**\*\*\*\*\*\*\*VaishaliSpringComUserWebAppP\*\*\*\*\*\*\*\*\*\*\*\*\*\*JPA\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WebContent/index.jsp\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<h1> This is Spring 4 Home Page</h1>

<a href="myCtrl/hello.obj"> Go To Helllo Welcome Page</a>

<br/>

<a href="loginCtrl/showLoginPage.obj"> Login</a>

<a href="displayAllUserInfo.obj"> Display All User Info</a>

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WEB-INF/jsp/Login.jsp\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

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

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

prefix="form"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<h3>${MsgObj}</h3>

<c:url var="myActionURL" value="validateLoginDetails.obj"/>

<h1>Enter UR Login Details</h1>

<form:form action="${myActionURL}" method="post"

modelAttribute="login">

<table border="1">

<tr>

<td>UserName:</td>

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

<td><form:errors path="userName"/></td>

</tr>

<tr>

<td> Password:</td>

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

<td><form:errors path="password"/></td>

</tr>

<tr>

<td>

<input type="submit"

name=" btnLogin" value="LOGIN" />

</td>

</tr>

</table>

</form:form>

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/WEB-INF/jsp/Register.jsp\*\*\*\*\*\*\***\*\*\*\*\*\*\*\*\*\*\*\*\*

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

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

prefix="c"%>

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

prefix="form"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<h3>${MsgObj}</h3>

<c:url var="regActionURL"

value="insertUserDetails.obj"/>

<h1> Enter UR Registration Details</h1>

<form:form action="${regActionURL}" modelAttribute="user"

method="post">

<table border="1">

<tr>

<td> FirstName:</td>

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

<td></td>

</tr>

<tr>

<td> LastName:</td>

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

<td></td>

</tr>

<tr>

<td> Gender:</td>

<td>

<form:radiobutton path="gender"

value="M" label="Male"/>

<form:radiobutton path="gender"

value="F" label="FeMale"/>

</td>

<td></td>

</tr>

<tr>

<td> Email:</td>

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

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

<td></td>

</tr>

<tr>

<td> SkillSet:</td>

<td>

<form:checkboxes

items="${skillList}" path="skillSet"/>

</td>

<td></td>

</tr>

<tr>

<td> City:</td>

<td>

<form:select path="city">

<form:option value="" label="Select City"/>

<form:options items="${cList}"/>

</form:select>

</td>

<td></td>

</tr>

<tr>

<td> Preffered User name:</td>

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

<td><form:errors path="uname"/></td>

<td></td>

</tr>

<tr>

<td> Password:</td>

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

<td><form:errors path="pwd"/></td>

</tr>

<tr>

<td> Date Of Birth:[dd-mm-yyyy]</td>

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

<td><form:errors path="dobDateText"/></td>

</tr>

<tr>

<input type="submit" name="btnRegister"

value="REGISTER"/>

</tr>

</table>

</form:form>

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WEB-INF/jsp/Failure.jsp\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

Sorry U R Invalid User

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/web-inf/jsp/Hello.jsp\*\*\*\*\*\*\*\***\*\*\*\*\*\*\*\*\*\*\*\*\***\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<h3> Welcome To CapGemin</h3>

<br/>

Today is :${today}

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/web-inf/jsp/ListAlluserDetails.jsp\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

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

prefix="c"%>

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

prefix="form"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

userListObj

<h3> Data is added in the table success\fully</h3>

<hr/>

<table border="1">

<tr>

<th>User Name</th>

<th> First Name</th>

<th> Last Name</th>

<th> City </th>

<th> Email</th>

</tr>

<c:forEach var="userObj" items="${userListObj}">

<tr>

<td>${userObj.uname}</td>

<td>${userObj.firstName}</td>

<td>${userObj.lastName}</td>

<td>${userObj.city}</td>

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

<td><a href="displayEditUser.obj?uid=${userObj.uname}">Edit User</a></td>

<td><a href="">Delete User</a></td>

</tr>

</c:forEach>

</table>

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/web-inf/jsp/Success.jsp\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<h1> Welcome :</h1>

