### Write a docker file to crate a docker image for your Springboot application and push that image to docker hub repository. then pull the same image from repository and run it.

1. Create a docker file and keep it at the same place where super pom.xml is available in springboot application.

Springboot maven project url : https://github.com/panidummy/Maven\_SpringBoot.git

Dockerfile:

**FROM** openjdk:8-jdk-alpine

**EXPOSE** 8081

**ARG** JAR\_FILE=target/\*.jar

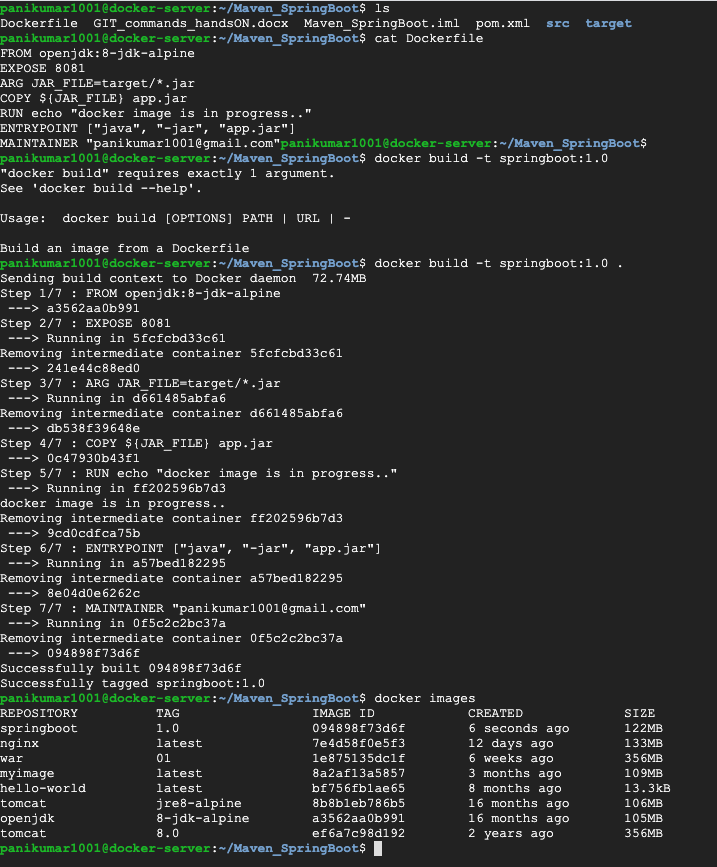
**COPY** ${JAR\_FILE} app.jar

**RUN** echo **"docker image is in progress.."**

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

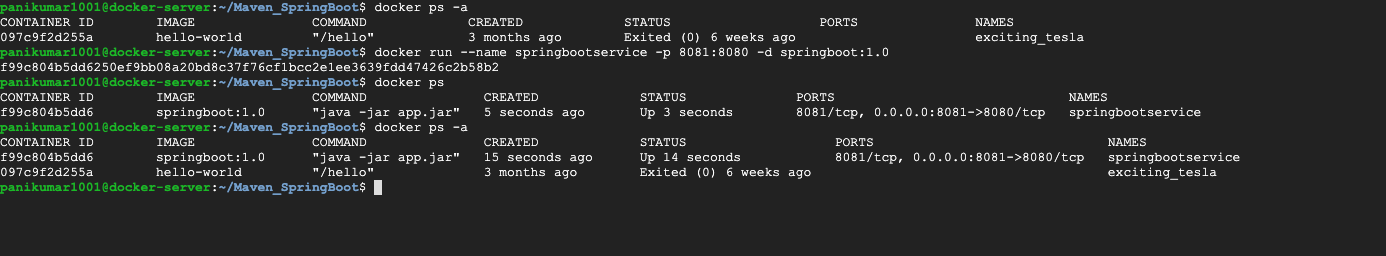
**MAINTAINER "panikumar1001@gmail.com"**

