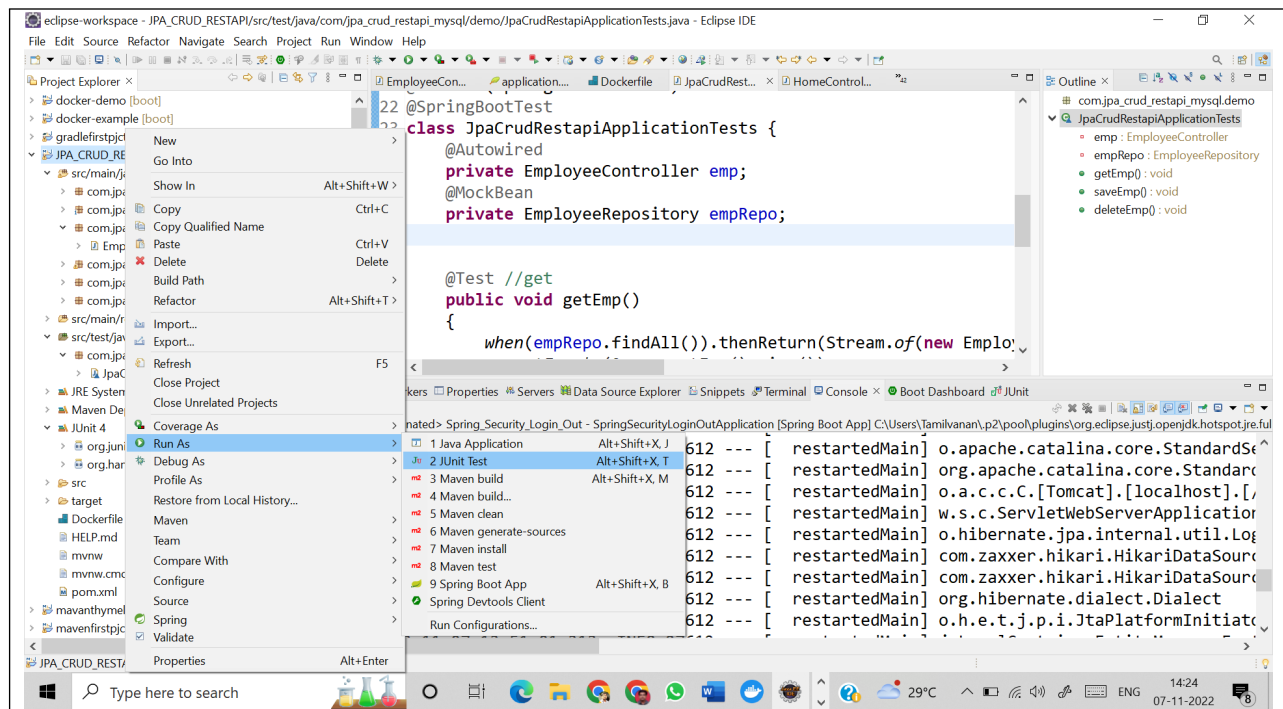


Ex:No:5	Testing and Deploying SpringBoot Application
Date:	
Aim	
<ul style="list-style-type: none">To perform Junit test for REST API and to deploy springboot application in docker.	
Procedure	
<p>1. Perform Junit testing in springboot application based on controller/service class</p> <p>Unit Testing is a one of the testing done by the developers to make sure individual unit or component functionalities are working fine.</p> <p>Add the following code in src/test/java application file</p> <pre>@RunWith(SpringRunner.class) @SpringBootTest class JpaCrudRestapiApplicationTests { @Autowired private EmployeeController emp; @MockBean private EmployeeRepository empRepo; @Test //get public void getEmp() { when(empRepo.findAll()).thenReturn(Stream.of(new Employee(10,"xxx",50,50000),new Employee(11,"yyy",40,35000)).collect(Collectors.toList())); assertEquals(2,emp.getEmp().size()); } @Test //Insert public void saveEmp() { Employee empl=new Employee(20,"UUU",19,10000); when(empRepo.save(empl)).thenReturn(empl); assertEquals(empl,emp.createEmp(empl)); } @Test //Delete public void deleteEmp() { int id=10; emp.deleteEmp(id); verify(empRepo,times(1)).deleteById(id); } }</pre>	

