In controller ‘;

/\*\*

\*

\*/

package com.virtusa.controller;

import java.util.Map;

import javax.validation.Valid;

import org.springframework.stereotype.Controller;

import org.springframework.validation.BindingResult;

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

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

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

import com.virtusa.model.MyUser;

/\*\*

\* @author padarbindas

\*

\*/

@Controller

public class MyController {

@RequestMapping(value = "/")

public String home() {

return "index";

}

@ModelAttribute("myUser")

public MyUser getLoginForm() {

return new MyUser();

}

@RequestMapping(value = "/myForm", method = RequestMethod.GET)

public String showForm(Map model) {

return "myForm";

}

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

public String validateForm(

@ModelAttribute("myUser") @Valid MyUser myUser,

BindingResult result, Map model) {

if (result.hasErrors()) {

return "myForm";

}

model.put("myUser", myUser);

return "success";

}

}

In model:

/\*\*

\*

\*/

package com.virtusa.model;

import javax.validation.constraints.Max;

import javax.validation.constraints.Min;

import javax.validation.constraints.NotNull;

import javax.validation.constraints.Size;

/\*\*

\* @author padarbindas

\*

\*/

public class MyUser {

@NotNull

@Size(min=1,max=20)

private String name;

@Min(0)

@Max(120)

private int age;

public MyUser(String name, int age) {

this.name = name;

this.age = age;

}

public MyUser() {

name = "";

age = 0;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

}

Index.jsp:

<%@page contentType='text/html' pageEncoding='UTF-8'%>

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

<!doctype html>

<html lang='en'>

<head>

<title>Welcome !!!</title>

</head>

<body>

<h1>

Spring Form Validation !!!

</h1>

<a href='<c:url value='/myForm'/>'>Go to the form!</a>

</body>

</html>

In myform.jsp:

<%@page contentType='text/html' pageEncoding='UTF-8'%>

<%@ taglib prefix='form' uri='http://www.springframework.org/tags/form' %>

<!doctype html>

<html>

<head>

<meta http-equiv='Content-Type' content='text/html; charset=UTF-8'>

<title>My User Form!</title>

</head>

<body>

<form:form method='post' action='myForm' commandName='myUser'>

<table>

<tr>

<td>Name: <font color='red'><form:errors path='name' /></font></td>

</tr>

<tr>

<td><form:input path='name' /></td>

</tr>

<tr>

<td>Age: <font color='red'><form:errors path='age' /></font></td>

</tr>

<tr>

<td><form:input path='age' /></td>

</tr>

<tr>

<td><input type='submit' value='Submit' /></td>

</tr>

</table>

</form:form>

</body>

</html>

In controller:

package com.virtusa.contact.controller;

import java.util.Map;

import com.virtusa.contact.domain.Contact;

import com.virtusa.contact.service.ContactService;

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

import org.springframework.stereotype.Controller;

import org.springframework.validation.BindingResult;

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;

@Controller

public class ContactController {

@Autowired

private ContactService contactService;

@RequestMapping("/index")

public String listContacts(Map<String, Object> map) {

map.put("contact", new Contact());

map.put("contactList", contactService.listContact());

return "contact";

}

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

public String addContact(@ModelAttribute("contact")

Contact contact, BindingResult result) {

contactService.addContact(contact);

return "redirect:/index";

}

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

public String deleteContact(@PathVariable("contactId")

Integer contactId) {

contactService.removeContact(contactId);

return "redirect:/index";

}

}

In dao:

package com.virtusa.contact.dao;

import java.util.List;

import com.virtusa.contact.domain.Contact;

public interface ContactDAO {

public void addContact(Contact contact);

public List<Contact> listContact();

public void removeContact(Integer id);

}

package com.virtusa.contact.dao;

import java.util.List;

import com.virtusa.contact.domain.Contact;

import org.hibernate.SessionFactory;

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

import org.springframework.stereotype.Repository;

@Repository

public class ContactDAOImpl implements ContactDAO {

@Autowired

private SessionFactory sessionFactory;

public void addContact(Contact contact) {

sessionFactory.getCurrentSession().save(contact);