2. Create a docker image :

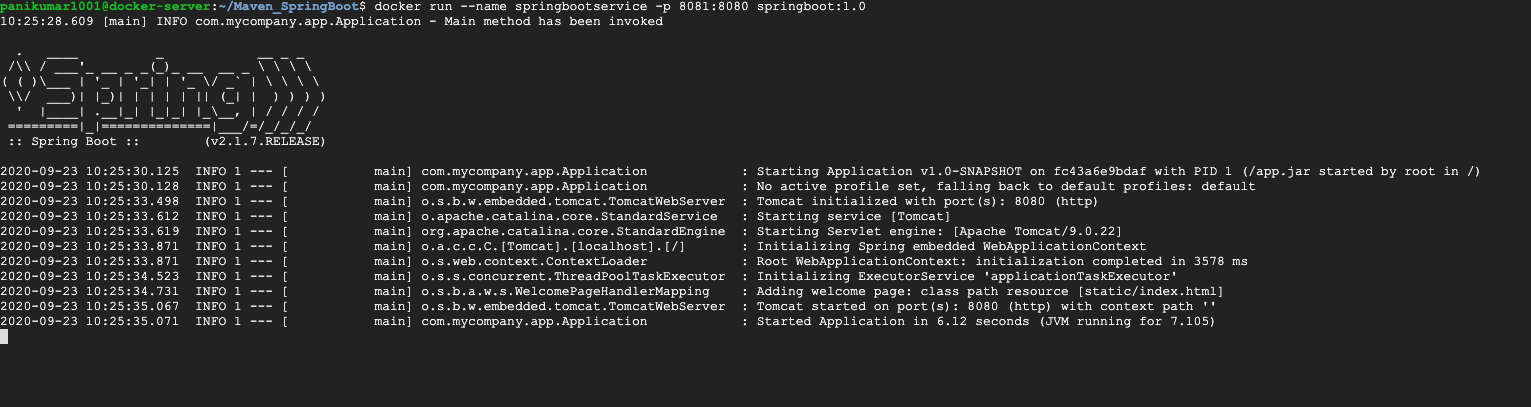
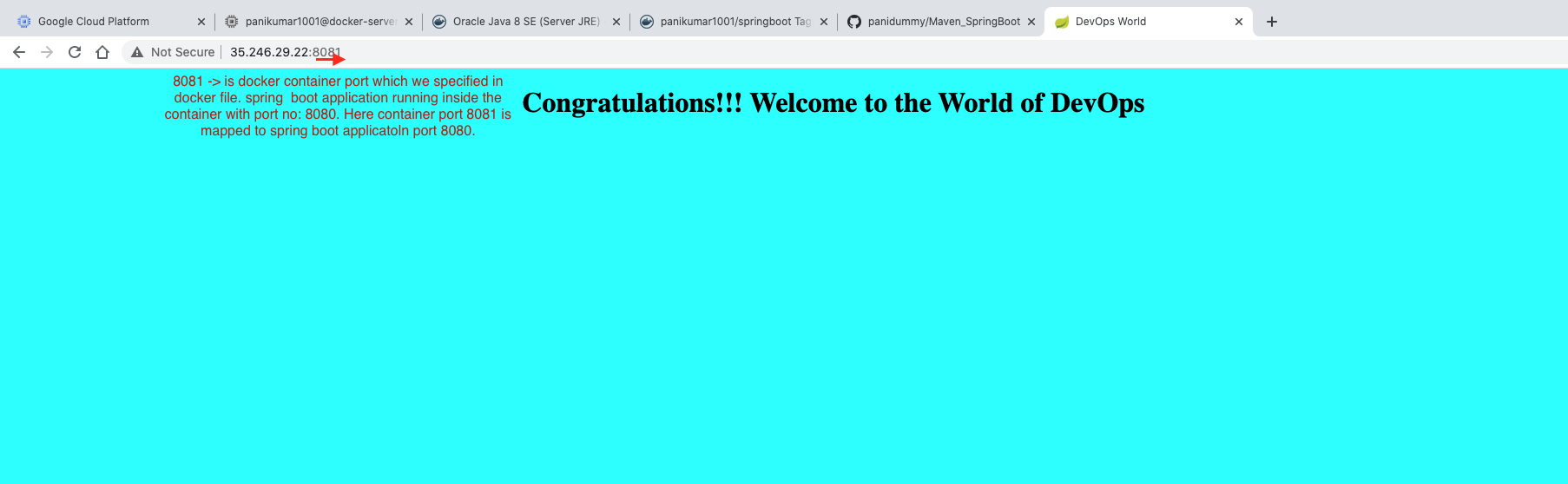