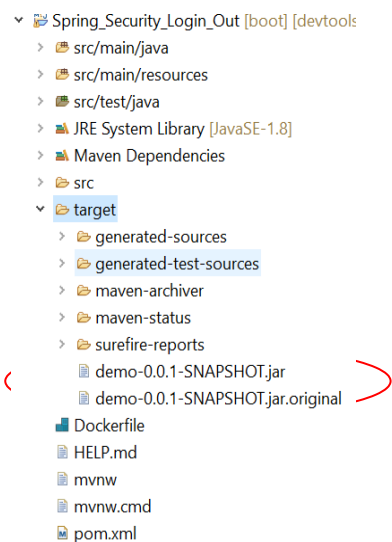


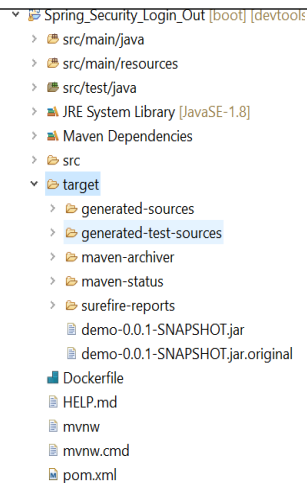
2. Deploy Locally a Spring Boot Application in Docker

Prerequisites

- Docker environment installed and configured..
- Have a Docker Hub account. Register for a free account.

Step: 1 create the .jar file for your SpringBoot Application





Dockerfile contains the commands to create the image

```
From openjdk:8
ADD target/*.jar secapp.jar
ENTRYPOINT ["java", "-jar", "secapp.jar"]
```

FROM - Must be the first non-comment instruction in the Dockerfile. This command creates a layer from the Docker image. In our case, we have used `java:8` which means this application will run on Java 8.

ADD - This command helps to take a source and destination. Normally, the source is your local copy. The **COPY** command also does same thing, but there is a small difference between the **COPY** and **ADD** commands.

ENTRYPOINT - It's similar to **CMD**, where our command/jar file will be executed.

Step 3: Run the command to build the image and deploy it to Docker

Create a Docker image file.

Open Command Prompt and follow the below steps...

The below command is used to create the image file

```
>> docker build -t secapp .
```

check the created image by using the "**docker images**" command

Create an image for mysql

```
>> docker pull mysql
```

Load mysql image into container

```
>>docker run -p 3308:3306 --name mysqlsec -e MYSQL_ROOT_PASSWORD=password -e  
MYSQL_DATABASE=User_Mgmt -e MYSQL_USER=user -e  
MYSQL_PASSWORD=password -d mysql
```

Create a network to connect the mysql with springboot application

```
>>docker create network spring-net
```

Check the created network is added in docker

```
>>docker network ls
```

Connect the created network with mysql

```
>>docker network connect spring-net mysqlsec
```

Check the connection is established with mysqlsec

```
>> docker container inspect mysqlsec
```

Load your springboot application in docker container

```
>>docker run -p 9090:8086 --name secapp --net spring-net -e MYSQL_USER=user -e  
MYSQL_PASSWORD=password -e MYSQL_HOST=mysqlsec -e MYSQL_PORT=3306  
secapp
```

Your application starts running now and Open your browser and type

localhost:9090

To access mysql with docker

```
Command Prompt - docker exec -it mysqlsec bash
C:\Users\Tamilvanan\eclipse-workspace>cd spring_security_login_out
C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker exec -it mysqlsec bash
bash-4.4# mysql -uuser -ppassword
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 28
Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