${UserObj}

<h2>U r a Valid User </h2>

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/web-inf/jsp/Updateuser.jsp\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

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

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

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

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

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<table border="1">

<form:form action="updateUser.obj"

modelAttribute="newUserDetailUpdate">

<tr>

<td> UR User Name:</td>

<td>${UserDetailsObject.uname}</td>

<%-- <td><form:errors path="uname"/></td> --%>

<td></td>

</tr>

<tr>

<td> </td>

<td><form:hidden path="uname"

value="${UserDetailsObject.uname}"/></td>

<td></td>

</tr>

<tr>

<td> Existing password:</td>

<td><form:input path="pwd"

value="${UserDetailsObject.pwd}"/></td>

<td><form:errors path="pwd"/></td>

<td></td>

</tr>

<tr>

<td> First Name:</td>

<td><form:input path="firstName"

value="${UserDetailsObject.firstName}" disabled="true"/></td>

<td></td>

</tr>

<tr>

<td> LastName:</td>

<td><form:input path="lastName"

value="${UserDetailsObject.lastName}"/></td>

<td></td>

</tr>

<tr>

<td> City:</td>

<td>

<form:select path="city">

<form:option value="" label="Select City"/>

<form:options items="${cList}"/>

</form:select>

</td>

<td></td>

</tr>

<tr>

<td> Email:</td>

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

value="${UserDetailsObject.email}"/></td>

<%-- <td><form:errors path="email"/></td> --%>

<td></td>

</tr>

<tr>

<td> SkillSet :</td>

<td>

<form:checkboxes path="skillSet" items="${skillList}"/>

</td>

<td></td>

</tr>

<input type="submit" name="btnUpdate"

value="Update User Details"/>

</form:form>

</table>

</body>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8WEB-INF/EMS-Servlet-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:context="http://www.springframework.org/schema/context"

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

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

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

xsi:schemaLocation="http://www.springframework.org/schema/mvc

http://www.springframework.org/schema/mvc/spring-mvc-3.2.xsd

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-3.2.xsd

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

http://www.springframework.org/schema/tx/spring-tx-3.2.xsd"> -->

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

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

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

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

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

http://www.springframework.org/schema/beans/spring-beans-4.0.xsd

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

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

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

http://www.springframework.org/schema/util/spring-util-3.0.xsd

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

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

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

http://www.springframework.org/schema/mvc/spring-mvc-3.2.xsd

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

http://www.springframework.org/schema/tx/spring-tx-3.2.xsd">

<context:annotation-config />

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

<mvc:annotation-driven validator="myBeansValidator" />

<bean id="myBeansValidator"

class="org.springframework.validation.beanvalidation.LocalValidatorFactoryBean" />

<bean

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

p:prefix="/WEB-INF/jsp/" p:suffix=".jsp" />

<!-- This bean is required to load ORM persistence and related annotations -->

<bean

class="org.springframework.orm.jpa.support.PersistenceAnnotationBeanPostProcessor" />

<!-- This bean will hold references to our database -->

<bean id="dataSource"

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

<property name="driverClassName" value="oracle.jdbc.driver.OracleDriver" />

<property name="url" value="jdbc:oracle:thin:@localhost:1521:XE" />

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

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

</bean>

<!-- Spring way of entity manager factory configuration -->

<bean id="entityManagerFactory"

class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">

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

<property name="packagesToScan" value="com.cg.bean" />

<property name="persistenceProviderClass"

value="org.hibernate.jpa.HibernatePersistenceProvider" />

<property name="jpaPropertyMap">

<map>

<entry key="hibernate.dialect" value="org.hibernate.dialect.Oracle10gDialect" />

<!-- <entry key="hibernate.hbm2ddl.auto" value="update" /> -->

</map>

</property>

</bean>

<!-- This bean will handle all transaction within an application -->

<bean id="transactionManager" class="org.springframework.orm.jpa.JpaTransactionManager">

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

</bean>

<!-- This is required to handle transactions via Annotations like @Transactional -->