3. Run docker image as service:

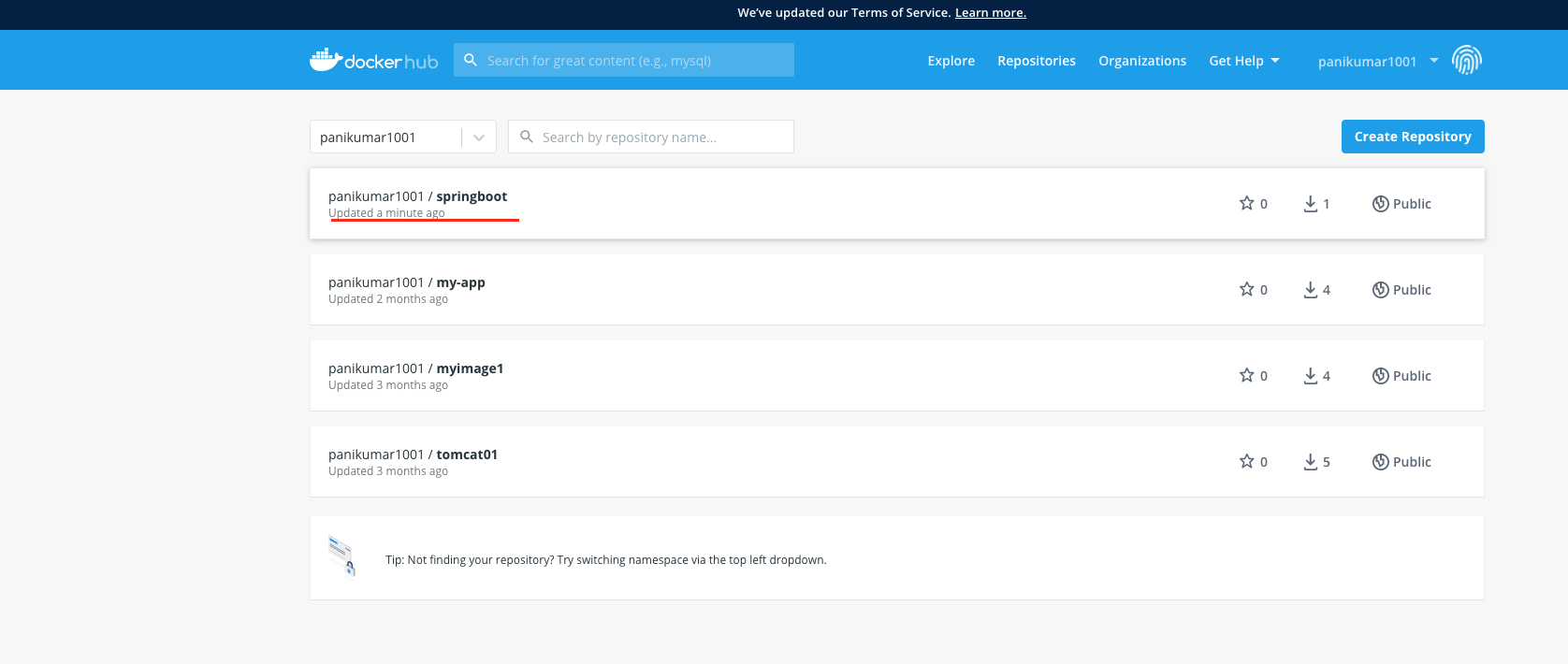
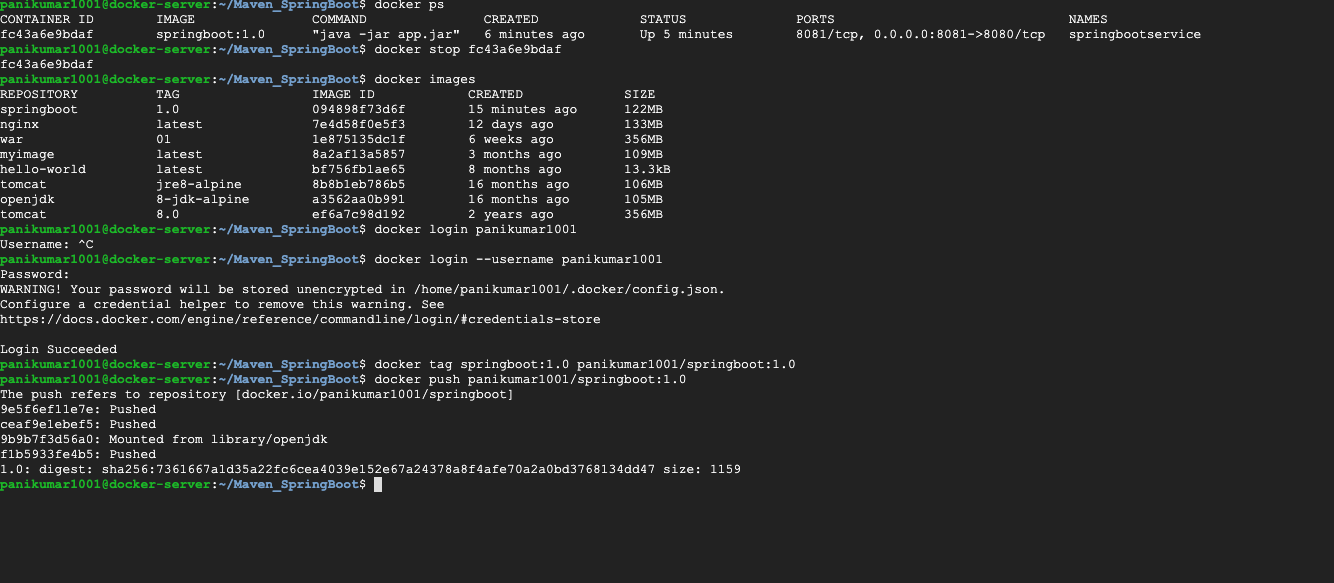
-> with detached mode (i.e using -d )



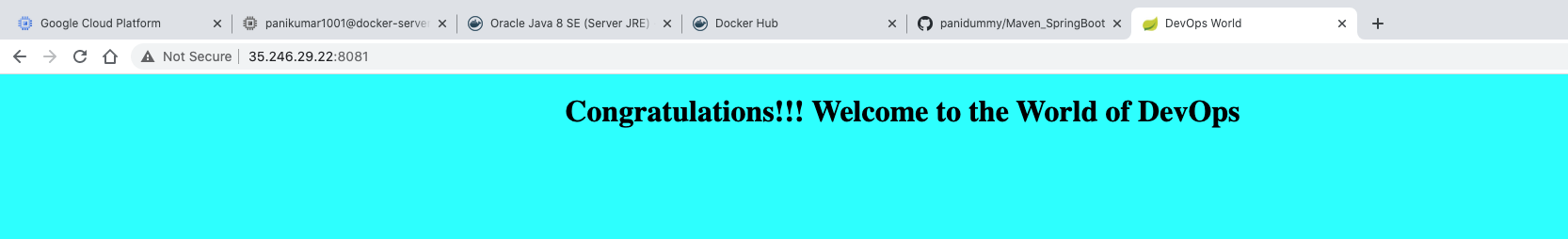
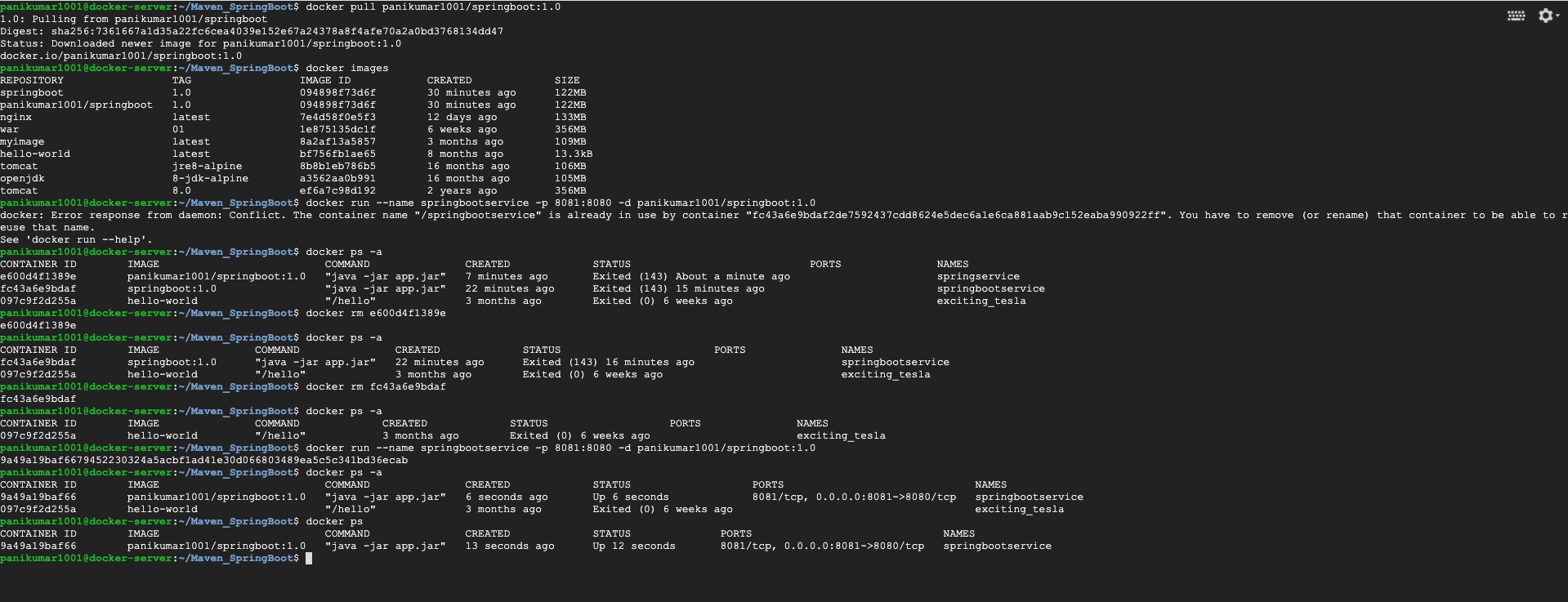
-> without detached mode (i.e with out using -d)



