# 实验四 STRUTS

实验内容

学习 struts2 应用的基本开发

实验目的

了解 Struts 框架的机制

握 Struts 应用的运行环境配置

掌握 Action、拦截器的编写与配置

掌握 Struts 类型转换器、输入校验框架的使用及配置

了解 ONGL、值栈，

掌握 Struts 常用标签

环境要求

服务器：Tomcat6.0 或更高

集成开发环境：MyEclipse2014 或更高

实验指导

**1. Struts 简单使用**

（1）建立一个 web 工程名为：strutsDemo，功能模拟用户登录。在工程名上点击右键，选择弹出菜 单“MyEclipse”->“Project Facets [Capabilites]”-> “Install Apache Struts(2.x) Facet”添加对 Struts 的支持 （如图 1）。在弹出的对话框中（图 2）选择合适的“Target Runtime”（Web 服务器），点击“next”打开 Struts 的控制配置（图 3），选择：URL pattern 为：/\*

然后点击“Finish”按钮，完成 struts2 支持的添加。

（2）编写视图文件 login.jsp——登录页面，表单代码如下：

窗体顶端

用户名：  
口令：  

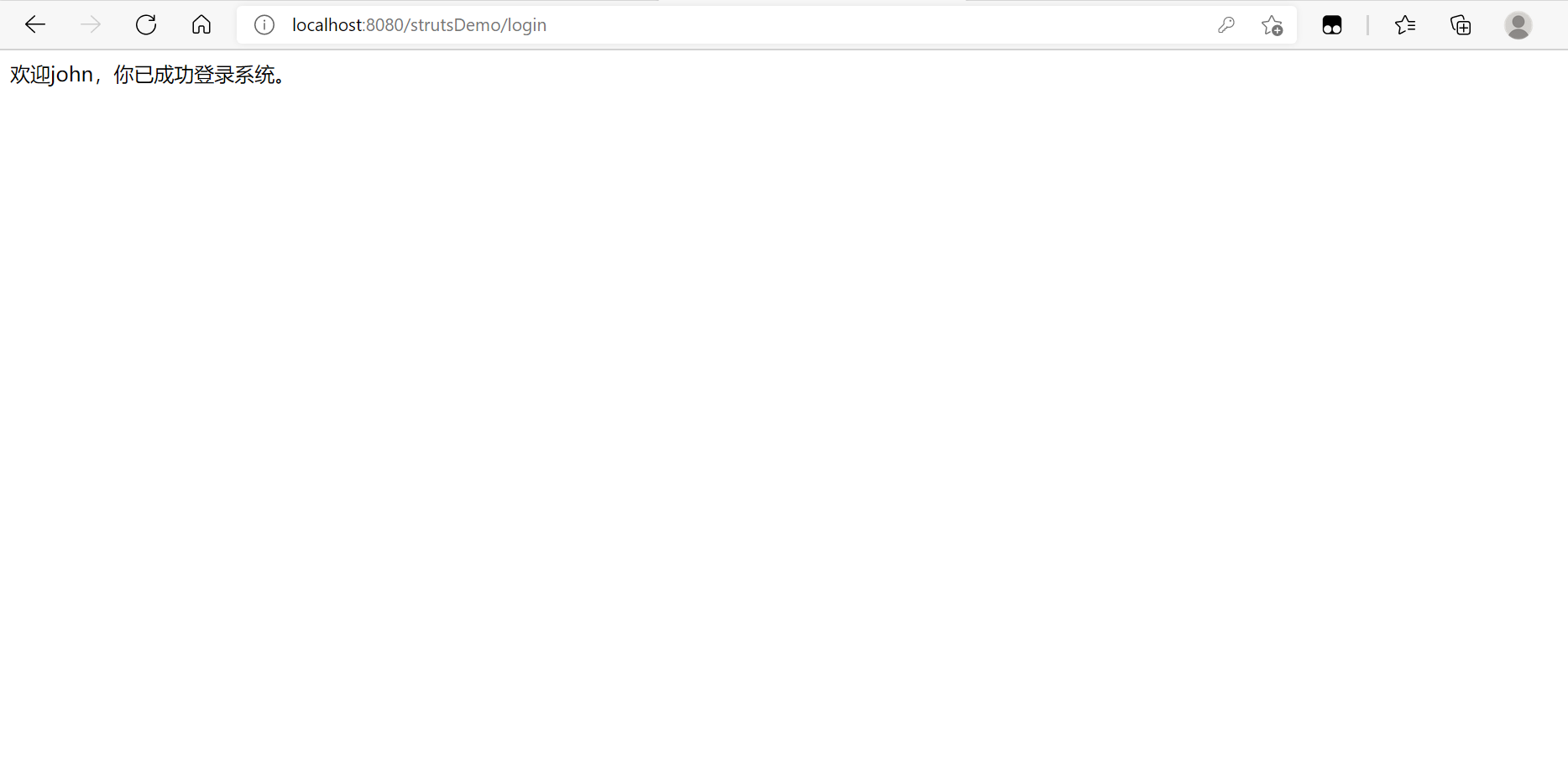

窗体底端

success.jsp——登录成功页面，正文区内容：

欢迎${uname}，你已成功登录系统。

error.jsp——登录失败页面： 用户名或密码有误，请重新登录





（3）编写 Action 创建一个普通 Java 类——LoginAction.java，所在包：actions，内容如下：

package actions;

import com.opensymphony.xwork2.Action;

public class LoginAction { private String uname;

//对应用登录表单的 uname 文本域

private String upasswd;//对应用登录表单的

upasswd 密码域

/\*\*

\* Struts 控制器默认调用 Action 的方法

\* @return 字符串，根据此值在配置文件中定义相应的视图

\*/

public String execute(){

if("john".equals(uname)&&"123456".equals(upasswd))

{ return Action.SUCCESS; //返回 Action 预定义的常量，值为 “success” }

else{ return "fail";//返回自定义值，表示登录失败 } }

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

public String getUname() { return uname; }

public void setUpasswd(String upasswd) { this.upasswd = upasswd; } public String getUpasswd() { return upasswd; } }