```
Command Prompt - docker exec -it mysqlsec bash
mysql> show databases;
+-----+
| Database |
+-----+
| User_Mgmt |
| information_schema |
| performance_schema |
+-----+
3 rows in set (0.00 sec)

mysql> use User_Mgmt;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

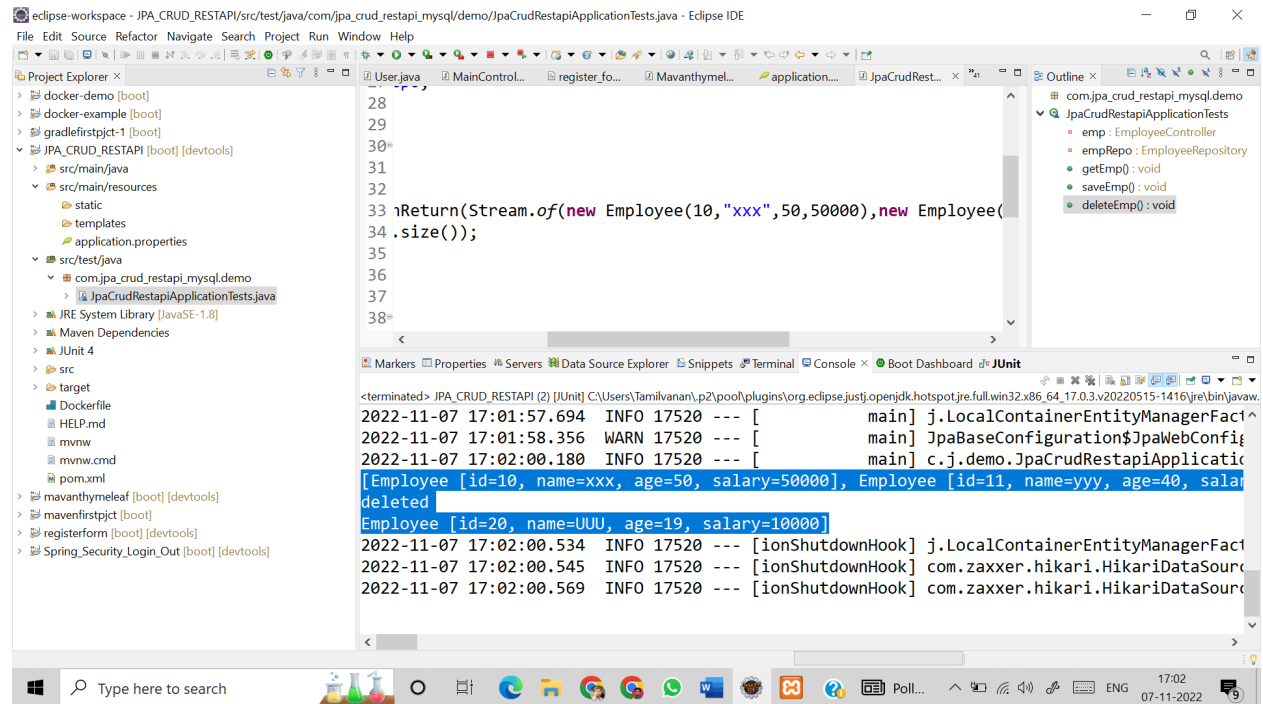
Database changed
mysql> select * from user;
Empty set (0.00 sec)

mysql>
```

SAMPLE PROGRAM

OUTPUT

Junit Test :



The screenshot displays the Eclipse IDE interface. The Project Explorer on the left shows the project structure for 'JPA_CRUD_RESTAPI'. The main editor shows the file 'JpaCrudRestapiApplicationTests.java' with line numbers 28 to 38. The console at the bottom shows the output of the JUnit test run, including log messages from JUnit and HikariDataSource. The test output is as follows:

```
<terminated> JPA_CRUD_RESTAPI (2) [JUnit] C:\Users\Tamilvanan\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64.17.0.3.v20220515-1416\jre\bin\javaw.  
2022-11-07 17:01:57.694 INFO 17520 --- [main] j.LocalContainerEntityManagerFacto  
2022-11-07 17:01:58.356 WARN 17520 --- [main] JpaBaseConfiguration$JpaWebConfig  
2022-11-07 17:02:00.180 INFO 17520 --- [main] c.j.demo.JpaCrudRestapiApplicatio  
[Employee [id=10, name=xxx, age=50, salary=50000], Employee [id=11, name=yyy, age=40, sala  
deleted  
Employee [id=20, name=UUU, age=19, salary=10000]  
2022-11-07 17:02:00.534 INFO 17520 --- [ionShutdownHook] j.LocalContainerEntityManagerFacto  
2022-11-07 17:02:00.545 INFO 17520 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSourc  
2022-11-07 17:02:00.569 INFO 17520 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSourc
```

Create a Docker image file for your application and mysql.

>>docker pull mysql

and

```
Command Prompt
C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker build -t secapp .
[+] Building 2.5s (7/7) FINISHED
=> [internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 31B 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/openjdk:8 2.2s
=> [internal] load build context 0.1s
=> => transferring context: 80B 0.0s
=> [1/2] FROM docker.io/library/openjdk:8@sha256:86e863cc57215cfb181bd319736d0baf625fe 0.0s
=> CACHED [2/2] ADD target/*.jar secapp.jar 0.0s
=> exporting to image 0.1s
=> => exporting layers 0.0s
=> => writing image sha256:1aa26d92ee5162c835f079e6c15aeb6984e1981647b82b618f7a25abfd7 0.0s
=> => naming to docker.io/library/secapp 0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fi
x them

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>
```

check the created image by using the "**docker images**" command

```
Command Prompt
C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
secapp              latest     1aa26d92ee51  3 minutes ago 573MB
app                 latest     4a084cc6915f  3 days ago   570MB
<none>              <none>     477c65d1e8b5  5 days ago   565MB
<none>              <none>     452399d06e3b  6 days ago   544MB
docker-example      latest     2f8864b0baf5  6 days ago   544MB
<none>              <none>     c6887ce0dd0e  6 days ago   570MB
<none>              <none>     8eba7b6ba336  6 days ago   570MB
docker101tutorial   latest     a429ce4d1795  7 days ago   28.9MB
yuvaranip/docker101tutorial latest     a429ce4d1795  7 days ago   28.9MB
mysql               latest     c2c2eba5ae85  10 days ago  535MB
ubuntu              latest     cdb68b455a14  13 days ago  77.8MB
alpine/git          latest     42a1cda0ba24  2 weeks ago  43.6MB
hello-world         latest     feb5d9fea6a5  13 months ago 13.3kB

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>
```

Load mysql image into container


```
Command Prompt
C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker run -p 3308:3306 --name
mysqlsec -e MYSQL_ROOT_PASSWORD=password -e MYSQL_DATABASE=User_Mgmt -e MYSQL_USER=user -e MY
SQL_PASSWORD=password -d mysql
7b57e1249fcc0323d0f8fede7f87bbaa11a036fd1d2ad2c71a34adf04b70dadbd

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
PORTS         NAMES
7b57e1249fcc   mysql         "docker-entrypoint.s..." 14 seconds ago Up 11 seconds
33060/tcp, 0.0.0.0:3308->3306/tcp mysqlsec
0e217fc8549b   app          "java -jar app.jar"      4 hours ago   Up 2 hours
0.0.0.0:9090->8086/tcp app
35f37952315e   mysql         "docker-entrypoint.s..." 4 hours ago   Up 4 hours
33060/tcp, 0.0.0.0:3307->3306/tcp mysqldb
db8cbec157e9   docker-exampl "java -jar docker-ex..." 6 days ago     Exited (143) 6 da
ys ago          intelligent_meitner
db235da30c30   452399d06e3b  "java -jar 'docker-e..." 6 days ago     Exited (1) 6 days
ago             mystifying_varahamihira
703ceac10fa8   452399d06e3b  "java -jar 'docker-e..." 6 days ago     Exited (1) 6 days
ago             zen_goldberg
6479c9a899e2   hello-world   "/hello"                 6 days ago     Exited (0) 6 days
ago             compassionate_mahavira
```

Create a network to connect the mysql with springboot application

