

CI/CD Spring Boot Application Project Source Code

Version History:

Author:	Shanmugam D
Purpose:	Project Source Code and GITHUB
Date:	27 th March 2022.
Version:	1.0

Table of Content:

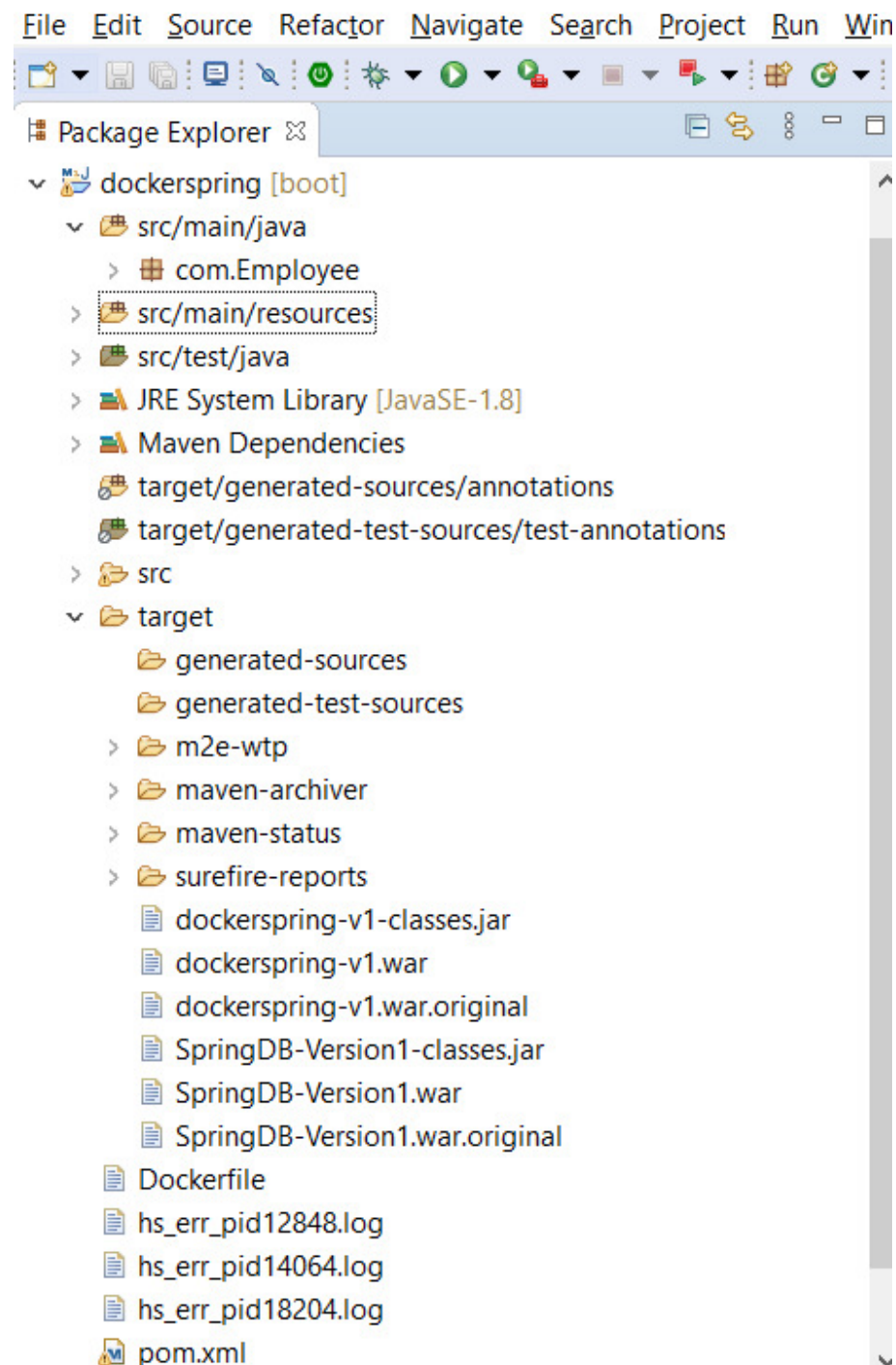
Project GITHUB Link:	3
Project Code:	4

Project GITHUB Link:

Repository Name:	CICD-SpringbootApplication
Github Link:	https://github.com/dshaan05/CICD-SpringbootApplication

Project Code:

Folder Structure



[Empcontroller.java](#)

```
package com.Employee;
```

```

import java.util.List;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

@Controller
public class Empcontroller {

    @Autowired
    EmpDAO dao;

    @RequestMapping("insert")
    public ModelAndView insert(@RequestParam("empno") int
empno,@RequestParam("empname") String empname,HttpServletRequest
req,HttpServletResponse res)
    {
        ModelAndView mv=new ModelAndView();
        Employee e=new Employee();
        e.setEmpno(empno);
        e.setEmpname(empname);
        Employee e1=dao.insert(e);
        if(e1!=null) {
            mv.setViewName("status");
        }
        return mv;
    }

    @RequestMapping("get")
    public ModelAndView get(HttpServletRequest req,HttpServletResponse res) {
        ModelAndView mv=new ModelAndView();
        List<Employee> e=dao.getall();
        mv.setViewName("display");
        mv.addObject("list", e);
        return mv;
    }
}

```

[EmpDAO.java](#)

```
package com.Employee;
```

```
import java.util.List;
```

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
```

```
@Service
public class EmpDAO {

    @Autowired
    EmRep empr;
    public Employee insert(Employee e) {
        return empr.save(e);
    }

    public List<Employee> getall(){
        return empr.findAll();
    }

}
```

Employee.java

```
package com.Employee;

import javax.persistence.Entity;
import javax.persistence.Id;

import lombok.Data;

@Data
@Entity
public class Employee {
    @Id
    int empno;
    String empname;
}
```

EmpRep.java

```
package com.Employee;

import org.springframework.data.jpa.repository.JpaRepository;

public interface EmRep extends JpaRepository<Employee,Integer> {

}
```

ServletInitializer.java

```
package com.Employee;

import org.springframework.boot.builder.SpringApplicationBuilder;
import org.springframework.boot.web.servlet.support.SpringBootServletInitializer;

public class ServletInitializer extends SpringBootServletInitializer {
```

```
@Override
protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {
    return application.sources(SpringDbApplication.class);
}
```

```
}
```

SpringDbApplication.java

```
package com.Employee;
```

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
public class SpringDbApplication {
```

```
    public static void main(String[] args) {
        SpringApplication.run(SpringDbApplication.class, args);
    }
```

```
}
```

application.properties

```
server.port=8080
#spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
#spring.datasource.url=jdbc:mysql://localhost:3306/test
spring.datasource.url=jdbc:mysql://docker-mysql:3306/test
spring.datasource.username=root
spring.datasource.password=Shaan@1015
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect
spring.mvc.view.prefix=/pages/
spring.mvc.view.suffix=.jsp
```

```
}
```

SpringDbApplicationTests.java

```
package com.Employee;
```

```
import org.junit.jupiter.api.Test;
import org.springframework.boot.test.context.SpringBootTest;
```

```
@SpringBootTest
class SpringDbApplicationTests {
```

```
    @Test
    void contextLoads() {
    }
```

```
}
```

Dockerfile

```
From openjdk:8
```

Expose 8090

Add sourceCode/cicdPrototype/cicdProj/target/dockerspring-v1.war dockerspring-v1.war

ENTRYPOINT ["java","-jar","/dockerspring-v1.war"]

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.6.0</version>
    <relativePath /> <!-- lookup parent from repository -->
  </parent>
  <groupId>com.example</groupId>
  <artifactId>dockerspring</artifactId>
  <version>v1</version>
  <packaging>war</packaging>
  <name>spring db</name>
  <description>Demo project for Spring Boot</description>
  <properties>
    <java.version>1.8</java.version>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-data-jpa</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <scope>runtime</scope>
    </dependency>
    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <optional>true</optional>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.apache.tomcat/tomcat-jasper -->
    <dependency>
      <groupId>org.apache.tomcat</groupId>
      <artifactId>tomcat-jasper</artifactId>
      <version>9.0.55</version>
    </dependency>

    <dependency>
```



```

        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-tomcat</artifactId>
        <scope>provided</scope>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
            <configuration>
                <excludes>
                    <exclude>
                        <groupId>org.projectlombok</groupId>
                        <artifactId>lombok</artifactId>
                    </exclude>
                </excludes>
            </configuration>
        </plugin>
        <plugin>
            <artifactId>maven-war-plugin</artifactId>
            <configuration>
                <attachClasses>true</attachClasses>
                <!-- <webXml>target/web.xml</webXml> -->
                <webResources>
                    <resource>
                        <directory>src/main/webapp</directory>
                        <filtering>true</filtering>
                    </resource>
                </webResources>
            </configuration>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>3.0.0-M5</version>
            <configuration>
                <skipTests>true</skipTests>
            </configuration>
        </plugin>
    </plugins>
</build>
</project>

```