Day 11 : 26 Sep 2024

Docker is an open-source platform for developer as well as system admin to build the application, ship and run distributed application in simplest way.

Docker is an advanced of virtualization.

Using docker we can create containerization application with help of docker engine.

Base machine Window if we want to run any application software we need system software ie OS.

One machine we can run multi OS.

Limitation of this approach we can run only one OS at time.

VM Ware software or Oracle Virtual BOX.

This software help use to run more than one OS.

With help of VM ware software we can achieve virtualization means we can Run OS in abstract model. Using Virtualization we can creation abstract version of an OS.

If my base OS contains 16 RAM or 1 TB external hard disk. If we want to run OS using VMware software we need to share resources like RAM and external memory.

Using Containerization we can create abstraction version of an application.

With help of docker we create the image. Image is responsible to run the application in an abstract way.



Docker images Docker images is a read only template file which is responsible to run the application with help of container.

To create Docker images we need to take help of Dockerfile and inside this we need to provide set of instruction to run the application or app with there dependencies.

docker –version

docker images this command display al images present in local machine.

docker pull imageName

hello-world is one of the pre defined images

docker images

docker engine by default configure with docker hub account.

Docker hub is an open source remote repository which help to pull as well as push images. It is like a github but in git hub we can push as well as pull any type of files. But in Docker hub we need to publish the images.

Please docker hub account with your personal email id.

Creating custom image to display welcome message.

**vim Dockerfile**

**type i to move insert mode**

FROM busybox

CMD ["echo","Welcome to Docker Image created by Akash"]

esc

wq!

docker build -t my-busybox . -f Dockerfile.txt

Creating image to run java program

Demo.java

class Demo {

public static void main(String args[]) {

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

}

}

Dockerfile

FROM openjdk:11

COPY Demo.java .

RUN javac Demo.java

CMD ["java","Demo"]

docker build -t my-java . -f Dockerfile

docker run my-java

creating image for spring boot application with contains one end point and same image we publish on docker hub account.

We can create jar or war file

1. Using eclipse IDE
2. Using mvn clean package

**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"]

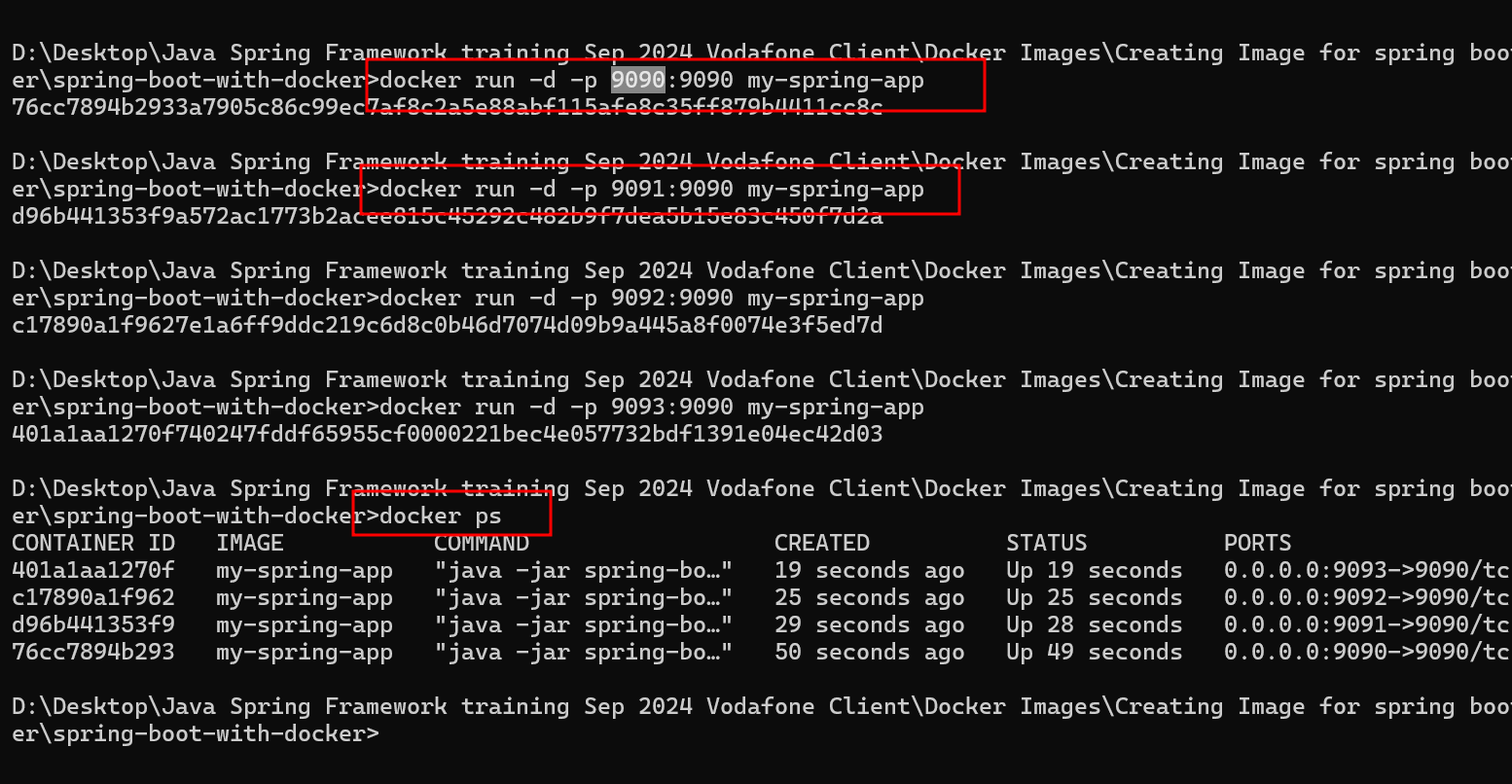
docker build -t my-spring-app . -f Dockerfile

