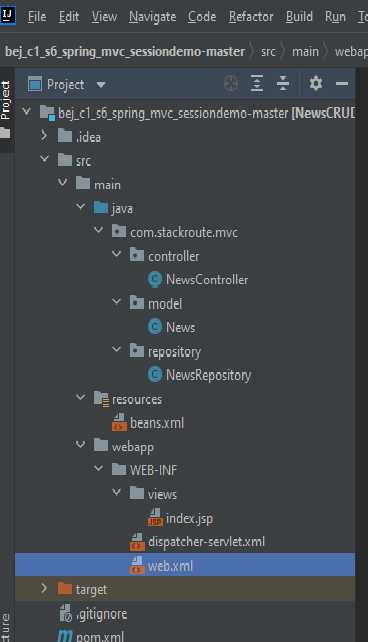
Demo

1. Folder structure:



2)Web.xml

"-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"  
 "http://java.sun.com/dtd/web-app\_2\_3.dtd" >  
  
<web-app>  
 <display-name>Archetype Created Web Application</display-name>  
   
 <servlet>  
 <servlet-name>dispatcher</servlet-name>  
 <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>  
 <load-on-startup>1</load-on-startup>  
 </servlet>  
  
 <servlet-mapping>  
 <servlet-name>dispatcher</servlet-name>  
 <url-pattern>/</url-pattern>  
 </servlet-mapping>  
   
</web-app>

3)dispatcher-servlet.xml

<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xsi:schemaLocation="  
 http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context.xsd">  
  
  
 <context:component-scan  
 base-package="com.stackroute.mvc.controller" />  
  
  
 <bean id="jspViewResolver"  
 class="org.springframework.web.servlet.view.InternalResourceViewResolver">  
  
 <property name="viewClass"  
 value="org.springframework.web.servlet.view.JstlView" />  
  
 <property name="prefix" value="/WEB-INF/views/" />  
  
 <property name="suffix" value=".jsp" />  
 </bean>  
  
</beans>

3)beans.xml

<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xsi:schemaLocation="  
 http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context.xsd">  
  
 <bean id="news" class="com.stackroute.mvc.model.News" scope="prototype"></bean>  
   
 <bean id="newsRepository" class="com.stackroute.mvc.repository.NewsRepository" scope="prototype"></bean>  
  
</beans>

NewsController.java

package com.stackroute.mvc.controller;  
  
import com.stackroute.mvc.repository.NewsRepository;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
import org.springframework.stereotype.Controller;  
import org.springframework.ui.ModelMap;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.ModelAttribute;  
import org.springframework.web.bind.annotation.PostMapping;  
  
import com.stackroute.mvc.model.News;  
  
@Controller  
public class NewsController   
{  
  
 ApplicationContext context = new ClassPathXmlApplicationContext("beans.xml");  
   
 News news = context.getBean("news",News.class);  
 NewsRepository newsRepository = context.getBean("newsRepository",NewsRepository.class);  
   
   
 public NewsController()  
 {  
 System.*out*.println("Employee Controller Instantiated ...");  
 }  
   
 @GetMapping(path="/")  
 public String getNewsHomePage()  
 {  
 System.*out*.println("Taking to home Page");  
 return "index";  
 }  
   
 @PostMapping("/saveNews")  
 public String saveNews(@ModelAttribute("news") News news,ModelMap model)  
 {  
 System.*out*.println("Taking to the save News Page");  
 System.*out*.println("Taking to the save News Page ...."+news);  
   
 this.newsRepository.addNews(news);  
 model.addAttribute("newsList",this.newsRepository.getNewsData());  
   
 return "index";  
 }  
}

News.java -🡪model

package com.stackroute.mvc.model;  
  
public class News   
{  
 private int newsId;  
 private String newsTitle;  
 private String newsText;  
 private String newsCategory;  
 public int getNewsId() {  
 return newsId;  
 }  
 public void setNewsId(int newsId) {  
 this.newsId = newsId;  
 }  
 public String getNewsTitle() {  
 return newsTitle;  
 }  
   
 public void setNewsTitle(String newsTitle) {  
 this.newsTitle = newsTitle;  
 }  
   
 public String getNewsText() {  
 return newsText;  
 }  
   
 public void setNewsText(String newsText) {  
 this.newsText = newsText;  
 }  
   
 public String getNewsCategory() {  
 return newsCategory;  
 }  
   
 public void setNewsCategory(String newsCategory) {  
 this.newsCategory = newsCategory;  
 }  
   
 @Override  
 public String toString() {  
 return "News [newsId=" + newsId + ", newsTitle=" + newsTitle + ", newsText=" + newsText + ", newsCategory="  
 + newsCategory + "]";  
 }  
}