**（4）编辑 struts.xml 配置 Action，在直接添加如下配置：**

/success.jsp /error.jsp

（5）部署后访问，查看结果。

**2. 基于注解的 Action**

将上例的 LoginAction 类修改，在类声明前添加如下内容对该类进行注解。

@Namespace("/")

@ParentPackage("struts-default") @Results({ @Result(location="/success.jsp"), @Result(name="fail",location="/error.jsp") })

删除 struts.xml 中对 loginAction 的配置，然后重新部署并访问。

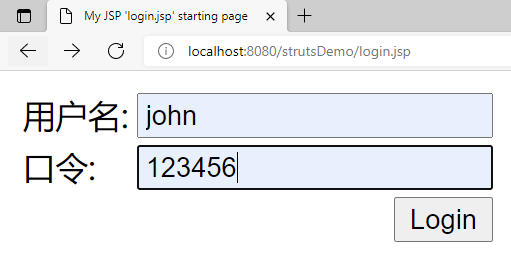
注：注意访问此种 Action 的名称。

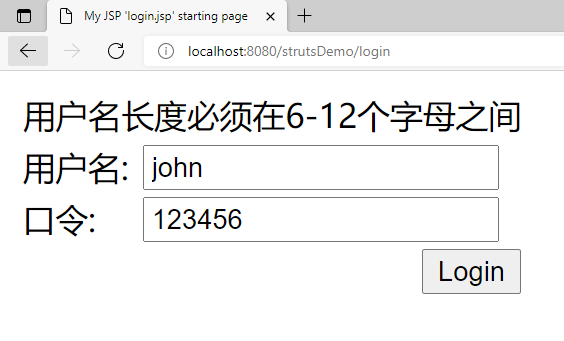
**3. 带有验证的 Action**

对前面的 LoginAction 进行改造，要求用户输入的用户名的长度必须在 6-12 个字符 之间，否则返回登录页面并显示错误信息：“用户名长度必须在 6-12 个字母之间”。方法： 将 LoginAction 的实现继承自 ActionSupport 类，并重写 validate 方法。

在 login 页面，用户名中输入少于 6 个或多于 12 个字母，点击“登录”按钮后的结果 如图：图 3

运行结果：





**4. 基于模型驱动的 Action**

使用模型驱动的 Action 来收集学生信息。

JavaBean：

public class Student{

private String id;

private String name;

private Integer age;

private String className //省略 getter/setter 方法;

}

**Action：**

public class ModelAction implements ModelDriven{

private Student student;

private String passwd; //除驱动 Bean 之外的其他属性

public String execute(){

return Action.SUCCESS;

}

@Override public Student getModel()

{

student = new Student();

return student;

}

public String getPasswd()

{ return passwd; }

public void setPasswd(String passwd)

{ this.passwd = passwd; }

}

**JSP 页面：**

数据提交（输入）页面——input.jsp：

窗体顶端

窗体底端

数据显示页面——info.jsp： 学号：${id}

姓名：${uname}

年龄：${age}

班级：${className}

初始口令：${passwd}

自行配置 Action 进行测试

**5. 访问普通 Action 的复合属性（bean/map/数组/list）**

实现 4 中的功能：

JavaBean：

public class Student{

private String id;

private String name;

private Integer age;

private String className

private List hobs; //省略 getter/setter 方法;

}

**Action:**

Public class stuAction{

private Student stu;

private String passwd;

public String execute(){

return Action.SUCCESS; }

public Student getStu()

{ return stu; }

public void setStu(Student stu)

{ this.stu = stu; }

public String getPasswd() {

return passwd;

}

public void setPasswd(String passwd)

{ this.passwd = passwd; } }

**JSP 页面：**

数据提交（输入）页面——input.jsp：

窗体顶端

学号：  
姓名：  
年龄：  
班级：  
爱好： 阅读  
足球  
旅游  
游泳  
初始口令：

窗体底端

数据显示页面——info.jsp： 学号：${stu.id}  
姓名：${stu.uname}  
年龄：${stu.age}  
班级：${stu.className}  
爱好：${stu.hobs}  
初始口令：${passwd} 自行配置 Action 进行测试

**6. 拦截器的使用**

（1）练习书中 token 拦截器，实现防止重复提交的功能。

（2）编写自定义拦截器，对上面的登录进行拦截，要求输入的用户名必须以 stu 开始， 否则返回登录页面，并显示相应信息。创建 NameInvoke(包名：invoke)，代码如下：

public class NameInvoke extends AbstractInterceptor

{

@Override

public String intercept(ActionInvocation arg0) throws Exception {Object action = arg0.getAction();//得到被拦截的 Action

if(action instanceof LoginAction)

{

LoginAction login = (LoginAction)action;

String uname = login.getUname();

if(uname != null && uname.startsWith("stu"))

{ return arg0.invoke(); }

else{

login.addFieldError("uname", "用户名必须以 stu 开始");

return Action.INPUT; } }

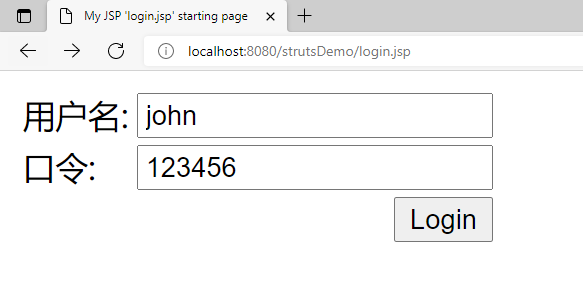
return null;

