Interface;

Methods to declare;

**package** studentpack;

**import** java.util.List;

**public** **interface** StudentDAO {

**public** **void** insertStudent(Student s);

**public** List<Student> showallStudents();

}

Student(POJO) class;

package studentpack;

import javax.validation.constraints.Min;

import org.hibernate.validator.constraints.NotEmpty;

public class Student {

@NotEmpty(message="please enter your name")

private String name;

@Min(value=10, message="enter valid id")

private int id;

@NotEmpty(message="please enter your fathername")

private String fathername;

@NotEmpty(message="address cannot be empty")

private String address;

@Min(value=10, message="enter valid phonenumber")

private long mobilenumber;

public Student() {

}

public Student(String name, int id, String fathername, String address, long mobilenumber) {

super();

this.name = name;

this.id = id;

this.fathername = fathername;

this.address = address;

this.mobilenumber = mobilenumber;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFathername() {

return fathername;

}

public void setFathername(String fathername) {

this.fathername = fathername;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public long getMobilenumber() {

return mobilenumber;

}

public void setMobilenumber(long mobilenumber) {

this.mobilenumber = mobilenumber;

}

@Override

public String toString() {

return "Student [name=" + name + ", id=" + id + ", fathername=" + fathername + ", address=" + address

+ ", mobilenumber=" + mobilenumber + "]";

}

}

To implement method in interface;

package studentpack;

import java.util.List;

import javax.sql.DataSource;

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

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.stereotype.Repository;

import productpack.Product;

import productpack.ProductMapper;

@Repository

public class StudentImpl implements StudentDAO {

DataSource datasource;

JdbcTemplate jdbctemplate;

@Autowired

public void setDatasource(DataSource datasource)

{

this.datasource = datasource;

this.jdbctemplate= new JdbcTemplate(datasource);

}

@Override

public void insertStudent(Student s) {

String sql="insert into student(name,id,fathername,address,mobilenumber) values (?,?,?,?,?)";

jdbctemplate.update(sql,s.getName(),s.getId(),s.getFathername(),s.getAddress(),s.getMobilenumber());

System.out.println("one record inserted");

}

@Override

public List<Student> showallStudents() {

String sql="select name,id,fathername,address,mobilenumber from student";

List<Student> studentlist=jdbctemplate.query(sql, new StudentMapper());

return studentlist;

}

}

Student mapper class;

package studentpack;

import java.sql.ResultSet;

import java.sql.SQLException;

import org.springframework.jdbc.core.RowMapper;

public class StudentMapper implements RowMapper<Student> {

@Override

public Student mapRow(ResultSet rs, int rownum) throws SQLException {

Student s= new Student();

s.setName(rs.getString(1));

s.setId(rs.getInt(2));

s.setFathername(rs.getString(3));

s.setAddress(rs.getString(4));

s.setMobilenumber(rs.getLong(5));

return s;

}

}

Controller class;

package studentpack;

import java.util.List;

import java.util.Map;

import javax.validation.Valid;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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;

@Controller

@RequestMapping(value="/page")

public class StudentController {

@Autowired

StudentDAO studentDAO;

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

public String insertRecord(Map<String,Object>Model)

{

Student s=new Student();

Model.put("data", s);

return "detail";

}

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

public String savedata(@Valid @ModelAttribute ("data") Student s,BindingResult result,Map<String,Object>Model)

{

if(result.hasErrors())

{

return "detail";

}

studentDAO.insertStudent(s);

return "pages";

}

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

public String viewlist( ModelMap model)

{

List<Student> s=studentDAO.showallStudents();

model.addAttribute("s", s);

return "allstudent";

}

}

JSP’s for frontend;

For entering students details;

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

pageEncoding=*"ISO-8859-1"*%>

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

</head>

<body style="background-color:*pink*;">

<div align=*"center"* >

<form:form action=*"add"* method=*"post"* commandName=*"data"*>

<h2>Registration Page</h2>

<p>Student Name</p>

<form:input path=*"name"*/><form:errors path=*"name"* cssStyle=*"color:red"*/>

<p>Student Id</p>

<form:input path=*"id"*/><form:errors path=*"id"* cssStyle=*"color:red"*/>

<p>Father Name</p>

<form:input path=*"fathername"*/> <form:errors path=*"fathername"* cssStyle=*"color:red"*/>

<p>Address</p>

<form:input path=*"address"*/> <form:errors path=*"address"* cssStyle=*"color:red"*/>

<p>Mobile Number</p>

<form:input path=*"mobilenumber"*/> <form:errors path=*"mobilenumber"* cssStyle=*"color:red"*/>

<br>

<input type=*"submit"* value=*"register"*/>

</form:form>

</div>

</body>

</html>

To display whether it is successfully inserted or not;

<%@ 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>Success</title>

</head>

<body style="background-color:*powderblue*;">

<h1>Successfully Inserted</h1>

Student Name:${s.name}

<br>

Student Id:${s.id}

<br>

Father Name:${s.fathername}

<br>

Address:${s.address}

<br>

Mobile Number:${s.mobilenumber}

<br>

<a href=*"view"*>View Full List</a>

</body>

</html>

To display all student details;

<%@ 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"*%>

<!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 style="background-color:*pink*;">

<h1 align=*"center"*>All Details Available</h1>

<div align=*"center"*>

<table border=*"0"*>

<tr>

<td colspan =*"1"* align=*"center"*></td>

</tr>

<tr>

<th>Student Name</th> <th>Student Id</th> <th>Father Name</th> <th>Address</th> <th>Mobile Number</th>

</tr>

<c:forEach items = *"*${s}*"* var=*"b"*>

<tr>

<td>${b.getName()}</td><td>${b.getId()}</td><td>${b.getFathername()}</td><td>${b.getAddress()}</td><td>${b.getMobilenumber()}</td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>

Dispatcher servlet for connection;

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

<beans xmlns=*"http://www.springframework.org/schema/beans"*

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

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

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

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

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

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

<context:annotation-config/>

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

<mvc:annotation-driven />

<bean

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

<property name=*"prefix"*>

<value>/WEB-INF/view/</value>

</property>

<property name=*"suffix"*>

<value>.jsp</value>

</property>

</bean>

<!-- <bean

class="org.springframework.web.servlet.handler.SimpleMappingExceptionResolver">

<property name="exceptionMappings">

<props>

<prop key="java.lang.ArithmeticException">matherror</prop>

<prop key="java.lang.ArrayIndexOutOfBouundsException">matherror</prop>

</props>

</property>

<property name="defaultErrorView" value="GeneralError"/>

</bean> -->

<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=*"system"*/>

</bean>

<!-- <bean id="myInterceptor" class="book.MyInterceptor">

</bean>

<bean class="org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter" />

<bean class="org.springframework.web.servlet.mvc.annotation.DefaultAnnotationHandlerMapping">

<property name="interceptors">

<list>

<ref bean="myInterceptor"/>

</list>

</property>

</bean>

<mvc:interceptors>

<mvc:interceptor>

<mvc:mapping path="/book/\*"/>

<ref bean="myInterceptor" />

</mvc:interceptor>

</mvc:interceptors>-->

</beans>

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>SpringWeb</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>