**Searching for a Specific User and Updating the User Information**

**Java file**

MainController



package usercrud.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

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 org.springframework.web.servlet.view.RedirectView;

import usercrud.dao.UserDao;

import usercrud.model.User;

@Controller

public class MainController {

@Autowired

private UserDao userdao;

@RequestMapping("/")

public String home() {

return "index";

}

@RequestMapping("/view")

public String View(Model m) {

List<User> users = userdao.getUsers();

m.addAttribute("users", users);

return "View";

}

@RequestMapping("adduser")

public String addUser() {

return "adduserform";

}

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

public RedirectView user(@ModelAttribute User user, HttpServletRequest request) {

System.out.println(user);

userdao.createUser(user);

RedirectView redirectView = new RedirectView();

redirectView.setUrl(request.getContextPath() + "/");

return redirectView;

}

@RequestMapping("/delete/{id}")

public RedirectView deleteuser(@PathVariable("id") int id, HttpServletRequest request) {

this.userdao.deleteProduct(id);

RedirectView redirectView = new RedirectView();

redirectView.setUrl(request.getContextPath() + "/");

return redirectView;

}

@RequestMapping("/update/{id}")

public String updateform(@PathVariable("id") int id,Model model) {

User user=this.userdao.getUser(id);

model.addAttribute("user", user);

return "updateform";

}

}

UserDao



package usercrud.dao;

import java.util.List;

import javax.transaction.Transactional;

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

import org.springframework.orm.hibernate5.HibernateTemplate;

import org.springframework.stereotype.Component;

import usercrud.model.User;

@Component

public class UserDao {

@Autowired

private HibernateTemplate hibernateTemplate;

// create User

@Transactional

public void createUser(User user) {

this.hibernateTemplate.saveOrUpdate(user);

}

//get all view User

public List<User> getUsers() {

List<User> Users = this.hibernateTemplate.loadAll(User.class);

return Users;

}

// delete the single project

@Transactional

public void deleteProduct(int id) {

User U = this.hibernateTemplate.load(User.class, id);

this.hibernateTemplate.delete(U);

}

//to get the single product

public User getUser(int id) {

return this.hibernateTemplate.get(User.class, id);

}

}

User



package usercrud.model;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class User {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private int id;

private String firstname;

private String lastname;

private String email;

private String gender;

private String date;

public User() {

super();

// TODO Auto-generated constructor stub

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

public String getDate() {

return date;

}

public void setDate(String date) {

this.date = date;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFirstname() {

return firstname;

}

public void setFirstname(String firstname) {

this.firstname = firstname;

}

public String getLastname() {

return lastname;

}

public void setLastname(String lastname) {

this.lastname = lastname;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

@Override

public String toString() {

return "User [id=" + id + ", firstname=" + firstname + ", lastname=" + lastname + ", email=" + email

+ ", gender=" + gender + ", date=" + date + "]";

}

}

jsp files

Index



<html>

<body>

<h2>Hello World!</h2>

<h1>This is my view</h1>

<a href="adduser">add user</a>

<a href="view">view user</a>

</body>

</html>

Adduserform



<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>user details</title>

</head>

<body>

<h1 align="center">Add User Details</h1>

<form action="user" method="post">

<label>First Name :</label> <input type="text" name=firstname

placeholder="Enter the first name"><br> <label>Last

Name :</label> <input type="text" name=lastname

placeholder="Enter the last name"><br> <label>Date

Of Birth :</label> <input type="text" name=date placeholder="dd/mm/yyy"><br>

<label>Gender :</label> <input type="text" required="required" name=gender ><br>

<label>Email :</label> <input type="email" name=email

placeholder="Enter the valdi email"><br>

<button type="submit">Add User</button>

</form>

</body>

</html>

Updateform



<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<%@page isELIgnored="false"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>user details</title>

</head>

<body>

<h1 align="center">change Product Details</h1>

<form action="${pageContext.request.contextPath }/user" method="post">

<label>ID :</label> <input type="number" name=id value="${user.id }"><br>

<label>First Name :</label> <input type="text" name=firstname

placeholder="Enter the first name" value="${user.firstname }"><br>

<label>Last Name :</label> <input type="text" name=lastname

placeholder="Enter the last name" value="${user.lastname }"><br>

<label>Date of Birth</label><input type="text" name="date">

<label>Gender :</label> <input type="text" required="required"

name=gender value="${user.gender }"><br> <label>Email

:</label> <input type="email" name=email placeholder="Enter the valdi email"

value="${user.email }"><br>

<button type="submit">update</button>

</form>

</body>

</html>

View



<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

<%@page isELIgnored="false"%>

<!DOCTYPE html>

<html>

<head>

<style>

table {

font-family: arial, sans-serif;

border-collapse: collapse;

width: 100%;