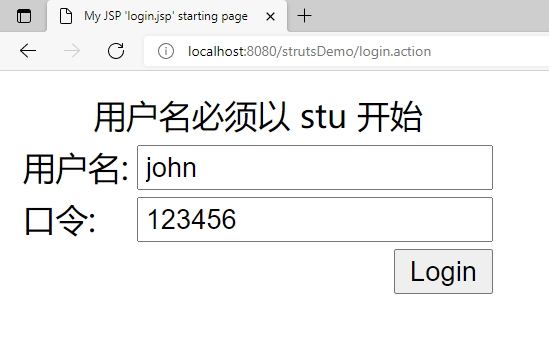
}

}

修改 Struts.xml 的配置为：

/success.jsp /login.jsp





**7. Struts 标签的使用**

1. 练习书中 P136-138 学生成绩录入与显示。

**核心代码：**

**Score.java**

**package** beans;

**public** **class** Score {

**private** **int** javaScore;

**private** **int** j2eeScore;

**private** **int** ccScore;

**private** **double** aveScore;

**public** **int** getCcScore() {

**return** ccScore;

}

**public** **void** setCcScore(**int** ccScore) {

**this**.ccScore = ccScore;

}

**public** **int** getJavaScore() {

**return** javaScore;

}

**public** **void** setJavaScore(**int** javaScore) {

**this**.javaScore = javaScore;

}

**public** **int** getJ2eeScore() {

**return** j2eeScore;

}

**public** **void** setJ2eeScore(**int** j2eeScore) {

**this**.j2eeScore = j2eeScore;

}

**public** **double** getAveScore() {

**return** aveScore;

}

**public** **void** setAveScore(**double** aveScore) {

**this**.aveScore = aveScore;

}

@Override

**public** String toString() {

**return** "Score [aveScore=" + aveScore + ", cScore=" + ccScore

+ ", j2eeScore=" + j2eeScore + ", javaScore=" + javaScore + "]";

}

}

**Student.java**

**package** beans;

**public** **class** Student {

**private** String name;

**private** **long** number;

**private** Score score;

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** **long** getNumber() {

**return** number;

}

**public** **void** setNumber(**long** number) {

**this**.number = number;

}

**public** Score getScore() {

**return** score;

}

**public** **void** setScore(Score score) {

**this**.score = score;

}

@Override

**public** String toString() {

**return** "Student [name=" + name + ", number=" + number + ", score="

+ score + "]";

}

}

**ScoreAction.java**

**package actions;**

**import java.math.BigDecimal;**

**import java.util.List;**

**import beans.Score;**

**import beans.Student;**

**import com.opensymphony.xwork2.ActionSupport;**

**public class ScoreAction extends ActionSupport {**

**private List<Student> students;**

**public List<Student> getStudents() {**

**return students;**

**}**

**public void setStudents(List<Student> students) {**

**this.students = students;**

**}**

**@Override**

**public String execute() throws Exception {**

**int size = students.size();**

**for(int i = 0; i<size; i++){**

**Student st = students.get(i);**

**Score score = st.getScore();**

**double aveScore = (score.getCcScore()+score.getJ2eeScore()+score.getJavaScore())/3.0;**

**BigDecimal b = new BigDecimal(aveScore);**

**aveScore = b.setScale(1, BigDecimal.ROUND\_HALF\_UP).doubleValue();**

**score.setAveScore(aveScore);**

**}**

**return super.SUCCESS;**

**}**

**}**

scores.jsp

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"UTF-8"*%>

<%@ taglib prefix=*"s"* uri=*"/struts-tags"*%>

<%@ taglib uri=*"/struts-dojo-tags"* prefix=*"ss"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme() + "://"

+ request.getServerName() + ":" + request.getServerPort()

+ path + "/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>输入学生成绩</title>

<s:head theme=*"xhtml"* />

<ss:head parseContent=*"true"* />

<style type=*"text/css"*>

**table** {

border-collapse: *collapse*;

border: *1px solid #000*;

}

**th,td** {

border: *1px solid #000*;

}

</style>

</head>

<body>

<s:form theme=*"simple"* action=*"showscores"*>

<table>

<thead>

<tr>

<th align=*"center"*>

姓名

</th>

<th align=*"center"*>

学号

</th>

<th align=*"center"*>

Java成绩

</th>

<th align=*"center"*>

C语言成绩

</th>

<th align=*"center"*>

J2EE成绩

</th>

</tr>

</thead>

<tbody>

<s:iterator value=*"new int[4]"* status=*"st"*>

<tr>

<td>

<s:textfield name=*"%{'students['+#st.index+'].name'}"*>

</s:textfield>

</td>

<td>

<s:textfield name=*"%{'students['+#st.index+'].number'}"*>

</s:textfield>

</td>

<td>

<s:textfield name=*"%{'students['+#st.index+'].score.javaScore'}"*>

</s:textfield>

</td>

<td>

<s:textfield name=*"%{'students['+#st.index+'].score.ccScore'}"*>

</s:textfield>

</td>

<td>

<s:textfield name=*"%{'students['+#st.index+'].score.j2eeScore'}"*>

</s:textfield>

</td>

</tr>

</s:iterator>

</tbody>

</table>

<s:submit value=*"提交"*></s:submit>

</s:form>

</body>

</html>

**showScores.jsp**

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"UTF-8"*%>

<%@ taglib prefix=*"s"* uri=*"/struts-tags"*%>

<%@ taglib uri=*"/struts-dojo-tags"* prefix=*"ss"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme() + "://"

+ request.getServerName() + ":" + request.getServerPort()

+ path + "/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>显示学生成绩</title>

<s:head theme=*"xhtml"* />

<ss:head parseContent=*"true"* />

<style type=*"text/css"*>

**table** {

border-collapse: *collapse*;

border: *1px solid #000*;

}

**th,td** {

border: *1px solid #000*;