}

public List<Contact> listContact() {

return sessionFactory.getCurrentSession()

.createQuery("from Contact")

.list();

}

public void removeContact(Integer id) {

Contact contact = (Contact)

sessionFactory.getCurrentSession().load(

Contact.class, id);

if (null != contact) {

sessionFactory.getCurrentSession().delete(contact);

}

}

}

In domain:

package com.virtusa.contact.domain;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="CONTACTS")

public class Contact {

@Id

@Column(name="ID")

@GeneratedValue

private Integer id;

@Column(name="FIRSTNAME")

private String firstname;

@Column(name="LASTNAME")

private String lastname;

@Column(name="EMAIL")

private String email;

@Column(name="TELEPHONE")

private String telephone;

public String getEmail() {

return email;

}

public String getTelephone() {

return telephone;

}

public void setEmail(String email) {

this.email = email;

}

public void setTelephone(String telephone) {

this.telephone = telephone;

}

public String getFirstname() {

return firstname;

}

public String getLastname() {

return lastname;

}

public void setFirstname(String firstname) {

this.firstname = firstname;

}

public void setLastname(String lastname) {

this.lastname = lastname;

}

public Integer getId() {

return id;

}

public void setId(Integer id) {

this.id = id;

}

}

In service:

package com.virtusa.contact.service;

import java.util.List;

import com.virtusa.contact.domain.Contact;

public interface ContactService {

public void addContact(Contact contact);

public List<Contact> listContact();

public void removeContact(Integer id);

}

package com.virtusa.contact.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.virtusa.contact.dao.ContactDAO;

import com.virtusa.contact.domain.Contact;

@Service

public class ContactServiceImpl implements ContactService {

@Autowired

private ContactDAO contactDAO;

@Transactional

public void addContact(Contact contact) {

contactDAO.addContact(contact);

}

@Transactional

public List<Contact> listContact() {

return contactDAO.listContact();

}

@Transactional

public void removeContact(Integer id) {

contactDAO.removeContact(id);

}

}

In index.jsp:

<jsp:forward page="index"></jsp:forward>

In springservle.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:aop="http://www.springframework.org/schema/aop"

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

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

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

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

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

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

xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd

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

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

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

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

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

http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util.xsd">

<context:annotation-config/>

<context:component-scan base-package="com.virtusa.contact" />

<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/jsp/" />

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

</bean>

<bean id="messageSource"

class="org.springframework.context.support.ReloadableResourceBundleMessageSource">

<property name="basename" value="classpath:messages" />

<property name="defaultEncoding" value="UTF-8" />

</bean>

<bean id="propertyConfigurer"

class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer"

p:location="/WEB-INF/jdbc.properties" />

<bean id="dataSource"

class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close"

p:driverClassName="${jdbc.driverClassName}"

p:url="${jdbc.databaseurl}" p:username="${jdbc.username}"

p:password="${jdbc.password}" />

<bean id="sessionFactory"

class="org.springframework.orm.hibernate3.LocalSessionFactoryBean">

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

<property name="configLocation">

<value>classpath:hibernate.cfg.xml</value>

</property>

<property name="configurationClass">

<value>org.hibernate.cfg.AnnotationConfiguration</value>

</property>

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">${jdbc.dialect}</prop>

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

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

</props>

</property>

</bean>

<tx:annotation-driven />

<bean id="transactionManager"

class="org.springframework.orm.hibernate3.HibernateTransactionManager">

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

</bean>

</beans>

In contact.jsp:

<%@taglib uri="http://www.springframework.org/tags" prefix="spring"%>

<%@taglib uri="http://www.springframework.org/tags/form" prefix="form"%>

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

<html>

<head>

<title>Spring 3 MVC Series - Contact Manager</title>

<style type="text/css">

body {

font-family: sans-serif;

}

.data, .data td {

border-collapse: collapse;

width: 100%;

border: 1px solid #aaa;

margin: 2px;

padding: 2px;

}

.data th {

font-weight: bold;

background-color: #5C82FF;

color: white;

}

</style>

</head>

<body>

<h2>Contact Manager</h2>

<form:form method="post" action="add.html" commandName="contact">

<table>

<tr>

<td><form:label path="firstname"><spring:message code="label.firstname"/></form:label></td>