4. push docker image to docker repository (i.e. docker hub).



5. Pull docker image from docker hub and run it:



### 2. Write a docker file to create docker image for a Dynamic java project (War project) and push that image into docker hub. Try running it in different environment by pulling it. Remove the local image and then pull from docker repository and run it if you don’t have another environment)

Dockerfile:

**FROM** tomcat:9.0

**EXPOSE** 8082

**ARG** WAR\_FILE=/target/\*.war

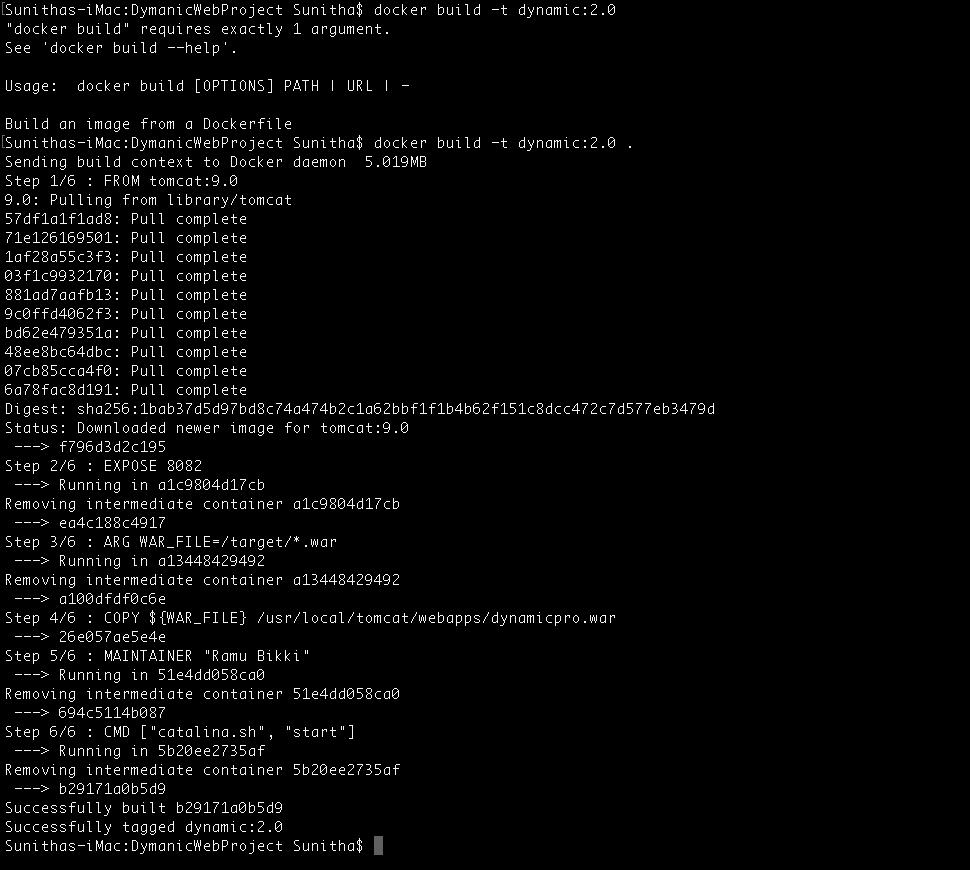
**COPY** ${WAR\_FILE} /usr/local/tomcat/webapps/dynamicpro.war