NewsRepositary.java

package com.stackroute.mvc.model;  
  
public class News   
{  
 private int newsId;  
 private String newsTitle;  
 private String newsText;  
 private String newsCategory;  
 public int getNewsId() {  
 return newsId;  
 }  
 public void setNewsId(int newsId) {  
 this.newsId = newsId;  
 }  
 public String getNewsTitle() {  
 return newsTitle;  
 }  
   
 public void setNewsTitle(String newsTitle) {  
 this.newsTitle = newsTitle;  
 }  
   
 public String getNewsText() {  
 return newsText;  
 }  
   
 public void setNewsText(String newsText) {  
 this.newsText = newsText;  
 }  
   
 public String getNewsCategory() {  
 return newsCategory;  
 }  
   
 public void setNewsCategory(String newsCategory) {  
 this.newsCategory = newsCategory;  
 }  
   
 @Override  
 public String toString() {  
 return "News [newsId=" + newsId + ", newsTitle=" + newsTitle + ", newsText=" + newsText + ", newsCategory="  
 + newsCategory + "]";  
 }  
}

then you need to associate your project weapp directory with tomcat

and start the tomcat

dependency file pom.xml

<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 http://maven.apache.org/maven-v4\_0\_0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
 <groupId>com.stackroute.spring.mvc</groupId>  
 <artifactId>NewsCRUD</artifactId>  
 <packaging>war</packaging>  
 <version>0.0.1-SNAPSHOT</version>  
 <name>NewsCRUD Maven Webapp</name>  
 <url>http://maven.apache.org</url>  
  
 <properties>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 <maven.compiler.source>11</maven.compiler.source>  
 <maven.compiler.target>11</maven.compiler.target>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>javax.servlet</groupId>  
 <artifactId>javax.servlet-api</artifactId>  
 <version>4.0.1</version>  
 <scope>provided</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-webmvc</artifactId>  
 <version>5.3.1</version>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-orm</artifactId>  
 <version>5.3.1</version>  
 </dependency>  
  
 <!-- <dependency>-->  
 <!-- <groupId>org.springframework</groupId>-->  
 <!-- <artifactId>spring-jdbc</artifactId>-->  
 <!-- <version>5.3.1</version>-->  
 <!-- </dependency>-->  
  
  
 <dependency>  
 <groupId>jstl</groupId>  
 <artifactId>jstl</artifactId>  
 <version>1.2</version>  
 </dependency>  
  
 <dependency>  
 <groupId>mysql</groupId>  
 <artifactId>mysql-connector-java</artifactId>  
 <version>5.1.6</version>  
 </dependency>  
  
 <dependency>  
 <groupId>org.hibernate</groupId>  
 <artifactId>hibernate-core</artifactId>  
 <version>5.4.25.Final</version>  
 </dependency>  
 <!--Test dependencies-->  
  
 <dependency>  
 <groupId>org.mockito</groupId>  
 <artifactId>mockito-all</artifactId>  
 <version>1.10.19</version>  
 <scope>test</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.mockito</groupId>  
 <artifactId>mockito-core</artifactId>  
 <version>3.2.4</version>  
 <scope>test</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.junit.platform</groupId>  
 <artifactId>junit-platform-launcher</artifactId>  
 <version>1.1.0</version>  
 <scope>test</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-test</artifactId>  
 <version>5.2.2.RELEASE</version>  
 </dependency>  
  
 <dependency>  
 <groupId>org.junit.platform</groupId>  
 <artifactId>junit-platform-surefire-provider</artifactId>  
 <version>1.1.0</version>  
 </dependency>  
  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter-engine</artifactId>  
 <version>5.1.0</version>  
 </dependency>  
 </dependencies>  
  
  
 <build>  
 <finalName>springhibernate</finalName>  
 <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->  
 <plugins>  
 <plugin>  
 <artifactId>maven-clean-plugin</artifactId>  
 <version>3.1.0</version>  
 </plugin>  
 <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_war\_packaging -->  
 <plugin>  
 <artifactId>maven-resources-plugin</artifactId>  
 <version>3.0.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-compiler-plugin</artifactId>  
 <version>3.8.0</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-surefire-plugin</artifactId>  
 <version>2.22.1</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-war-plugin</artifactId>  
 <version>3.2.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-install-plugin</artifactId>  
 <version>2.5.2</version>  
 </plugin>  
 <plugin>  
 <artifactId>maven-deploy-plugin</artifactId>  
 <version>2.8.2</version>  
 </plugin>  
 </plugins>  
 </pluginManagement>  
 </build>  
</project>