<tx:annotation-driven transaction-manager="transactionManager" />

<!-- <jpa:repositories base-package="com.cg.dao" entity-manager-factory-ref="entityManagerFactory"

transaction-manager-ref="transactionManager"/> -->

</beans>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*web-inf/web.xml\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

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

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"

id="WebApp\_ID" version="3.1">

<display-name>VaishaliSpringComUserWebAppP</display-name>

<welcome-file-list>

<welcome-file>index.jsp</welcome-file>

</welcome-file-list>

<servlet>

<servlet-name>EMSServlet</servlet-name>

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

</servlet>

<servlet-mapping>

<servlet-name>EMSServlet</servlet-name>

<url-pattern>\*.obj</url-pattern>

</servlet-mapping>

</web-app>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*tableScript\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

drop table cg\_userdetails cascade constraints;

drop table cg\_users cascade constraints;

CREATE TABLE cg\_userdetails

(

user\_name varchar2(15) primary key,first\_name VARCHAR2(10),

last\_name VARCHAR2(10),user\_city varchar2(10),

user\_skill varchar2(20),user\_email varchar2 (20),user\_gender varchar2(1),

dob Date);

CREATE table cg\_users(user\_name varchar2(15) primary key ,

password varchar2(20));

select \* from cg\_users;

select \* from cg\_userdetails;

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Login.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.bean;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

import javax.validation.constraints.Pattern;

import javax.validation.constraints.Size;

import org.hibernate.validator.constraints.NotEmpty;

@Entity

@Table(name="cg\_users")

public class Login

{

@Id

@Column(name="user\_name") private String userName;

@Column(name="password") private String password;

@NotEmpty(message="User name Mandatory")

@Size(min=5,message="UserName Should Have Min 5 Character")

public String getUserName() { return userName; }

public void setUserName(String userName) { this.userName = userName; }

@NotEmpty(message="Password Is Mandatory")

public String getPassword() { return password; }

public void setPassword(String password) { this.password = password; }

@Override

public String toString() {

return "Login [userName=" + userName + ", password=" + password + "]";

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RegisterDto.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.bean;

import java.util.Arrays;

import java.util.Date;

import javax.persistence.\*;

import javax.validation.constraints.Pattern;

import org.hibernate.validator.constraints.Email;

import org.hibernate.validator.constraints.NotEmpty;

@Entity

@Table(name="cg\_userDetails")

public class RegisterDto

{

@Id

@Column(name="user\_name")

@NotEmpty(message="Mandatory User name")

@Pattern(regexp="[A-Za-z]{1,15}",

message="Only Charactes allowed and Start with Caplital")

private String uname;

@Override

public String toString() {

return "RegisterDto [uname=" + uname + ", pwd=" + pwd

+ ", confirmPassword=" + confirmPassword + ", firstName="

+ firstName + ", lastName=" + lastName + ", email=" + email

+ ", skillSet=" + Arrays.toString(skillSet) + ", gender="

+ gender + ", city=" + city + ", skillSetStr=" + skillSetStr

+ ", dobUtilDate=" + dobUtilDate + ", dobDateText="

+ dobDateText + "]";

}

@Transient

private String pwd;

@Transient

private String confirmPassword;

@Column(name="first\_name")

private String firstName;

@Column(name="last\_name")

private String lastName;

@Column(name="user\_email")

private String email;

@Transient

private String skillSet[];

@Column(name="user\_gender")

private char gender;

@Column(name="user\_city")

private String city;

@Column(name="user\_skill")

private String skillSetStr;

@Column(name="dob")

@Temporal(TemporalType.DATE)

private Date dobUtilDate;

public Date getDobUtilDate() { return dobUtilDate; }

public void setDobUtilDate(Date dobUtilDate) { this.dobUtilDate = dobUtilDate; }

public String getDobDateText() { return dobDateText; }

public void setDobDateText(String dobDateText) { this.dobDateText = dobDateText; }

@Transient

@Pattern(regexp="\\d{2}-\\d{2}-\\d{4}",

message="Enter Date in dd-mm-yyyy format ")