```
Command Prompt
C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker network create spring-secapp
58ecb77a356bac2e6946ec422f3650adc5b5a0c08ce35b27cfb464a0fab77dc

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker network ls
docker: 'networkl' is not a docker command.
See 'docker --help'

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker network ls
NETWORK ID     NAME      DRIVER  SCOPE
08b4ccbe749a   bridge    bridge   local
e5ea40f4cee5   host      host     local
3ca7c76502f4   mysqlnet  bridge   local
08352dd15fc9   none      null     local
7052491010e5   spring-app bridge   local
67c8773baa7f   spring-net bridge   local
58ecb77a356b   spring-secapp bridge   local

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>
```

```
Command Prompt

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker network connect spring-net mysqlsec

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>
```

English (India)
English (India) keyboard
To switch input methods, press
Windows key+Space.

Type here to search

29°C

14:09
07-11-2022

Load your springboot application in docker container

```
Command Prompt - docker run -p 9092:8076 --name secapp --net spring-net -e MYSQL_USER=user -e MYSQL_PASSWORD=password -e MYSQL_HOST=mysqlsec -e MYSQL_PORT=3306 secapp

C:\Users\Tamilvanan\eclipse-workspace\Spring_Security_Login_Out>docker run -p 9092:8076 --name secapp --net spring-net -e MYSQL_USER=user -e MYSQL_PASSWORD=password -e MYSQL_HOST=mysqlsec -e MYSQL_PORT=3306 secapp

:: Spring Boot ::
(v2.6.5)

