:17

COPY Demo.java .

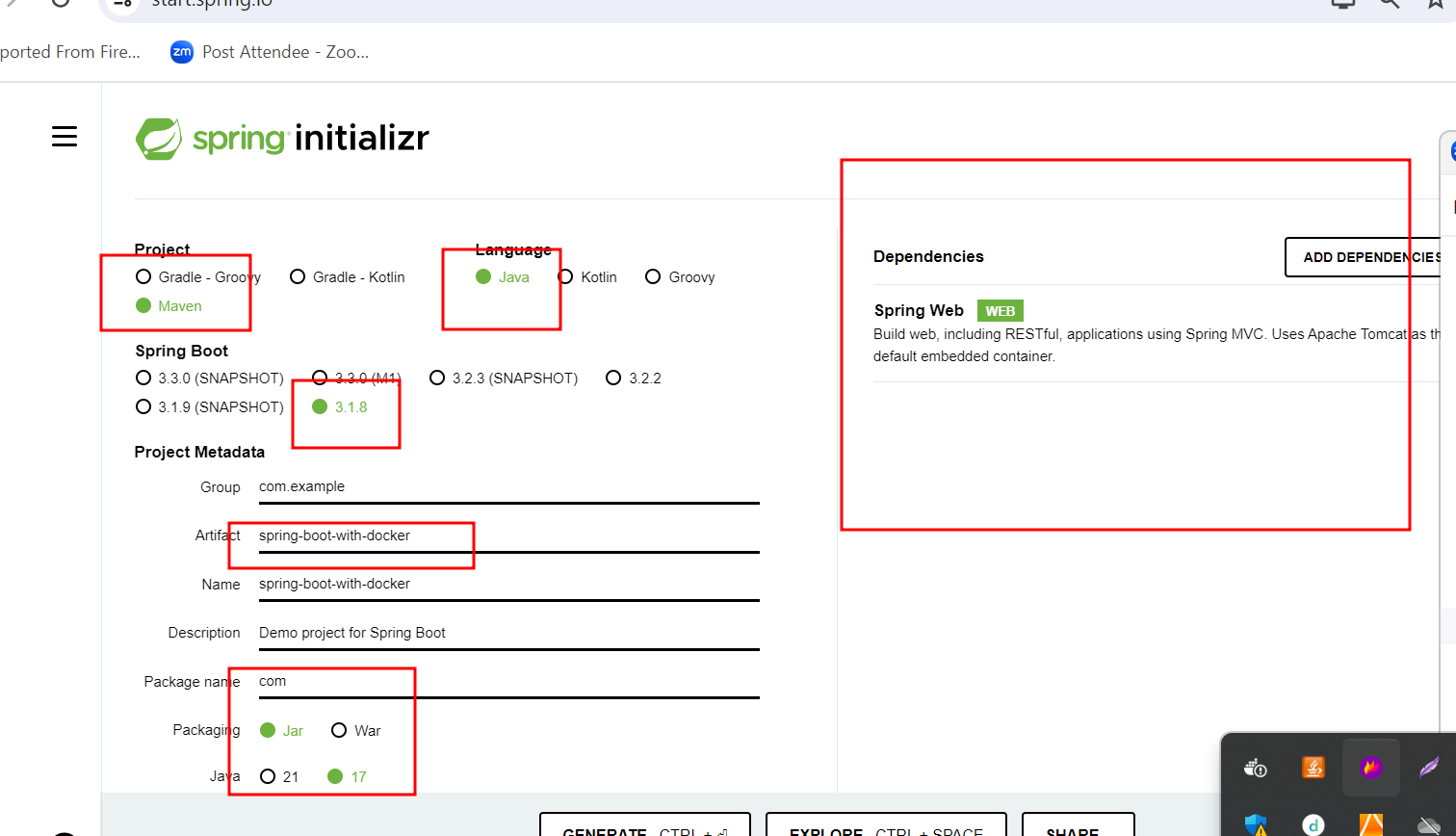
RUN javac Demo.java

CMD [ "java","Demo" ]

docker build -t my-java . -f Dockerfile

docker run my-java

1. Creating docker image to run simple spring boot rest api



Create spring boot rest api

Using Maven tool create jar file

Using mvn package command.

Dockerfile

FROM openjdk:17

COPY /target/spring-boot-with-docker-0.0.1-SNAPSHOT.jar .

CMD ["java","-jar","spring-boot-with-docker-0.0.1-SNAPSHOT.jar"]

To create the image

docker build -t my-spring-boot . -f Dockerfile

run the image (if image is responsible to run web application we need to use below command).

docker run -d -p 9090:9090 my-spring-boot

-d : detached mode

-p : publish port number

Left side expose port number. It can be same or different.

Right side actual port number of application

docker run -d -p 9191:9090 my-spring-boot

docker run -d -p 9292:9090 my-spring-boot

docker run -d -p 9393:9090 my-spring-boot

To check the running containers.

docker container ls : which is use to display all running contains.

Or

docker ps it is use to display only running container.

docker ps -a : it is use to display all container (state can be running or stopped mode)

stop, start and delete the container

docker stop containerId/containerName

docker start contaienrId/containerName

docker rm containerId/contaienrName : if container is running you can’t delete.

1st option

Stop and delete

2nd option use -f deleted

docker rm containerId/containerName -f

remove the image

docker rmi imageName/imageId : if image link with container it doesn’t allow do delete.

docker rmi imageName/imageId -f

or

delete the container then delete the image.