2. How to dockerize the spring boot application?

Dockerizing a Spring Boot application involves creating a Docker image of your application and then running it as a container. Here are the general steps to follow:
Please download a spring project to your local machine or EC2 instance from the GIT hub.

git clone https://github.com/benoynsreedhar/springwebapp

Change the folder to **cd springwebapp** and build the spring project with maven.
mvn clean package

1. **Create a Dockerfile:** This file will contain instructions on how to build your Docker image. Create the Dockerfile for a Spring Boot application: **sudo vi Dockerfile**

```
FROM openjdk:11-jdk-slim
VOLUME /tmp
ARG JAR_FILE=target/*.jar
COPY ${JAR_FILE} app.jar
ENTRYPOINT ["java","-jar","/app.jar"]
```

This Dockerfile starts with the official OpenJDK 11 image, sets a volume for temporary files, copies the Spring Boot JAR file into the image, and sets the command to run the JAR file.

2. Build the Docker image: Once you have created the Dockerfile, you can build the Docker image using the following command in the directory where the Dockerfile is located:

```
sudo docker build -t my-spring-boot-app .
```

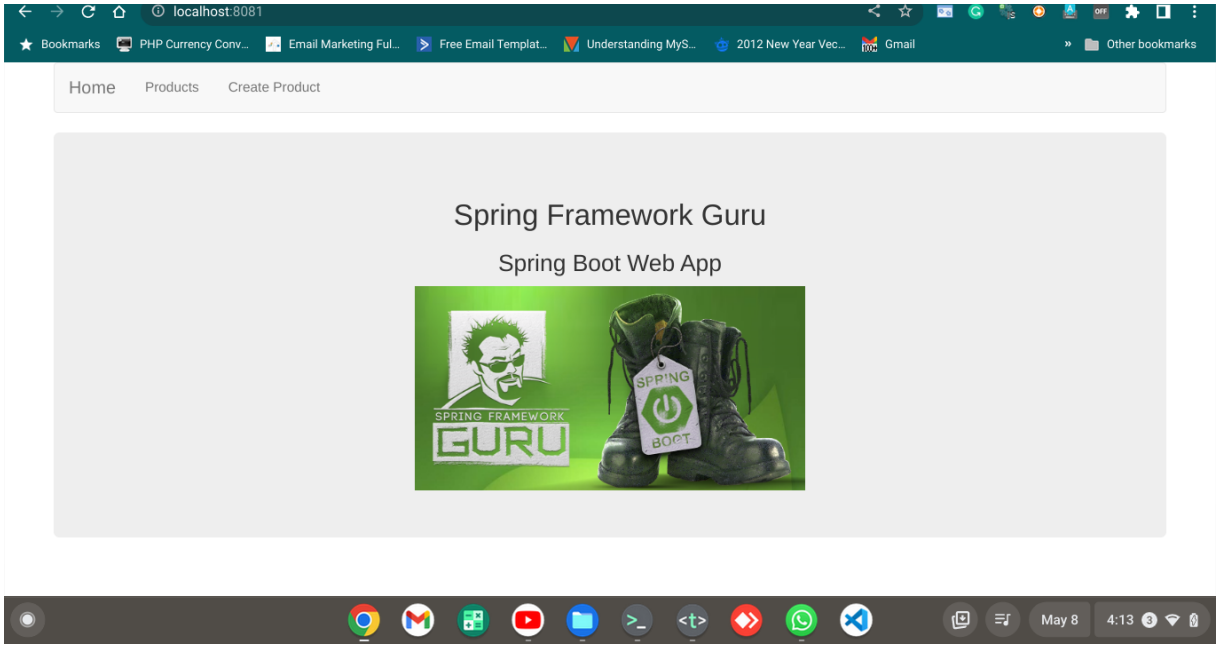
This command will create a Docker image with the tag `my-spring-boot-app` based on the instructions in the Dockerfile.

3. Run the Docker container: After the Docker image is built, you can run it as a Docker container using the following command:

```
sudo docker run -p 8081:8080 my-spring-boot-app
```

This command will run the Docker container and map port 8080 in the container to port 8080 on the host machine.

Local machine: **http://localhost:8081/**
On Your AWS Instance : **http://<Instance Public IP>:8081/**



That's it! Your Spring Boot application is now running in a Docker container.