}

</style>

</head>

<body>

<table>

<thead>

<tr>

<th align=*"center"*>

姓名

</th>

<th align=*"center"*>

学号

</th>

<th align=*"center"*>

Java成绩

</th>

<th align=*"center"*>

C语言成绩

</th>

<th align=*"center"*>

J2EE成绩

</th>

<th align=*"center"*>

平均成绩

</th>

</tr>

</thead>

<tbody>

<s:iterator value=*"students"* status=*"st"*>

<tr>

<td align=*"center"*>

<s:property value=*"name"* />

</td>

<td>

<s:property value=*"number"* />

</td>

<td align=*"center"*>

<s:property value=*"score.javaScore"* />

</td>

<td align=*"center"*>

<s:property value=*"score.ccScore"* />

</td>

<td align=*"center"*>

<s:property value=*"score.j2eeScore"* />

</td>

<td align=*"center"*>

<s:property value=*"score.aveScore"* />

</td>

</tr>

</s:iterator>

</tbody>

</table>

</body>

</html>

**struts.xml**

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

<!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">

<struts>

<package name=*"default"* extends=*"struts-default"*>

<action name=*"scores"*>

<result>/scores.jsp</result>

</action>

<action name=*"showscores"* class=*"actions.ScoreAction"*>

<result name=*"success"*>/showScores.jsp</result>

</action>

<action name=*"cellphone"*>

<result>/cellPhone.jsp</result>

</action>

<action name=*"showscellphone"* class=*"actions.CellPhoneAction"*>

<result name=*"success"*>/showCellPhone.jsp</result>

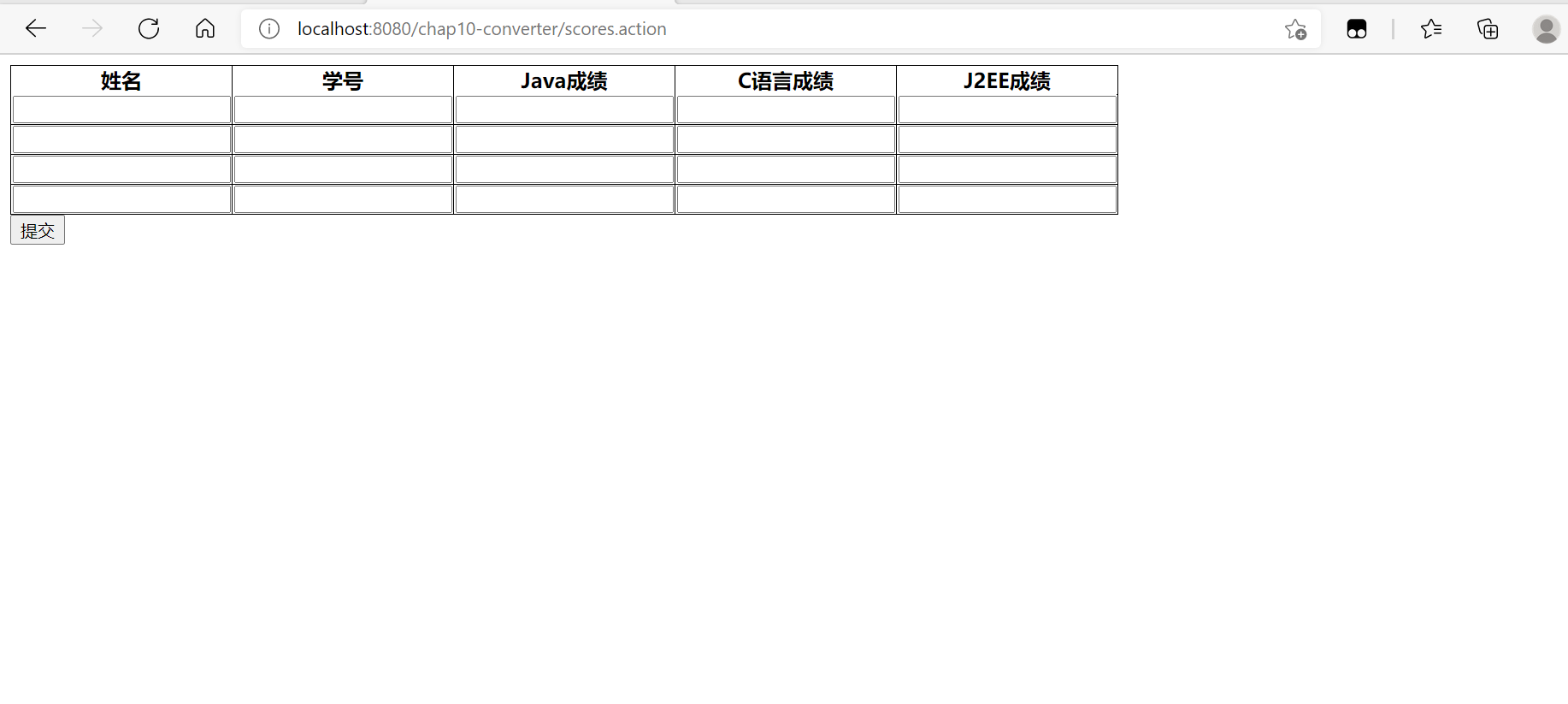
<result name = *"input"*>/cellPhone.jsp</result>