<td><form:input path="firstname" /></td>

</tr>

<tr>

<td><form:label path="lastname"><spring:message code="label.lastname"/></form:label></td>

<td><form:input path="lastname" /></td>

</tr>

<tr>

<td><form:label path="email"><spring:message code="label.email"/></form:label></td>

<td><form:input path="email" /></td>

</tr>

<tr>

<td><form:label path="telephone"><spring:message code="label.telephone"/></form:label></td>

<td><form:input path="telephone" /></td>

</tr>

<tr>

<td colspan="2">

<input type="submit" value="<spring:message code="label.addcontact"/>"/>

</td>

</tr>

</table>

</form:form>

<h3>Contacts</h3>

<c:if test="${!empty contactList}">

<table class="data">

<tr>

<th>Name</th>

<th>Email</th>

<th>Telephone</th>

<th>&nbsp;</th>

</tr>

<c:forEach items="${contactList}" var="contact">

<tr>

<td>${contact.lastname}, ${contact.firstname} </td>

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

<td>${contact.telephone}</td>

<td><a href="delete/${contact.id}">delete</a></td>

</tr>

</c:forEach>

</table>

</c:if>

</body>

</html>

User login:

Servlet:

package com.virtusa.controller;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import com.virtusa.dao.UserDAO;

import com.virtusa.user.UserBean;

/\*\*

\*

\* @author padarbindas

\*

\*/

public class LoginServlet extends HttpServlet {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, java.io.IOException {

try {

UserBean user = new UserBean();

user.setUserName(request.getParameter("uname"));

user.setPassword(request.getParameter("upass"));

user = UserDAO.login(user);

if (user.isValid()) {

HttpSession session = request.getSession(true);

session.setAttribute("currentSessionUser", user);

//response.sendRedirect("userLogged.jsp"); // logged-in page

request.getRequestDispatcher("userLogged.jsp").forward(request, response);

}

else

//response.sendRedirect("invalidLogin.jsp"); // error page

request.getRequestDispatcher("invalidLogin.jsp").forward(request, response);

}

catch (Throwable theException) {

System.out.println(theException);

}

}

}

In userdao:

package com.virtusa.dao;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.Statement;

import com.virtusa.user.UserBean;

import com.virtusa.util.ConnectionManager;

/\*\*

\*

\* @author padarbindas

\*

\*/

public class UserDAO {

static Connection currentCon = null;

static ResultSet rs = null;

public static UserBean login(UserBean bean) {

// preparing some objects for connection

Statement stmt = null;

String username = bean.getUsername();

String password = bean.getPassword();

String searchQuery = "select \* from Users where username='" + username

+ "' AND password='" + password + "'";

// "System.out.println" prints in the console; Normally used to trace

// the process

System.out.println("Your user name is " + username);

System.out.println("Your password is " + password);

System.out.println("Query: " + searchQuery);

try {

// connect to DB

currentCon = ConnectionManager.getConnection();

stmt = currentCon.createStatement();

rs = stmt.executeQuery(searchQuery);

boolean more = rs.next();

// if user does not exist set the isValid variable to false

if (!more) {

System.out

.println("Sorry, you are not a registered user! Please sign up first");

bean.setValid(false);

}

// if user exists set the isValid variable to true

else if (more) {

String firstName = rs.getString("FirstName");

String lastName = rs.getString("LastName");

System.out.println("Welcome " + firstName);

bean.setFirstName(firstName);

bean.setLastName(lastName);

bean.setValid(true);

}

}

catch (Exception ex) {

System.out.println("Log In failed: An Exception has occurred! "

+ ex);

}

// some exception handling

finally {

if (rs != null) {

try {

rs.close();

} catch (Exception e) {

}

rs = null;

}

if (stmt != null) {

try {

stmt.close();

} catch (Exception e) {

}

stmt = null;

}

if (currentCon != null) {

try {

currentCon.close();

} catch (Exception e) {

}

currentCon = null;

}

}

return bean;

}

}

In userbean:

package com.virtusa.user;