String dobDateText;

public String getUname() { return uname; }

public RegisterDto(String uname) {

super();

this.uname = uname;

}

public RegisterDto() { super(); }

public void setUname(String uname) { this.uname = uname; }

@NotEmpty(message="Mandatory Passwod")

@Pattern(regexp="[a-zA-Z0-9]{5,}",

message="Minimum 5 char and only small cplital letteres and digts are allowed.")

public String getPwd() { return pwd; }

public void setPwd(String pwd) { this.pwd = pwd; }

public String getConfirmPassword() { return confirmPassword; }

public void setConfirmPassword(String confirmPassword) {

this.confirmPassword = confirmPassword;

}

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; }

@NotEmpty(message="Manadatory Field")

@Email(message=" Plaese enter valid email address")

public String getEmail() { return email; }

public void setEmail(String email) { this.email = email; }

public String[] getSkillSet() { return skillSet; }

public void setSkillSet(String[] skillSet) { this.skillSet = skillSet; }

public char getGender() { return gender; }

public void setGender(char gender) { this.gender = gender; }

public String getCity() { return city; }

public void setCity(String city) { this.city = city; }

public String getSkillSetStr() { return skillSetStr; }

public void setSkillSetStr(String skillSetStr) { this.skillSetStr = skillSetStr; }

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*HelloController.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.ctrl;

import java.time.LocalDate;

import org.springframework.stereotype.Controller;

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

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

import org.springframework.web.servlet.ModelAndView;

@Controller

@RequestMapping("myCtrl")

public class HelloController

{

@RequestMapping(value="/hello",

method=RequestMethod.GET)

public ModelAndView sayMsg()

{

String td=LocalDate.now().toString();

return new ModelAndView("Hello","today",td);

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LoginController.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.ctrl;

import javax.validation.Valid;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.validation.BindingResult;

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

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

import com.cg.bean.Login;

import com.cg.service.ILoginService;

@Controller

@RequestMapping("loginCtrl")

public class LoginController

{

@Autowired

ILoginService loginService;

@RequestMapping(value="/showLoginPage")

public String displyLoginPage(Model model)

{

Login lg=new Login();

model.addAttribute("login",lg);

return "Login";

}

@RequestMapping(value="/validateLoginDetails")

public String validateUserDetails(

@ModelAttribute(value="login")

@Valid Login lgg,

BindingResult result,

Model model)

{

if(result.hasErrors())

{

return "Login";

}

String un=lgg.getUserName();

String pw=lgg.getPassword();

Login existingUserObj=loginService.getUserByUserName(un);

System.out.println(" in login ctrl user exist : "+ existingUserObj);

if(existingUserObj==null)

{

//model.addAttribute("MsgObj","User Does not Exist");

//return "Register";

return "redirect:/displayRegisterPage.obj";

}

else

{

if(un.equalsIgnoreCase(existingUserObj.getUserName())

&&

(pw.equalsIgnoreCase(existingUserObj.getPassword())))

{

model.addAttribute("UserObj", un);

return "Success";

}

else

{

model.addAttribute("MsgObj","Enter Correct Password");

return "Login";

} } }}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RegisterController.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.ctrl;

import java.util.ArrayList;

import java.util.List;

import javax.validation.Valid;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

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 org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.servlet.ModelAndView;

import com.cg.bean.Login;

import com.cg.bean.RegisterDto;

import com.cg.service.ILoginService;

import sun.net.RegisteredDomain;

@Controller

public class RegisterController

{

@Autowired

ILoginService loginService;

ArrayList<String> cityList=null;

ArrayList<String> skillSetList=null;

@RequestMapping(value="displayRegisterPage")

public String showRegistrationPage(Model model)

{

RegisterDto userObj=new RegisterDto();

model.addAttribute("user",userObj);

cityList=new ArrayList<String>();

//Add City In the Model

cityList.add("Gaziabad");

cityList.add("Pune");

cityList.add("Kolkatta");

cityList.add("Chennai");

model.addAttribute("cList",cityList);

skillSetList=new ArrayList<String>();

skillSetList.add("Java");

skillSetList.add("C++");

skillSetList.add("SAP");

skillSetList.add("DotNet");

model.addAttribute("skillList",skillSetList);

model.addAttribute("MsgObj","User Does not Exist");

return "Register";

