10 July – 2024

Day 2

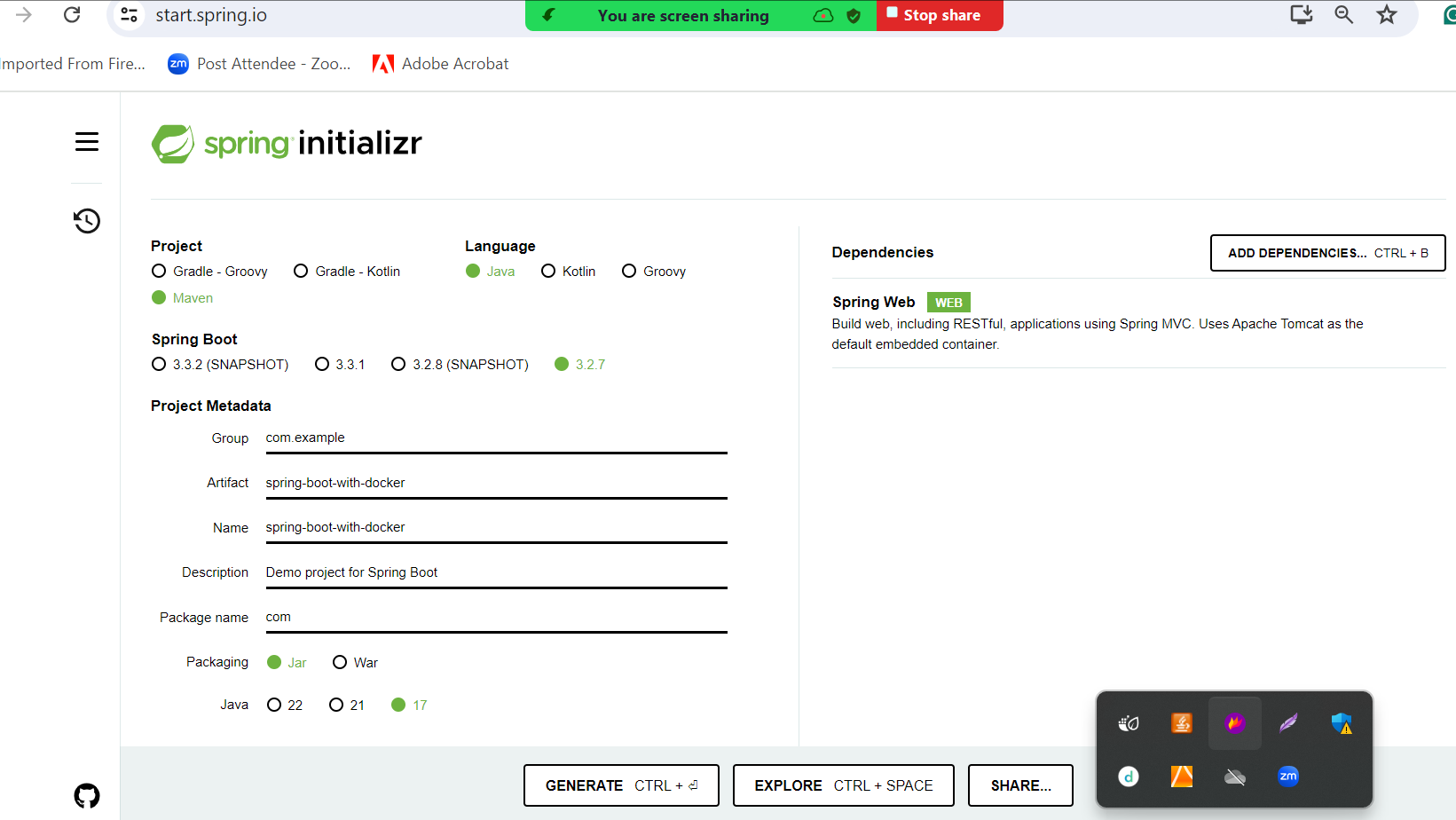
Integration and deployment

Docker

Docker compose

1. Creating image to run spring boot application

Using spring initializer we create simple spring boot application with web starter.



Please create more than one end points

Then create jar file using eclipse IDE with help of maven run 🡪 maven install

Then create 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-boot . -f Dockerfile**

if image is responsible to run the web application we need run the command as

docker run -d -p 9090:9090 imageName

-d : detached mode or background

-p publish port number

Right side or red colour port number is actual application port number is spring boot running port number.

Left side or green colour port number is expose port number it can be same or other number

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

docker run -d -p 9091:9090 my-spring-boot

docker run -d -p 9092:9090 my-spring-boot

then check on browser using port number as

90, 91 and 92

<http://localhost:9090>

<http://locahost:9091>

<http://locahost:9092>

if we want to see all running container we need to run the command as below. The below command display only running container.

docker ps

or

docker container ls

docker ps -a it display all container present in our local machine doesn’t matter they are running or stop.

To check specific contains logs details

docker logs containerId/containerName

stop the container

docker stop containerId/containerName

start the container

docker start containerId/containerName

remove container

docker rm containerId/containerName make sure container not running

or

docker rm containerId/containerName -f

command to remove the image

docker rmi imageName/imageId

or

docker rmi imageName/imageId -f

to remove all stopped container, images, network and cache memory.

docker system prune -a

mysql images

docker hub internally provide mysql image

docker run -e MYSQL\_ROOT\_PASSWORD=root -d -p 3307:3306 mysql:8.0

-e :ENVIRONMENT DETAILS like MYSQL\_ROOT\_PASSWORD as root

-d : detached mode

-p publish port number

docker ps