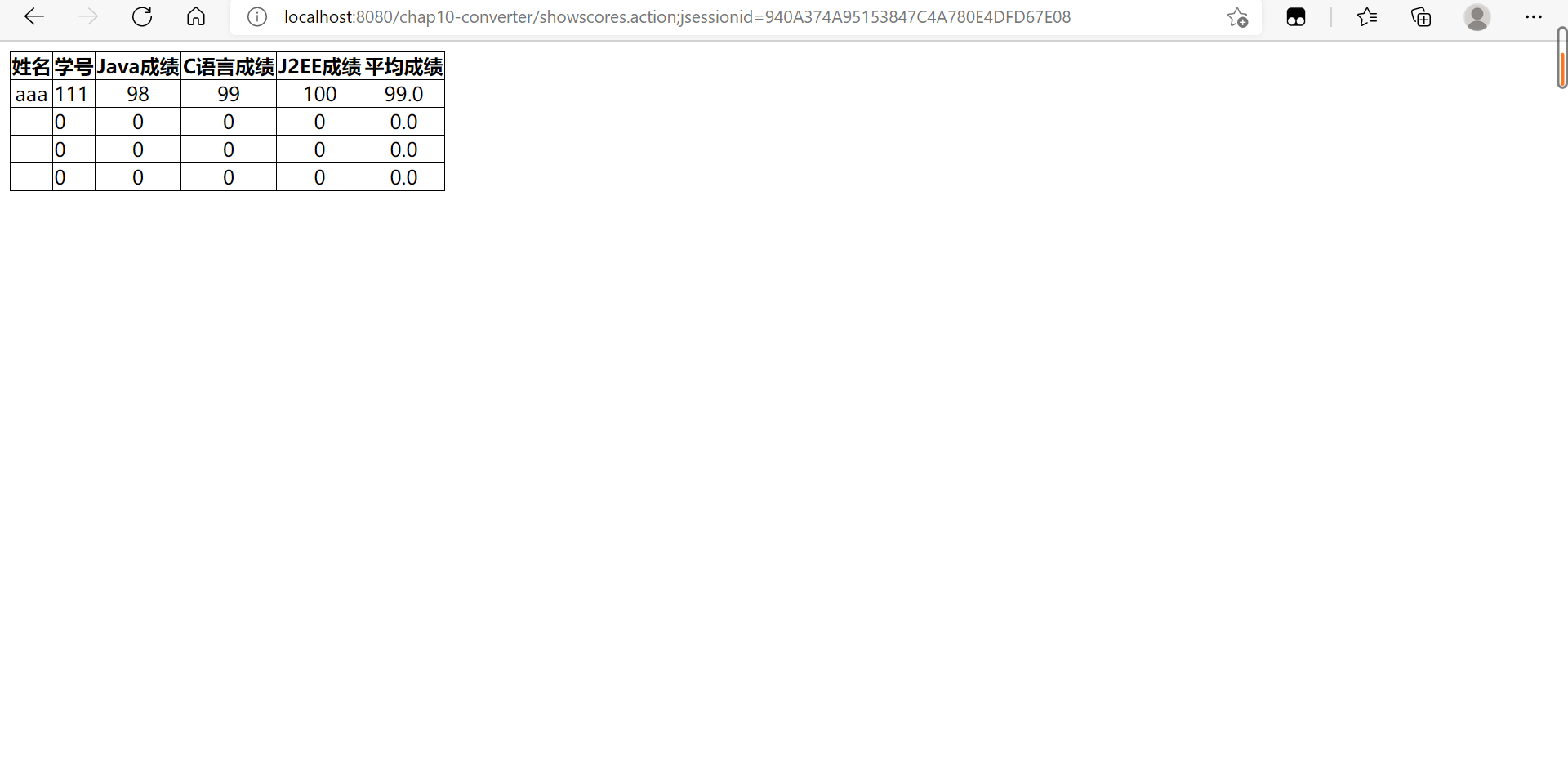
</action>

</package>

</struts>

**运行结果：**





2. 练习书中 P152 页代码 8-51 到代码 8-53 学生信息注册页面的显示，并自己编写一个 Action 及 JSP 完成对注册信息的接收与显示。

**8. 校验框架使用**

1. 练习书中 P155 页代码 8-56 到 8-60 学生注册信息的校验。

关键代码：

success.jsp：

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"utf-8"*%>

<%@ taglib prefix=*"s"* uri=*"/struts-tags"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+path+"/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>My JSP 'success.jsp' starting page</title>

<meta http-equiv=*"pragma"* content=*"no-cache"*>

<meta http-equiv=*"cache-control"* content=*"no-cache"*>

<meta http-equiv=*"expires"* content=*"0"*>

<meta http-equiv=*"keywords"* content=*"keyword1,keyword2,keyword3"*>

<meta http-equiv=*"description"* content=*"This is my page"*>

<!--

<link rel="stylesheet" type="text/css" href="styles.css">

-->

</head>

<body>

学生信息如下：

<hr />

姓名：

<s:property value=*"studentName"* />

<br>

学号：

<s:property value=*"studentNo"* />

<br>

年龄：

<s:property value=*"age"* />

<br>

<hr />

</body>

</html>

studentRegister.jsp：

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"utf-8"*%>

<%@ taglib prefix=*"s"* uri=*"/struts-tags"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme() + "://"

+ request.getServerName() + ":" + request.getServerPort()

+ path + "/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>学生信息注册</title>

</head>

<body>

<s:form action = *"Register"*>

<s:textfield label=*"姓名"* name=*"studentName"*></s:textfield>

<s:textfield label=*"学号"* name=*"studentNo"*></s:textfield>

<s:textfield label=*"年龄"* name=*"age"*></s:textfield>

<s:submit value=*"提交"* align=*"center"*></s:submit>

</s:form>

</body>

</html>

StudentRegisterAction.java：

**package** actions;

**import** com.opensymphony.xwork2.ActionSupport;

**public** **class** StudentRegisterAction **extends** ActionSupport {

**private** String studentName;

**private** String studentNo;

**private** **int** age;

**public** String getStudentName() {

**return** studentName;

}

**public** **void** setStudentName(String studentName) {

**this**.studentName = studentName;

}

**public** String getStudentNo() {

**return** studentNo;

}

**public** **void** setStudentNo(String studentNo) {

**this**.studentNo = studentNo;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

@Override

**public** String execute() **throws** Exception {

**return** *SUCCESS*;

}

}

struts.xml：

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

<!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">

<struts>

<package name=*"default"* namespace=*"/"* extends=*"struts-default"*>

<action name=*"WelcomeRegister"*>

<result>/studentRegister.jsp</result>

</action>

<action name=*"Register"* class=*"actions.StudentRegisterAction"*>

<result name=*"success"*>/success.jsp</result>

<result name=*"input"*>/studentRegister.jsp</result>

</action>

<action name=*"WelcomeVisitorRegister"*>

<result>/visitor/studentVisitorRegister.jsp</result>

</action>

<action name=*"VisitorRegister"* class=*"actions.VisitorRegisterAction"*>

<result name=*"success"*>/success.jsp</result>

<result name=*"input"*>/visitor/studentVisitorRegister.jsp</result>

</action>

<action name=*"WelcomeMyVisitorRegister"*>

<result>/visitor/studentMyVisitorRegister.jsp</result>

</action>

<action name=*"MyVisitorRegister"* class=*"actions.MyValidatorAction"*>

<result name=*"success"*>/success.jsp</result>

<result name=*"input"*>/visitor/studentMyVisitorRegister.jsp</result>

</action>

</package>

</struts>

validators.xml：

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

<!DOCTYPE validators PUBLIC

"-//OpenSymphony Group//XWork Validator Config 1.0//EN"

