Source Code:

1. **package** com.javatpoint.beans,.class Emp:

**package** com.javatpoint.beans;

**public** **class** Emp {

**private** **int** id;

**private** String name;

**private** **float** salary;

**private** String designation;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **float** getSalary() {

**return** salary;

}

**public** **void** setSalary(**float** salary) {

**this**.salary = salary;

}

**public** String getDesignation() {

**return** designation;

}

**public** **void** setDesignation(String designation) {

**this**.designation = designation;

}

}

2.package com.javatpoint.controllers, class EmpController

package com.javatpoint.controllers;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import com.javatpoint.beans.Emp;

import com.javatpoint.dao.EmpDao;

@Controller

public class EmpController {

@Autowired

EmpDao dao;//will inject dao from xml file

/\*It displays a form to input data, here "command" is a reserved request attribute

\*which is used to display object data into form

\*/

@RequestMapping("/empform")

public String showform(Model m){

m.addAttribute("command", new Emp());

return "empform";

}

/\*It saves object into database. The @ModelAttribute puts request data

\* into model object. You need to mention RequestMethod.POST method

\* because default request is GET\*/

@RequestMapping(value="/save",method = RequestMethod.POST)

public String save(@ModelAttribute("emp") Emp emp){

dao.save(emp);

return "redirect:/viewemp";//will redirect to viewemp request mapping

}

/\* It provides list of employees in model object \*/

@RequestMapping("/viewemp")

public String viewemp(Model m){

List<Emp> list=dao.getEmployees();

m.addAttribute("list",list);

return "viewemp";

}

/\* It displays object data into form for the given id.

\* The @PathVariable puts URL data into variable.\*/

@RequestMapping(value="/editemp/{id}")

public String edit(@PathVariable int id, Model m){

Emp emp=dao.getEmpById(id);

m.addAttribute("command",emp);

return "empeditform";

}

/\* It updates model object. \*/

@RequestMapping(value="/editsave",method = RequestMethod.POST)

public String editsave(@ModelAttribute("emp") Emp emp){

dao.update(emp);

return "redirect:/viewemp";

}

/\* It deletes record for the given id in URL and redirects to /viewemp \*/

@RequestMapping(value="/deleteemp/{id}",method = RequestMethod.GET)

public String delete(@PathVariable int id){

dao.delete(id);

return "redirect:/viewemp";

}

}

3.package com.javatpoint.dao , Class EmpDao

package com.javatpoint.dao;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.List;

import org.springframework.jdbc.core.BeanPropertyRowMapper;

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.jdbc.core.RowMapper;

import com.javatpoint.beans.Emp;

public class EmpDao {

JdbcTemplate template;

public void setTemplate(JdbcTemplate template) {

this.template = template;

}

public int save(Emp p){

String sql="insert into Emp99(name,salary,designation) values('"+p.getName()+"',"+p.getSalary()+",'"+p.getDesignation()+"')";

return template.update(sql);

}

public int update(Emp p){

String sql="update Emp99 set name='"+p.getName()+"', salary="+p.getSalary()+",designation='"+p.getDesignation()+"' where id="+p.getId()+"";

return template.update(sql);

}

public int delete(int id){

String sql="delete from Emp99 where id="+id+"";

return template.update(sql);

}

public Emp getEmpById(int id){

String sql="select \* from Emp99 where id=?";

return template.queryForObject(sql, new Object[]{id},new BeanPropertyRowMapper<Emp>(Emp.class));

}

public List<Emp> getEmployees(){

return template.query("select \* from Emp99",new RowMapper<Emp>(){

public Emp mapRow(ResultSet rs, int row) throws SQLException {

Emp e=new Emp();

e.setId(rs.getInt(1));

e.setName(rs.getString(2));

e.setSalary(rs.getFloat(3));

e.setDesignation(rs.getString(4));

return e;

}

});

}

}

4.spring-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-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"*>

<context:component-scan base-package=*"com.javatpoint.controllers"*></context:component-scan>

<bean class=*"org.springframework.web.servlet.view.InternalResourceViewResolver"*>

<property name=*"prefix"* value=*"/WEB-INF/jsp/"*></property>

<property name=*"suffix"* value=*".jsp"*></property>

</bean>

<bean id=*"ds"* class=*"org.springframework.jdbc.datasource.DriverManagerDataSource"*>

<property name=*"driverClassName"* value=*"com.mysql.jdbc.Driver"*></property>

<property name=*"url"* value=*"jdbc:mysql://localhost:3306/test"*></property>

<property name=*"username"* value=*"root"*></property>

<property name=*"password"* value=*"1432"*></property>

</bean>

<bean id=*"jt"* class=*"org.springframework.jdbc.core.JdbcTemplate"*>

<property name=*"dataSource"* ref=*"ds"*></property>

</bean>

<bean id=*"dao"* class=*"com.javatpoint.dao.EmpDao"*>

<property name=*"template"* ref=*"jt"*></property>

</bean>

</beans>

1. Web.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://java.sun.com/xml/ns/javaee"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"* id=*"WebApp\_ID"* version=*"3.0"*>

<display-name>SpringMVC</display-name>

<servlet>

<servlet-name>spring</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>spring</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>

6.inde.jsp

<a href=*"empform"*>Add Employee</a>

<a href=*"viewemp"*>View Employees</a>

7. 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 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com</groupId>

<artifactId>SpringSetup</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>SpringMVCPagination Maven Webapp</name>

<url>http://maven.apache.org</url>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>3.8.1</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.1.1.RELEASE</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.tomcat/tomcat-jasper -->

<dependency>

<groupId>org.apache.tomcat</groupId>

<artifactId>tomcat-jasper</artifactId>

<version>9.0.12</version>

</dependency>

<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>servlet-api</artifactId>

<version>3.0-alpha-1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>jstl</artifactId>

<version>1.2</version>

</dependency>

<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.11</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-jdbc -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-jdbc</artifactId>

<version>5.1.1.RELEASE</version>

</dependency>

</dependencies>

<build>

<finalName>SpringMVCPagination</finalName>

</build>

</project>