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@RequestMapping(value="/insertUserDetails")

public String insertUserDetails(

@ModelAttribute(value="user")@Valid RegisterDto rd,

BindingResult result,Model model)

{

if(result.hasErrors())

{

model.addAttribute("skillList",skillSetList);

model.addAttribute("cList",cityList);

return "Register";

}

System.out.println(" In reg ctrl rd is : "+rd);

Login lg=new Login();

lg.setUserName(rd.getUname());

lg.setPassword(rd.getPwd());

RegisterDto tempRd=loginService.addAllUserDeatils(rd);

System.out.println("user details addedd\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Login logDto=loginService.addUser(lg);

System.out.println("user addedd\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println(" rd and log data is Added...."+

tempRd);

return "redirect:/displayAllUserInfo.obj";

}

/\*...................displayEditUser..action..................\*/

@RequestMapping(value="displayEditUser",

method=RequestMethod.GET)

public String displayUpdateUserPage(

@RequestParam("uid") String un,Model model)

{

skillSetList=new ArrayList<String>();

skillSetList.add("C");

skillSetList.add("C++");

skillSetList.add("Java");

skillSetList.add("Unix");

Login userInfo=loginService.getUserByUserName(un);

RegisterDto userDetailInfo=loginService.getRegisterUserDetails(un);

userDetailInfo.setPwd(userInfo.getPassword());

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*"+userInfo);

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*"+userDetailInfo);

model.addAttribute("newUserDetailUpdate",new RegisterDto());

model.addAttribute("UserDetailsObject",userDetailInfo);

model.addAttribute("skillList", skillSetList);

cityList=new ArrayList<String>();

//Add City In the Model

cityList.add("Gaziabad");

cityList.add("Pune");

cityList.add("Kolkatta");

cityList.add("Chennai");

model.addAttribute("cList",cityList);

return "UpdateUser";

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Update Users\*\*\*\*\*\*\*\*\*\*\*\*\*/

@RequestMapping("updateUser")

public String updateUserDetails(

@ModelAttribute("newUserDetailUpdate") RegisterDto u,

BindingResult result,

Model model)

{

System.out.println("In controler update \*\*\*\*\*\*\*\*"+u);

loginService.updateUserDetailsInfo(u);

System.out.println("after update \*\*\*\*\*\*\*\*"+u);

Login lg=new Login();

lg.setUserName(u.getUname());

lg.setPassword(u.getPwd());

System.out.println("In controler update \*\*\*\*\*\*\*\*"+lg);

loginService.updateUserInfo(lg);

//if((rowUserUpdated==1)&&(rowLoginUpdated==1))

//{

System.out.println("both row updated update \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

return "redirect:/displayAllUserInfo.obj";

//}

//return "index";

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*List all users\*\*\*\*\*\*\*\*\*\*/

@RequestMapping("displayAllUserInfo")

public String displayAllUserInfo(Model model)

{

ArrayList<RegisterDto> uL=loginService.

getAllUserDetails();

model.addAttribute("userListObj", uL);

return "ListAllUserDetails";

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*ILoginDao.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8**

package com.cg.dao;

import java.util.ArrayList;

import com.cg.bean.Login;

import com.cg.bean.RegisterDto;

public interface ILoginDao

{

public Login getUserByUserName(String un);

public boolean isValid(String un);

public boolean isUserExist(String un);

public RegisterDto addAllUserDeatils(RegisterDto regDto);

public Login addUser(Login log);

public ArrayList<RegisterDto> getAllUserDetails();

public RegisterDto getRegisterUserDetails(String un);

public void updateUserDetailsInfo(RegisterDto u);

public void updateUserInfo(Login lg);