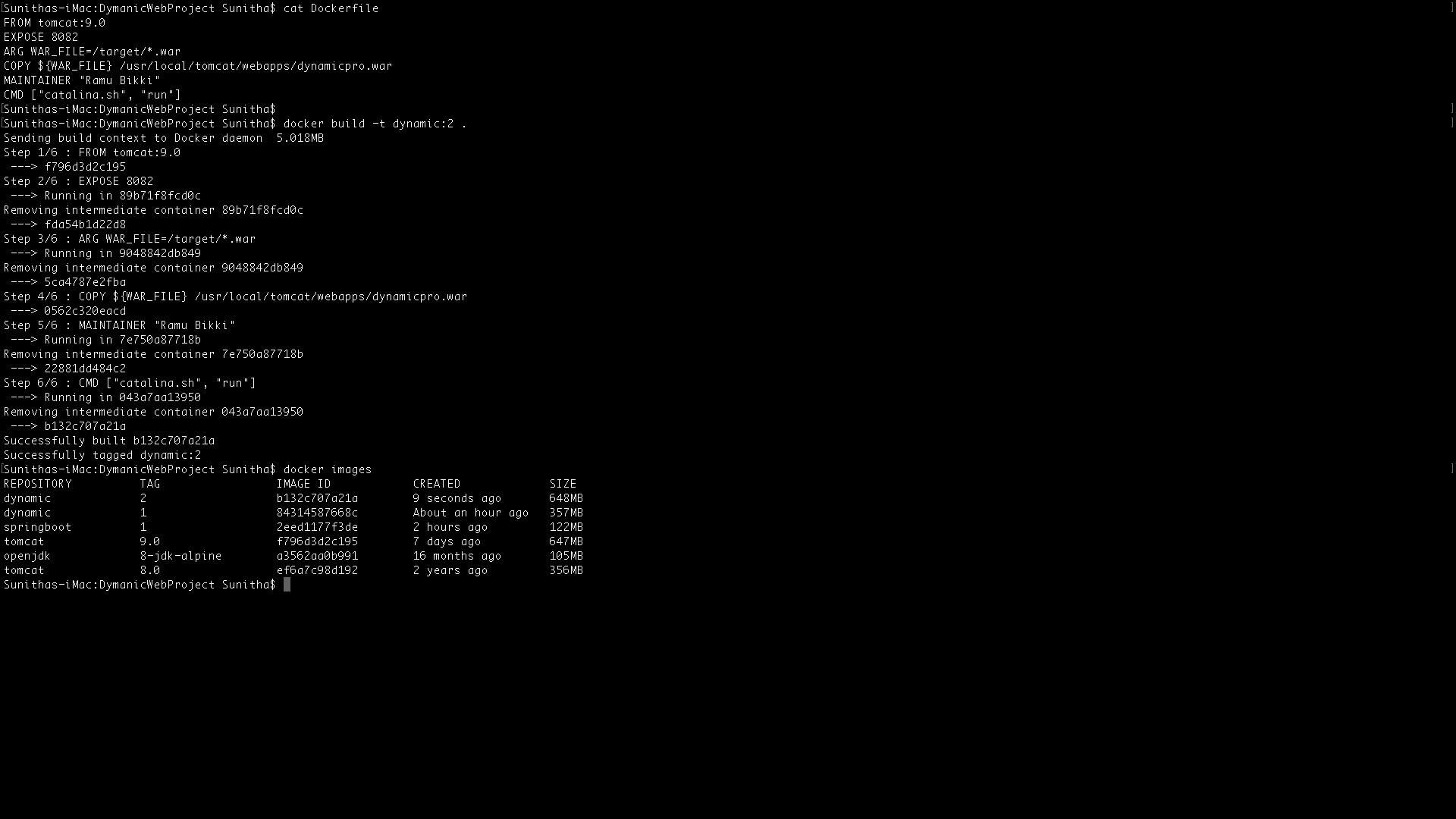
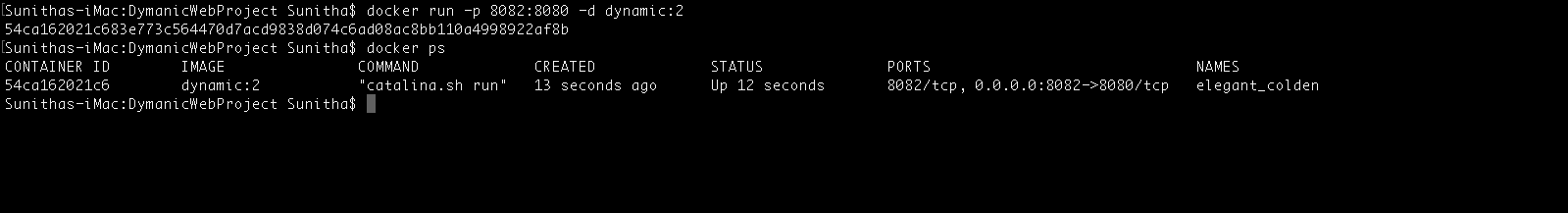
**MAINTAINER "Ramu Bikki"**