2022-11-07 08:46:50.776 INFO 1 --- [main] c.s.d.SpringSecurityLoginOutApplication : Starting SpringSecurityLoginOutApplication v0.0.1-SNAPSHOT using Java 1.8.0_342 on 822a38017852 with PID 1 (/secapp.jar started by root in /)
2022-11-07 08:46:50.781 INFO 1 --- [main] c.s.d.SpringSecurityLoginOutApplication : No active profile set, falling back to 1 default profile: "default"
2022-11-07 08:46:52.116 INFO 1 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2022-11-07 08:46:52.202 INFO 1 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 68 ms. Found 2 JPA repository interfaces.
2022-11-07 08:46:53.277 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8076 (http)
2022-11-07 08:46:53.295 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-11-07 08:46:53.295 INFO 1 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.60]
2022-11-07 08:46:53.393 INFO 1 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2022-11-07 08:46:53.393 INFO 1 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2512 ms
2022-11-07 08:46:54.035 INFO 1 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH0000204: Processing PersistenceUnitInfo [name: default]
2022-11-07 08:46:54.118 INFO 1 --- [main] org.hibernate.Version : HHH0000412: Hibernate ORM core version
```

Type here to search

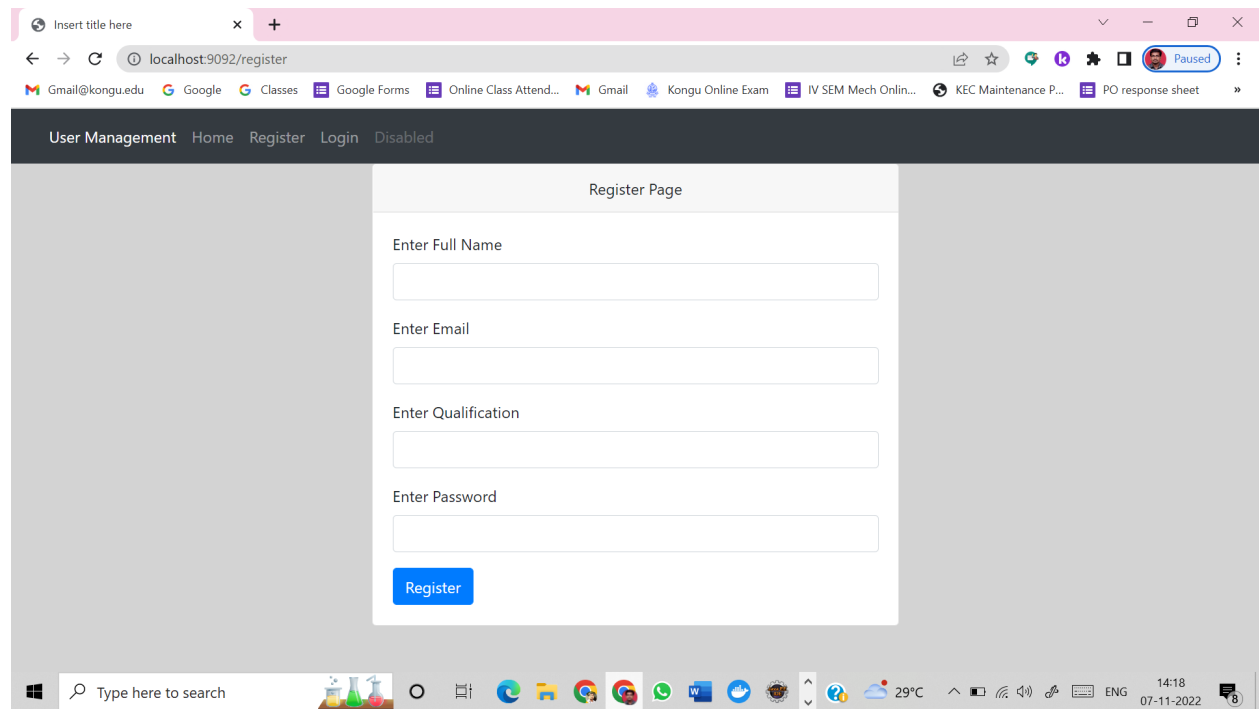
29°C

14:17
07-11-2022

```
Command Prompt - docker run -p 9092:8076 --name secapp --net spring-net -e MYSQL_USER=user -e MYSQL_PASSWORD=password -e MYSQL_HOST=mysqlsec -e MYSQL_PORT=3306 secapp
(http)
2022-11-07 08:46:53.295 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-11-07 08:46:53.295 INFO 1 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.60]
2022-11-07 08:46:53.393 INFO 1 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2022-11-07 08:46:53.393 INFO 1 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2512 ms
2022-11-07 08:46:54.035 INFO 1 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2022-11-07 08:46:54.118 INFO 1 --- [main] org.hibernate.Version : HHH000412: Hibernate ORM core version 5.6.7.Final
2022-11-07 08:46:54.428 INFO 1 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
2022-11-07 08:46:54.577 INFO 1 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2022-11-07 08:46:55.018 INFO 1 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2022-11-07 08:46:55.042 INFO 1 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQL5InnoDBDialect
2022-11-07 08:46:55.827 INFO 1 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2022-11-07 08:46:55.842 INFO 1 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2022-11-07 08:46:55.939 WARN 1 --- [main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed during view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
2022-11-07 08:46:57.013 INFO 1 --- [main] o.s.s.web.DefaultSecurityFilterChain : Will not secure any request
2022-11-07 08:46:57.552 INFO 1 --- [main] o.s.b.a.w.s.WelcomePageHandlerMapping : Adding welcome page template: index
2022-11-07 08:46:58.413 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8076 (http) with context path ''
2022-11-07 08:46:58.429 INFO 1 --- [main] c.s.d.SpringSecurityLoginOutApplication : Started SpringSecurityLoginOutApplication in 8.23 seconds (JVM running for 9.082)
```

Your application starts running now from docker and Open your browser and type

localhost:9092



RESULT