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LoginDaoImpl.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.dao;

import java.util.ArrayList;

import javax.persistence.EntityManager;

import javax.persistence.PersistenceContext;

import javax.persistence.Query;

import javax.persistence.TypedQuery;

import javax.transaction.Transactional;

import org.springframework.stereotype.Component;

import org.springframework.stereotype.Repository;

import com.cg.bean.Login;

import com.cg.bean.RegisterDto;

import com.cg.util.MyStringDateUtil;

@Repository("loginDao")

@Transactional

public class LoginDaoImpl implements ILoginDao

{

@PersistenceContext

private EntityManager entityManager;

public EntityManager getEntityManager() {

return entityManager;

}

public void setEntityManager(EntityManager entityManager) {

this.entityManager = entityManager;

}

@Override

public Login getUserByUserName(String un)

{

Login log=entityManager.find(Login.class, un);

System.out.println(" In Login Dao Rec retrieve is "+ log);

return log;

}

@Override

public boolean isValid(String un)

{

return false;

}

@Override

public boolean isUserExist(String un)

{

return false;

}

@Override

public RegisterDto addAllUserDeatils(RegisterDto regDto) {

System.out.println(" in dao regi ster obj is "+ regDto);

//Convert skillset array to string using , separator

String skillSetString=MyStringDateUtil.fromArrayToCommaSeparatedString(regDto.getSkillSet());

regDto.setSkillSetStr(skillSetString);

String gender=new Character(regDto.getGender()).toString();

System.out.println("In dao skill is ..............................."+skillSetString+ "before persist : "+regDto);

entityManager.persist(regDto);

entityManager.flush();

System.out.println(" In Dao class after persist................"+regDto);

return regDto;

}

@Override

public Login addUser(Login log) {

System.out.println(" in dao log obj is "+ log);

entityManager.persist(log);

entityManager.flush();

System.out.println(" In Dao class after log ersist................"+log);

return log;

}

@Override

public ArrayList<RegisterDto> getAllUserDetails() {

String fetchUserQry="SELECT u FROM RegisterDto u ";

TypedQuery<RegisterDto> tq=entityManager.createQuery(fetchUserQry, RegisterDto.class);

ArrayList <RegisterDto> uList=(ArrayList)tq. getResultList();

return uList;

}

@Override

public RegisterDto getRegisterUserDetails(String un) {

RegisterDto regDto=entityManager.find(RegisterDto.class, un);

System.out.println(" In getRegisterUserDetails Dao Rec retrieve is "+ regDto);

return regDto;

}

@Override

public void updateUserDetailsInfo(RegisterDto u) {

String skillSetString=MyStringDateUtil.fromArrayToCommaSeparatedString

(u.getSkillSet());

u.setSkillSetStr(skillSetString);

Query query1 = entityManager.createQuery

("UPDATE RegisterDto rd SET rd.lastName = :ln,"

+ " rd.email =:eml ,rd.city = :ct, "

+ " rd.skillSetStr=:ss WHERE rd.uname= :un");

Query query2 = entityManager.createQuery

("UPDATE Login log SET log.password = :pw WHERE log.userName= :un");

query1.setParameter("ln", u.getLastName());

query1.setParameter("eml", u.getEmail());

query1.setParameter("ct", u.getCity());

query1.setParameter("ss", u.getSkillSetStr());

query1.setParameter("un", u.getUname());

query2.setParameter("pw", u.getPwd());

query2.setParameter("un", u.getUname());

int dataUp1=query1.executeUpdate();

int dataUp2=query2.executeUpdate();

}

@Override

public void updateUserInfo(Login lg) {

Login lgg=entityManager.find(Login.class, lg.getUserName());

entityManager.merge(lgg);

entityManager.flush();

Login lggg=entityManager.find(Login.class, lg.getUserName());

System.out.println("in updateUserInfo updated..........."+lggg);System.out.println("In dao after merge and flush..........."+ lggg); }

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*ILoginService.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.service;

import java.util.ArrayList;

import com.cg.bean.Login;

import com.cg.bean.RegisterDto;

public interface ILoginService {

public Login getUserByUserName(String un);

public boolean isValid(String un);

public boolean isUserExist(String un);

public RegisterDto addAllUserDeatils(RegisterDto regDto);

public Login addUser(Login log);

public ArrayList<RegisterDto>

getAllUserDetails();

public RegisterDto getRegisterUserDetails(String un);

public void updateUserDetailsInfo(RegisterDto u);

public void updateUserInfo(Login lg);}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LoginServiceImpl.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package com.cg.service;

import java.util.ArrayList;

import javax.transaction.Transactional;