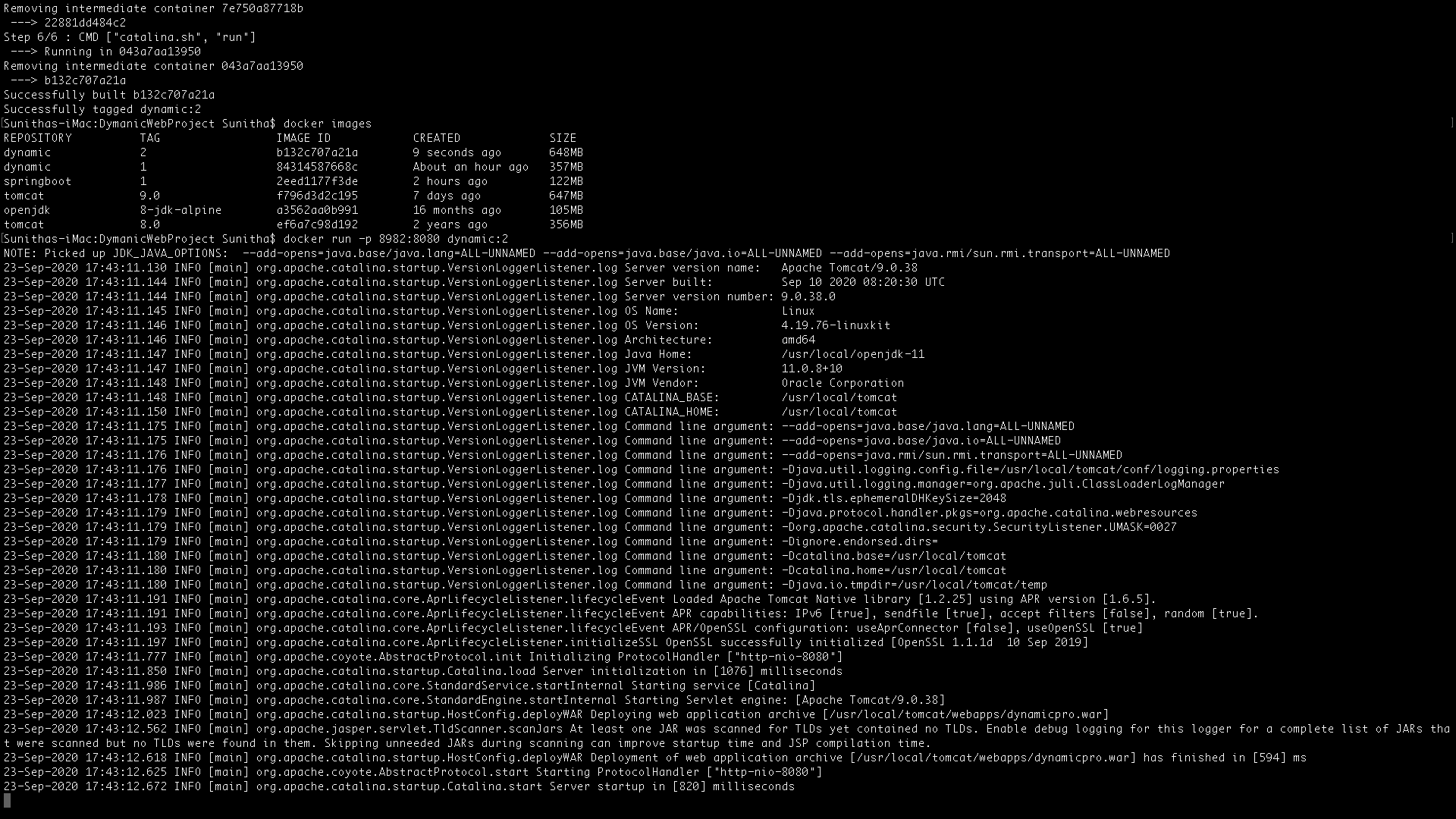
**CMD** [**"catalina.sh"**, **"run"**]