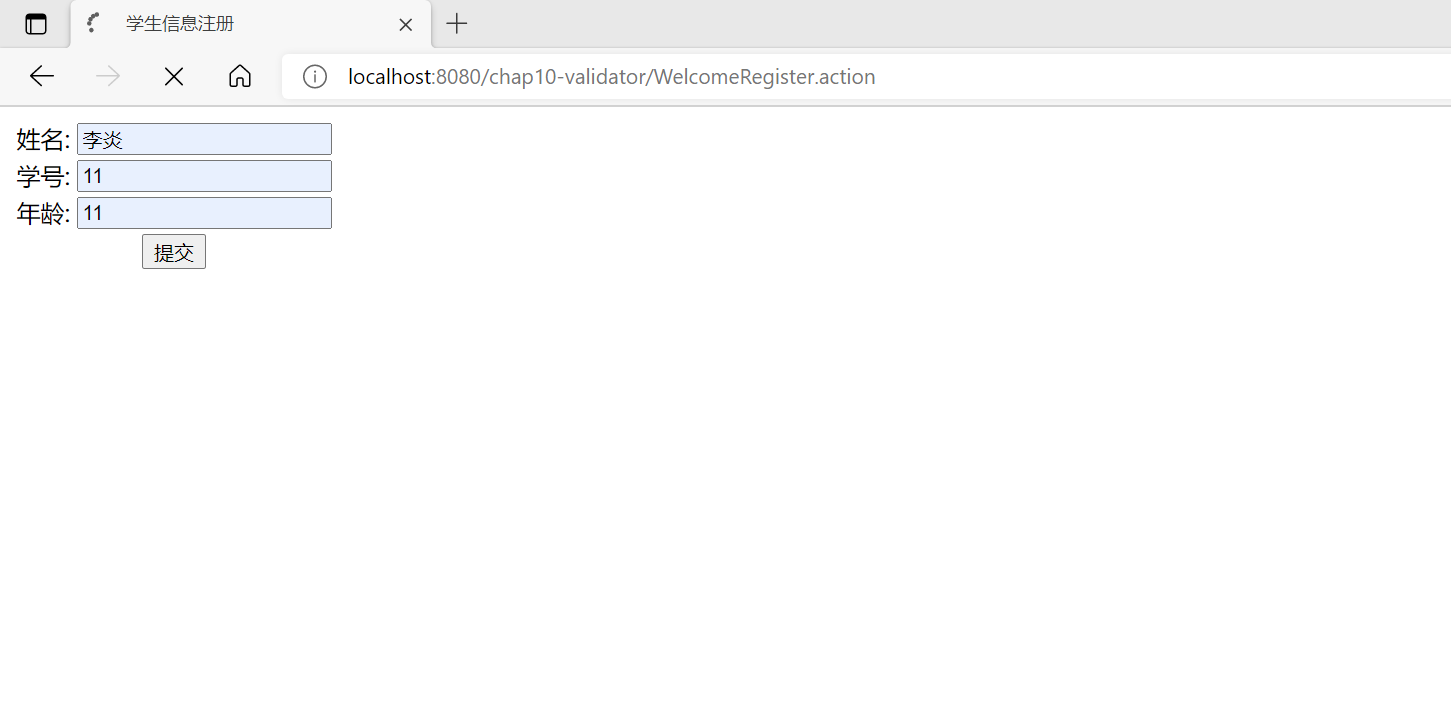
"http://www.opensymphony.com/xwork/xwork-validator-config-1.0.dtd">