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

import org.springframework.stereotype.Service;

import com.cg.bean.Login;

import com.cg.bean.RegisterDto;

import com.cg.dao.ILoginDao;

import com.cg.util.MyStringDateUtil;

@Service("loginService")

@Transactional

public class LoginServiceImpl implements

ILoginService

{

@Autowired

ILoginDao loginDao;

public ILoginDao getLoginDao() {

return loginDao;

}

public void setLoginDao(ILoginDao loginDao) { this.loginDao = loginDao; }

@Override

public Login getUserByUserName(String un) {return loginDao.getUserByUserName(un); }

@Override

public boolean isValid(String un) { return false; }

@Override

public boolean isUserExist(String un) { return false; }

@Override

public RegisterDto addAllUserDeatils(RegisterDto regDto) {

regDto.setDobUtilDate(MyStringDateUtil.

fromStringToSqlDate( regDto.getDobDateText()));

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*in service \*\*\*\*\*\*\*"+

regDto.getDobUtilDate());

return loginDao.addAllUserDeatils(regDto);

}

@Override

public Login addUser(Login log) {

return loginDao.addUser(log);

}

@Override

public ArrayList<RegisterDto> getAllUserDetails() {

// TODO Auto-generated method stub

return loginDao.getAllUserDetails();

}

@Override

public RegisterDto getRegisterUserDetails(String un) {

return loginDao.getRegisterUserDetails(un);

}

@Override

public void updateUserDetailsInfo(RegisterDto u) {

// TODO Auto-generated method stub

loginDao.updateUserDetailsInfo(u);

}

@Override

public void updateUserInfo(Login lg) {// TODO Auto-generated method stub

loginDao.updateUserInfo(lg);

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*MyStringDateUtil.java\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8**

package com.cg.util;

import java.sql.Date;

import java.text.DateFormat;

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

import java.util.ArrayList;

import java.util.Iterator;

public class MyStringDateUtil

{

public static LocalDate fromSqlToLocalDate(Date sqlDate)

{

LocalDate dobLocalDate=sqlDate.toLocalDate();

return dobLocalDate;

}

public static ArrayList<String> fromStringToArrayList(String str)

{

ArrayList<String> stringList=new ArrayList();

for(String tempStr:str.split(","))

{

stringList.add(tempStr);

}

return stringList;

}

public static String[] fromArrayListToArray(ArrayList<String> strList)

{

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*From Array List To String Array \*\*\*\*");

String stringArray[]=strList.toArray (new String[strList.size()]);

for(int i=0;i<stringArray.length;i++)

{

System.out.println(stringArray[i]);

}

return stringArray;

}

public static String fromArrayToCommaSeparatedString(String strArray[])

{

System.out.println("\*\*\*\*\*\*\*\*\*\*From Array String\*\*\*\*\*\*\*\*\*\*");

StringBuffer str=new StringBuffer();

for (String tempStr:strArray )

{

str.append(tempStr+",");

//str+=tempStr+",";

}

return str.toString();

}

public static Date fromLocalToSqlDate(LocalDate localDate)

{

System.out.println(" from local to sql.....................");

java.sql.Date sqlDate=java.sql.Date.valueOf(localDate);

return sqlDate;

}

public static LocalDate fromStringToLocalDate(String dobText)

{

LocalDate localDob=LocalDate.

parse(dobText, DateTimeFormatter.ofPattern("dd-MM-yyyy"));

return localDob;

}

public static Date fromStringToSqlDate(String dobText)

{

LocalDate localDob=LocalDate. parse(dobText, DateTimeFormatter.ofPattern("dd-MM-yyyy"));

java.sql.Date sqlDate=java.sql.Date.valueOf(localDob);

return sqlDate;

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Thank You\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***