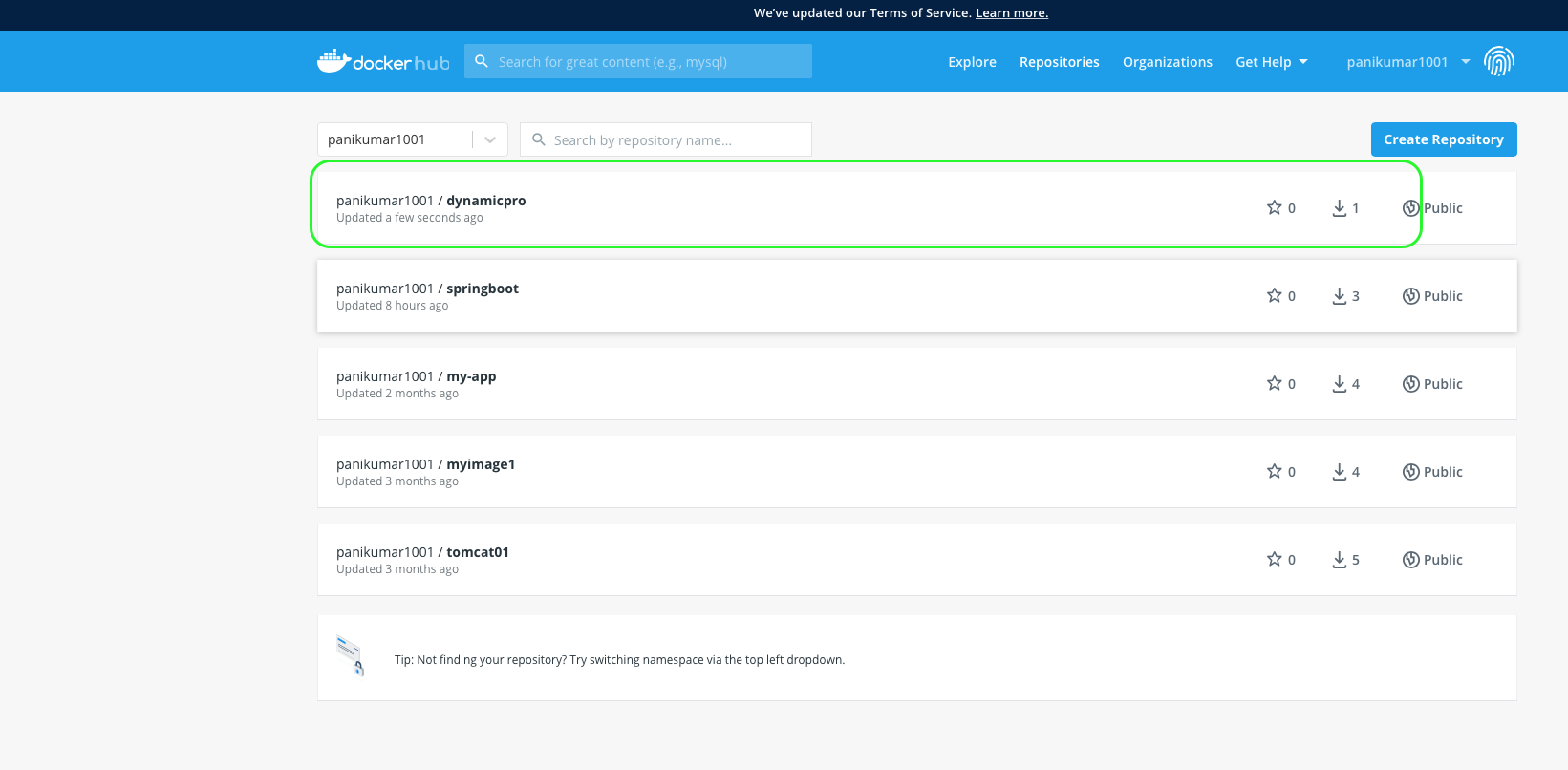
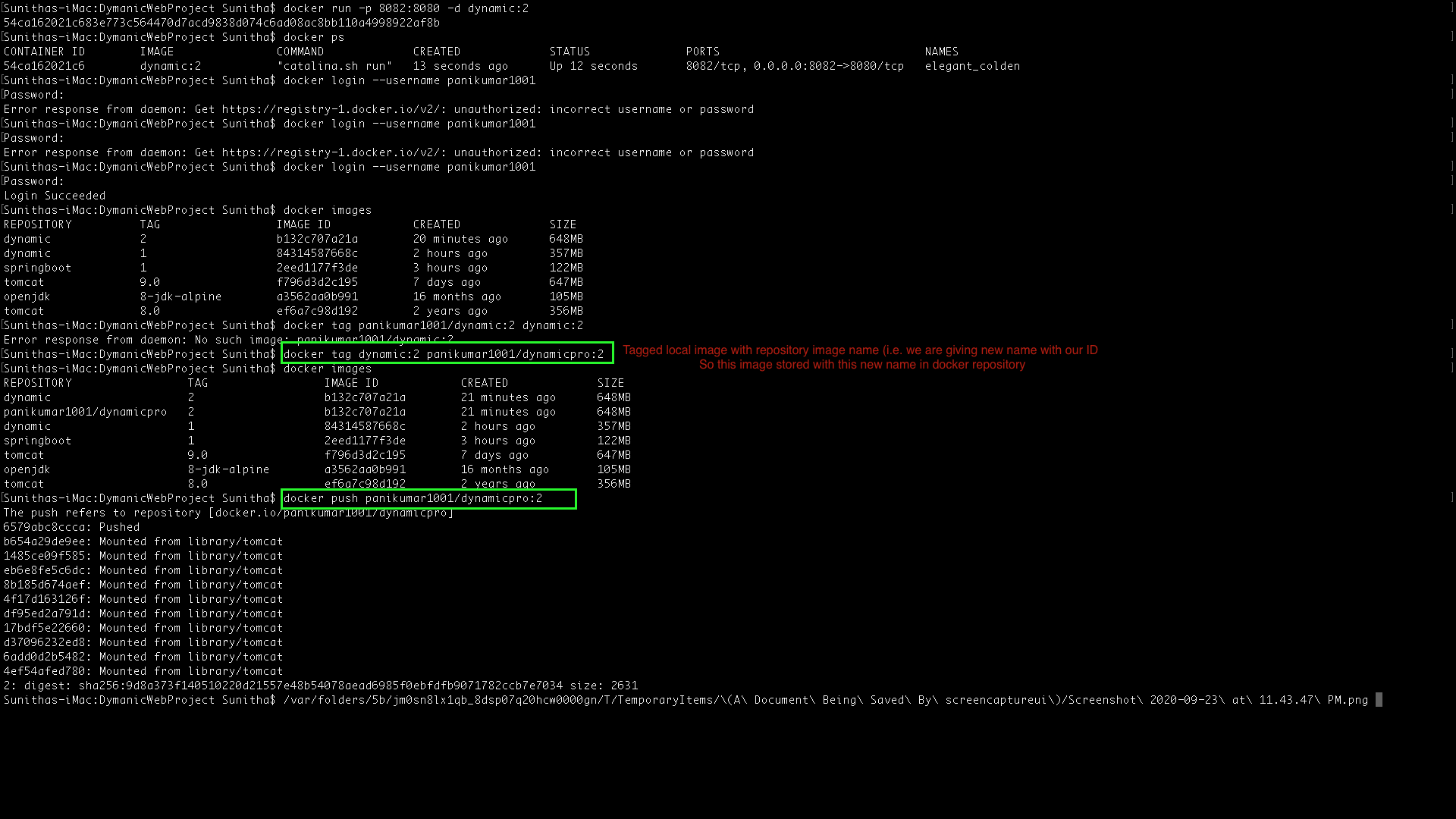
This Docker file is available in git repository