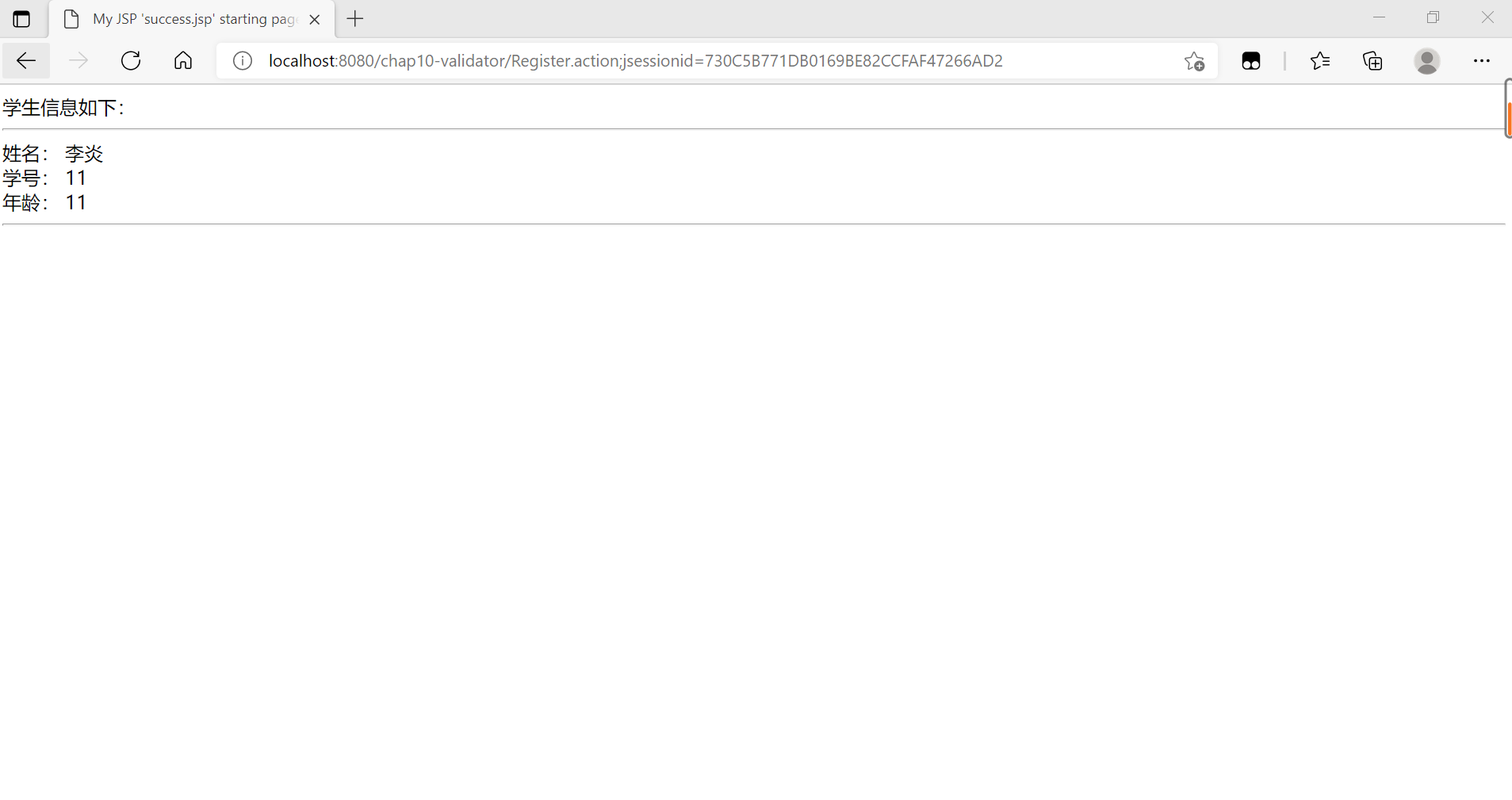
<validators>

<validator name=*"myvalidator"* class=*"actions.MyValidator"*/>

</validators>

运行结果：





2. 练习书中 P159 页代码 8-66 到 8-71 的自定义转换器的使用。

核心代码：

StudentRegisterAction.java

**package** actions;

**import** com.opensymphony.xwork2.ActionSupport;

**public** **class** StudentRegisterAction **extends** ActionSupport {

**private** String studentName;

**private** String studentNo;

**private** **int** age;

**public** String getStudentName() {

**return** studentName;

}

**public** **void** setStudentName(String studentName) {

**this**.studentName = studentName;

}

**public** String getStudentNo() {

**return** studentNo;

}

**public** **void** setStudentNo(String studentNo) {

**this**.studentNo = studentNo;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

@Override

**public** String execute() **throws** Exception {

**return** *SUCCESS*;

}

}

StudentRegisterAction-validation.xml

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

<!DOCTYPE validators PUBLIC

"-//Apache Struts//XWork Validator 1.0.2//EN"

"http://struts.apache.org/dtds/xwork-validator-1.0.2.dtd">

<validators>

<field name=*"schoolName"*>

<field-validator type=*"myvalidator"*>

<param name=*"school"*>西北农林科技大学</param>

<message>选择的学校必须是西北农林科技大学</message>

</field-validator>

</field>

</validators>

**struts.xml**

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

<!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">

<struts>

<package name=*"default"* namespace=*"/"* extends=*"struts-default"*>

<action name=*"WelcomeRegister"*>

<result>/studentRegister.jsp</result>

</action>

<action name=*"Register"* class=*"actions.StudentRegisterAction"*>

<result name=*"success"*>/success.jsp</result>

<result name=*"input"*>/studentRegister.jsp</result>

</action>

<action name=*"WelcomeVisitorRegister"*>

<result>/visitor/studentVisitorRegister.jsp</result>

</action>

<action name=*"VisitorRegister"* class=*"actions.VisitorRegisterAction"*>

<result name=*"success"*>/success.jsp</result>

<result name=*"input"*>/visitor/studentVisitorRegister.jsp</result>

</action>

<action name=*"WelcomeMyVisitorRegister"*>

<result>/visitor/studentMyVisitorRegister.jsp</result>

</action>

<action name=*"MyVisitorRegister"* class=*"actions.MyValidatorAction"*>

<result name=*"success"*>/success.jsp</result>

<result name=*"input"*>/visitor/studentMyVisitorRegister.jsp</result>

</action>

</package>

</struts>

studentRegister.jsp

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"utf-8"*%>

<%@ taglib prefix=*"s"* uri=*"/struts-tags"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme() + "://"

+ request.getServerName() + ":" + request.getServerPort()

+ path + "/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>学生信息注册</title>

</head>

<body>

<s:form action = *"Register"*>

<s:textfield label=*"姓名"* name=*"studentName"*></s:textfield>

<s:textfield label=*"学号"* name=*"studentNo"*></s:textfield>

<s:textfield label=*"年龄"* name=*"age"*></s:textfield>

<s:submit value=*"提交"* align=*"center"*></s:submit>

</s:form>

</body>

</html>

success.jsp

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"utf-8"*%>

<%@ taglib prefix=*"s"* uri=*"/struts-tags"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+path+"/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>My JSP 'success.jsp' starting page</title>