if image is responsible to run the web application

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

-d detached mode

-p publish port number



Publish this image in docker hub account

docker login this command connect local machine with remote docker hub account

before publish we need to create the tag for that image. Tag is unique identity for the image

docker tag my-spring-app akashkale/my-spring-app:v1

docker push akashkale/my-spring-app:v1

docker run -d -p 8181:9090 akashkale/my-spring-app:v1

using docker ps

<http://localhost:8181>

Spring boot security

Using security we can achieve authentication as well as authorization

Core Java core security classes or interfaces

JEE security API

Server ie Tomcat or web logic provide their own security.

Spring framework provide security features

Spring boot.

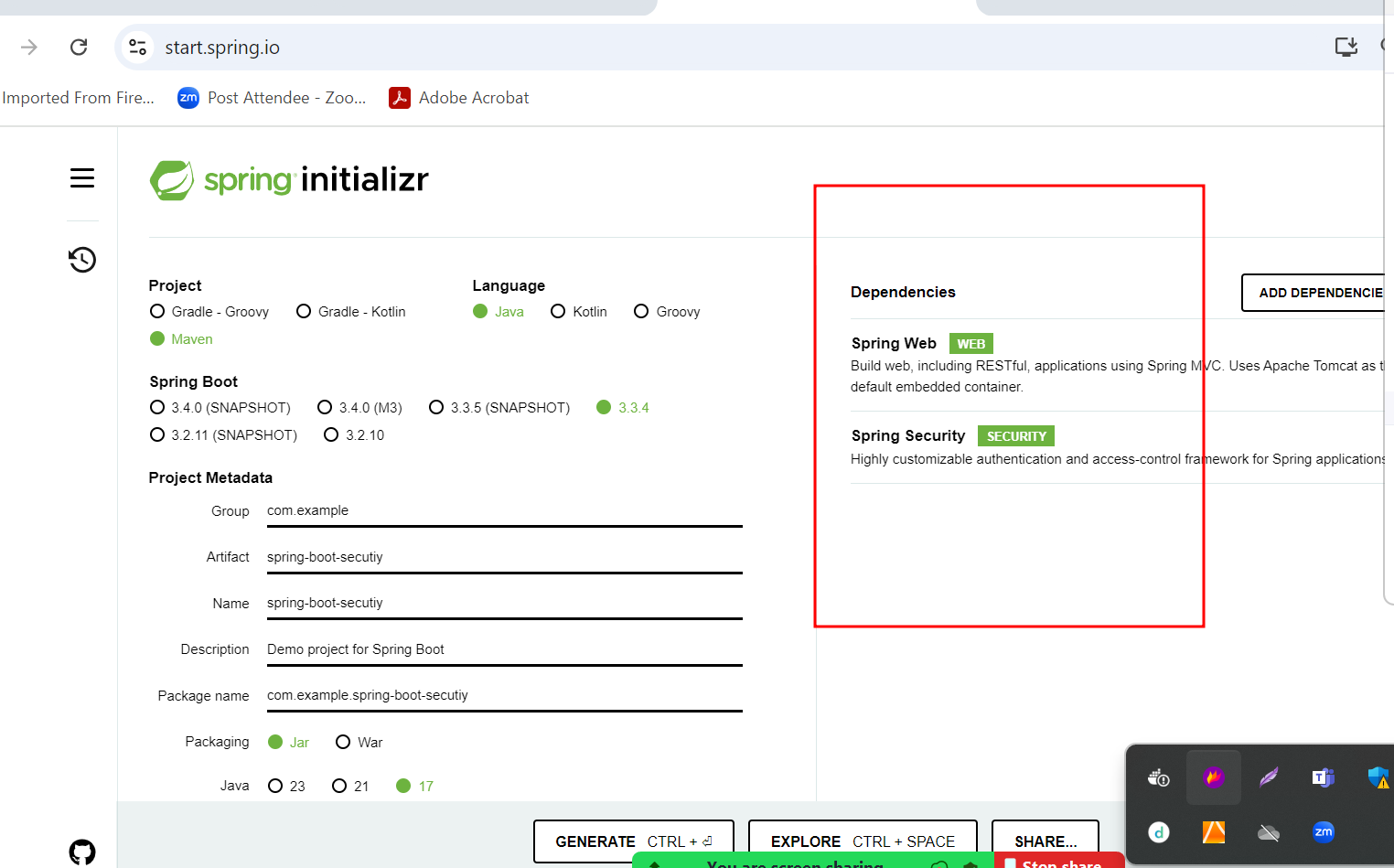
Authentication allow us to access the resources

Authorization allow to access the resources base upon the role.

Spring boot provided security starter.

Web starter

Security starter



By default spring boot provide use login page

Username as user and password random password display on console.

Spring boot provide few pre defined end point ie logout

If we want custom user name and password

We can configure using

1. application.properties
2. Using spring security classes

Spring boot proved few API ie UserDetailsService and UserDetails. Both are interfaces which hold user information retrieve from in memory or db.

Spring boot 2.x version spring classes are different

Spring boot 3.x version spring boot clases are different