}

td, th {

border: 1px solid #dddddd;

text-align: left;

padding: 8px;

}

tr:nth-child(even) {

background-color: #dddddd;

}

</style>

<meta charset="UTF-8">

<title>View User Details</title>

</head>

<body>

<h1 align="center">Welcome to User Details Portal</h1>

<table>

<tr>

<th>id</th>

<th>First Name</th>

<th>Last name</th>

<th>Date of birth</th>

<th>Gender</th>

<th>Email</th>

<th>Action</th>

</tr>

<c:forEach items="${users}" var="v">

<tr>

<td>${v.id}</td>

<td>${v.firstname}</td>

<td>${v.lastname}</td>

<td>${v.date}</td>

<td>${v.gender}</td>

<td>${v.email}</td>

<td><a href="delete/${v.id }"><button>Delete</button></a></td>

<td><a href="update/${v.id }"><button>Edit</button></a></td>

</tr>

</c:forEach>

</table>

<a href="adduser" style="align-items: center;"><button>Add

user</button></a>

</body>

</html>

XML files

spring-servlet



<?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:context="http://www.springframework.org/schema/context"

xmlns:p="http://www.springframework.org/schema/p"

xmlns:tx="http://www.springframework.org/schema/tx"

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

http://www.springframework.org/schema/tx

http://www.springframework.org/schema/tx/spring-tx.xsd

">

<tx:annotation-driven />

<context:component-scan base-package="usercrud" />

<bean

class="org.springframework.web.servlet.view.InternalResourceViewResolver"

name="viewResolver">

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

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

</bean>

<!-- data source -->

<bean

class="org.springframework.jdbc.datasource.DriverManagerDataSource"

name="ds">

<property name="driverClassName"

value="com.mysql.jdbc.Driver" />

<property name="url"

value="jdbc:mysql://localhost:3306/userdetails" />

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

<property name="password" value="Harsh@123" />

</bean>

<!-- LocalSessionFactoryBean -->

<bean

class="org.springframework.orm.hibernate5.LocalSessionFactoryBean"

name="factory">

<!-- data source -->

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

<!-- hibernate properties -->

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">org.hibernate.dialect.MySQL5Dialect</prop>

<prop key="hibernate.show\_sql">true</prop>

<prop key="hibernate.hbm2ddl.auto">update</prop>

</props>

</property>

<!-- annotated classes -->

<property name="annotatedClasses">

<list>

<value>usercrud.model.User</value>

</list>

</property>

</bean>

<bean class="org.springframework.orm.hibernate5.HibernateTemplate"

name="hibernateTemplate">

<property name="sessionFactory" ref="factory"></property>

</bean>

<bean

class="org.springframework.orm.hibernate5.HibernateTransactionManager"

name="transactionManager">

<property name="sessionFactory" ref="factory"></property>

</bean>

</beans>

Web



<!DOCTYPE web-app PUBLIC

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

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

<load-on-startup>2</load-on-startup>

</servlet>

<servlet-mapping>

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

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

</servlet-mapping>

</web-app>

Pom



<?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.usercrud</groupId>

<artifactId>UserCurd</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>UserCurd Maven Webapp</name>

<!-- FIXME change it to the project's website -->

<url>http://www.example.com</url>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.source>1.7</maven.compiler.source>

<maven.compiler.target>1.7</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.11</version>

<scope>test</scope>

</dependency>

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

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.22</version>

</dependency>

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

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>5.3.22</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.hibernate.orm/hibernate-core -->

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.4.2.Final</version>

</dependency>

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

<dependency>

<groupId>mysql</groupId>

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

<version>8.0.28</version>

</dependency>

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

<dependency>

<groupId>jstl</groupId>

<artifactId>jstl</artifactId>

<version>1.2</version>

</dependency>

</dependencies>

<build>

<finalName>UserCurd</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>