<meta http-equiv=*"pragma"* content=*"no-cache"*>

<meta http-equiv=*"cache-control"* content=*"no-cache"*>

<meta http-equiv=*"expires"* content=*"0"*>

<meta http-equiv=*"keywords"* content=*"keyword1,keyword2,keyword3"*>

<meta http-equiv=*"description"* content=*"This is my page"*>

<!--

<link rel="stylesheet" type="text/css" href="styles.css">

-->

</head>

<body>

学生信息如下：

<hr />

姓名：

<s:property value=*"studentName"* />

<br>

学号：

<s:property value=*"studentNo"* />

<br>

年龄：

<s:property value=*"age"* />

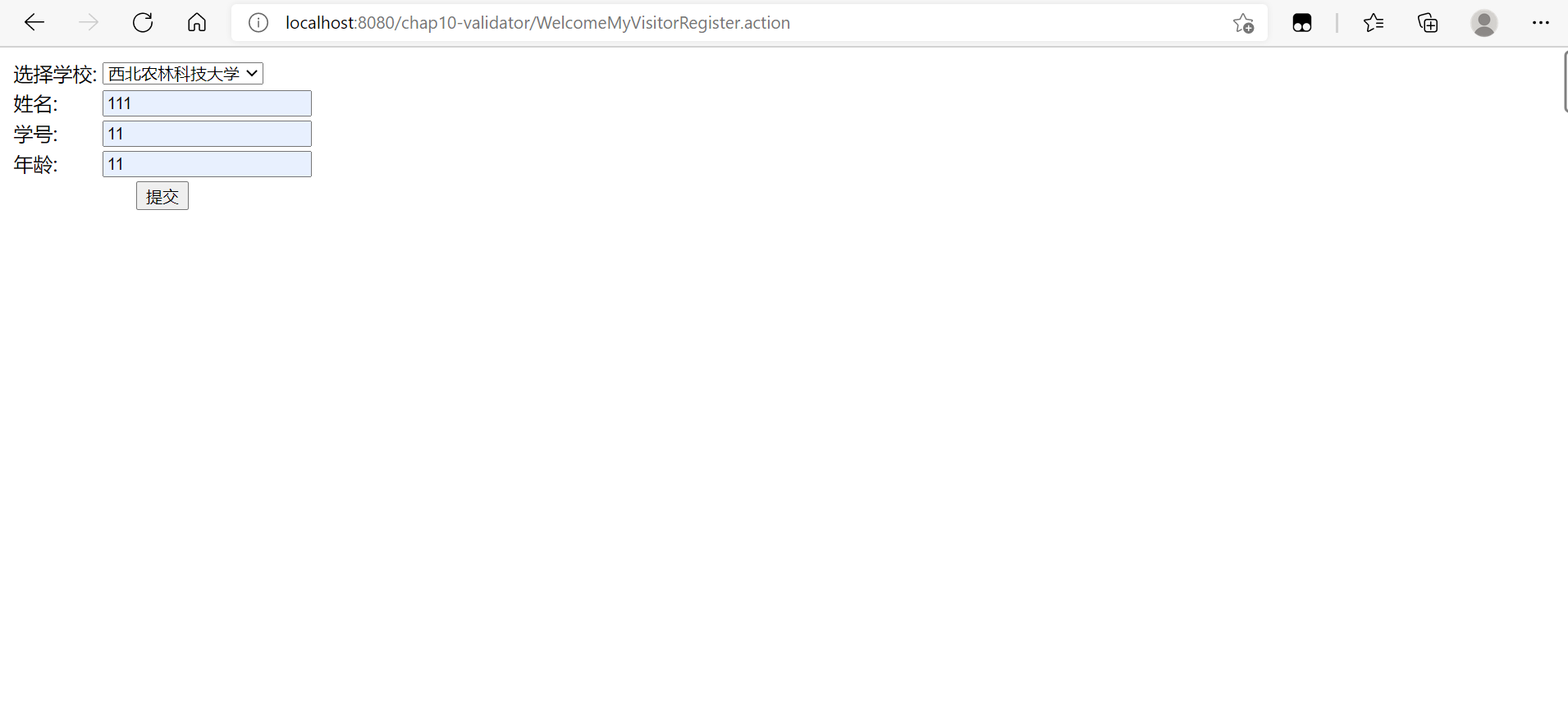
<br>

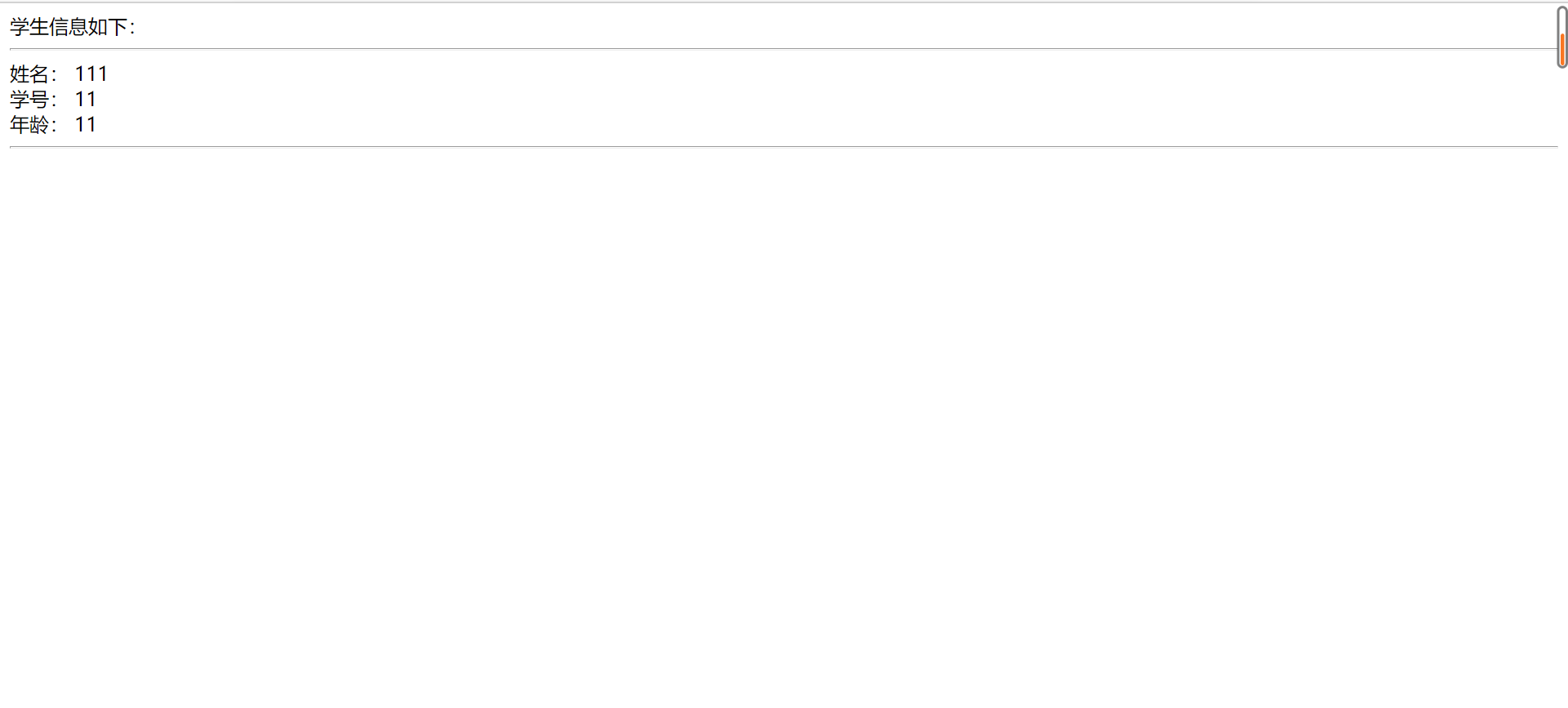
<hr />

</body>

</html>

运行结果：





**扩展作业**

将实习三中的控制器及视图使用 Struts 及其标签实现

**总结：**

通过对struts框架的学习，我更加体会到了解它的运行原理和执行流程的重要性，以及Action组件和Struts-config.xml这个核心配置文件的使用。同时，Action ActionForm biz dao 这几个层之间的关系，本质上还是之前学过的MVC的设计模式，这也是对之前所学的一个巩固。