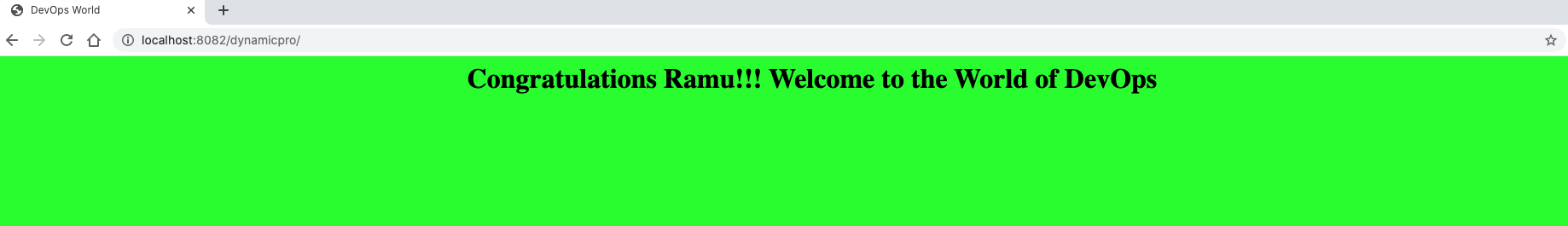
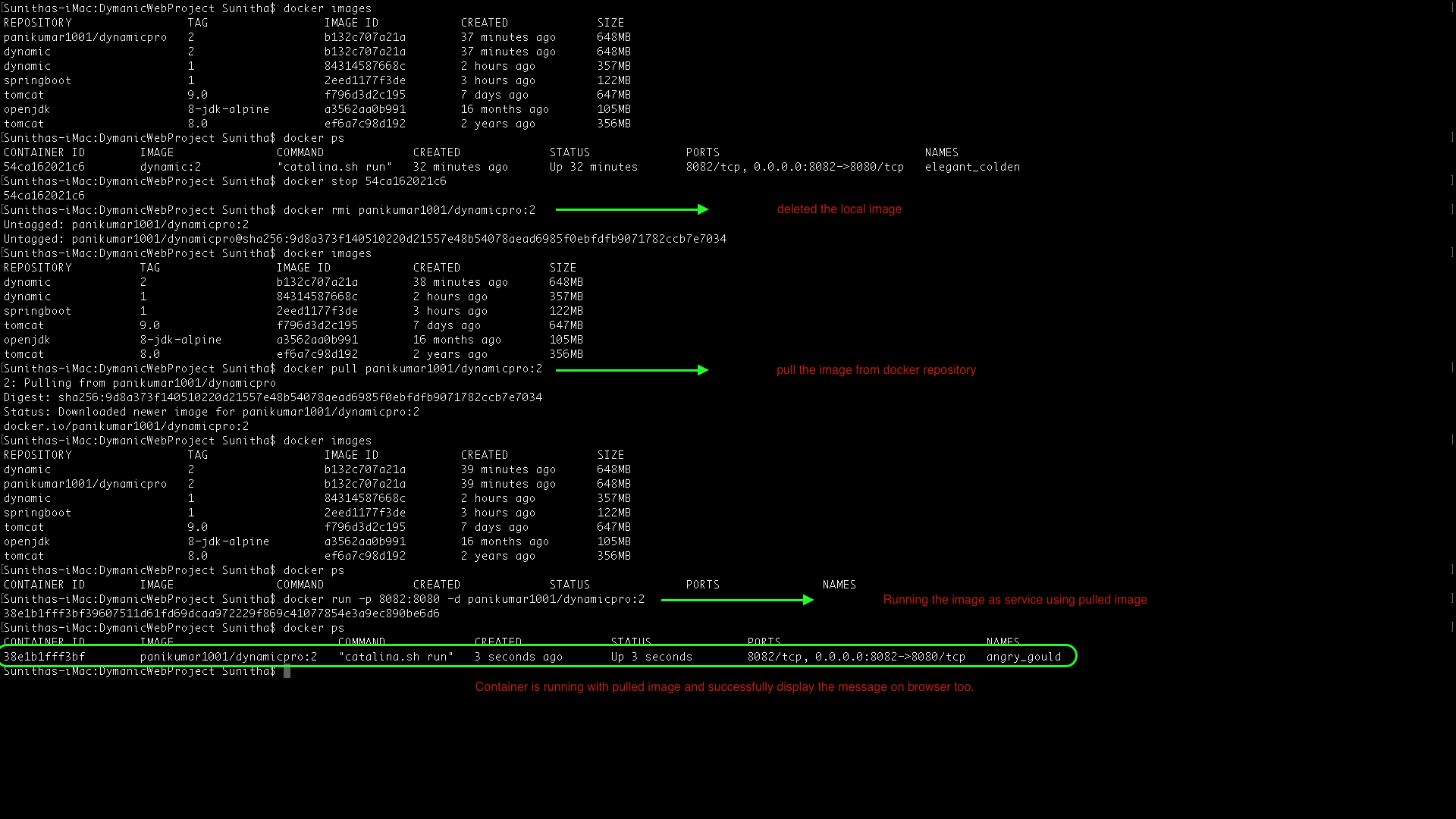
<https://github.com/panidummy/DynamicWeb.git>











### 3. Write a docker compose file to run spring boot application along with mySql database container and establish connection between springboot application container and mysql container. We need to write lines of code in spring boot application to connect mysql db which is running in another container.

docker-compose.yml;

**version**: **'3'**

**services**:

**docker-mysql**:

**restart**: always

**container\_name**: docker-mysql

**image**: mysql

**volumes**:

- ./sql:/docker-entrypoint-initdb.d

**ports**:

- **"6033:3306"**

**spring-boot-app**:

**restart**: on-failure

**build**: ./

**expose**:

- **"8080"**

**ports**:

- 8080:8082

**environment**:

**WAIT\_HOSTS**: mysql:3306

**depends\_on**:

- docker-mysql