/\*\*

\*

\* @author padarbindas

\*

\*/

public class UserBean {

private String username;

private String password;

private String firstName;

private String lastName;

public boolean valid;

public String getFirstName() {

return firstName;

}

public void setFirstName(String newFirstName) {

firstName = newFirstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String newLastName) {

lastName = newLastName;

}

public String getPassword() {

return password;

}

public void setPassword(String newPassword) {

password = newPassword;

}

public String getUsername() {

return username;

}

public void setUserName(String newUsername) {

username = newUsername;

}

public boolean isValid() {

return valid;

}

public void setValid(boolean newValid) {

valid = newValid;

}

}

By hybernet:

Controller:

package com.virtusa.controller;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletConfig;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.hibernate.HibernateException;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;

import org.hibernate.classic.Session;

import com.virtusa.model.User;

import com.virtusa.service.IUserService;

import com.virtusa.service.UserImpl;

/\*\*

\* Servlet implementation class AddingUserServlet

\*/

public class AddingUserServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

private static SessionFactory sessionFactory = null;

private Session session = null;

private Transaction transation = null;

// private User user = null;

public AddingUserServlet() {

// user =new User();

}

/\*\*

\* @see Servlet#init(ServletConfig)

\*/

public void init(ServletConfig config) throws ServletException {

/\* try{

sessionFactory = new Configuration()

.configure().buildSessionFactory();

}

catch(HibernateException he){

he.printStackTrace();

throw new ExceptionInInitializerError();

}

}

public static SessionFactory getSessionFactory(){

return sessionFactory;

}\*/

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// TODO Auto-generated method stub

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

PrintWriter out = response.getWriter();

response.setContentType("text/html");

String userName = request.getParameter("userName");

String password = request.getParameter("password");

String firstName = request.getParameter("firstName");

String lastName = request.getParameter("lastName");

User user = new User(userName,password,firstName,lastName);

IUserService service = new UserImpl();

service.addUser(user);

/\* session = getSessionFactory().openSession();

transation = session.beginTransaction();

user.setFirstName(userName);

user.setPassword(password);

user.setUserName(firstName);

user.setLastName(lastName);

session.save(user);

out.println("got");

transation.commit();\*/

}

/\*\*

\* @see Servlet#destroy()

\*/

public void destroy() {

session.close();

}

}

Dao:

package com.virtusa.DAO;

import org.hibernate.HibernateException;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;

import com.virtusa.model.User;

public class UserDAO {

private static SessionFactory sessionFactory = null;

private Session session;

private Transaction transaction;

public UserDAO() {

try{

sessionFactory = new Configuration()

.configure().buildSessionFactory();

}

catch(HibernateException he){

he.printStackTrace();

throw new ExceptionInInitializerError();

}

}

public static SessionFactory getSessionFactory(){

return sessionFactory;

}

public void addUser(User user) {

session = getSessionFactory().openSession();

transaction = session.beginTransaction();

user.setFirstName(user.getUserName());

user.setPassword(user.getPassword());

user.setUserName(user.getFirstName());

user.setLastName(user.getLastName());

session.save(user);

System.out.println("got");

transaction.commit();

}

}

Model:

package com.virtusa.model;

import javax.persistence.Column;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

import javax.persistence.Entity;

@Entity

@Table(name ="UserDetails")

public class User {

@Id

@Column(name ="userId")

@GeneratedValue(strategy = GenerationType.AUTO)

private int userId;

@Column(name ="userName")

private String userName;

@Column(name ="password")

private String password;

@Column(name = "firstname")

private String firstName;

@Column(name = "lastName")

private String lastName;

public User() {

}

public User(String userName, String password, String firstName,

String lastName) {

super();

this.userName = userName;

this.password = password;

this.firstName = firstName;

this.lastName = lastName;

}

public User(int userId, String userName, String password, String firstName,

String lastName) {

super();

this.userId = userId;

this.userName = userName;

this.password = password;

this.firstName = firstName;

this.lastName = lastName;

}

public int getUserId() {

return userId;

}

public void setUserId(int userId) {

this.userId = userId;

}

public String getUserName() {

return userName;

}

public void setUserName(String userName) {

this.userName = userName;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

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;

}

}