12.26.30

Jsp访问页面时输入http://localhost:8080/工程名/项目名

**1.使用jsp完成一个java web项目，实现用户登录功能，三个页面**

**登录页面login.jsp,登录逻辑处理页面loginSubmit.jsp，欢迎页面welcome.jsp。**

**用户在登录页面输入用户名和密码，如果用户名为”admin”，密码为”123”，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。（无须连接数据库）**

步骤：建一个web project工程，建三个jsp文件

1. 登录界面：login.jsp

<body>

欢迎来到登录页面！

<form action=*"loginSubmit.jsp"* method=*"post"*>

请输入你的账号:<input name=*"userName"*type=*"text"*><BR>

请输入你的密码:<input name=*"password"*type=*"password"*><BR>

<input type=*"submit"* value=*"登录"*>

</form>

</body>

（2）处理页面：loginSubmit.jsp

<body>

<%

//从表单中获取提交的用户名密码

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

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

//验证用户名和密码

**if**(userName.equals("admin") && password.equals("123")){

response.sendRedirect("welcome.jsp");

}

**else**{

response.sendRedirect("login.jsp");

}

%>

</body>

（3）登录界面：body中写登录成功

**2.使用jsp完成一个java web项目，实现用户登录功能，三个页面**

**登录页面login.jsp,登录逻辑处理页面loginSubmit.jsp，欢迎页面welcome.jsp。**

**用户在登录页面输入用户名和密码，如果数据库中存在，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。（须连接数据库，sqlserver和access均可）**

步骤：login.jsp和welcome.jsp代码一样

loginSubmit.jsp：<body>

<%

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

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

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection conn = DriverManager.getConnection("jdbc:odbc:Employ","sasa","123456");

Statement stat = conn.createStatement();

ResultSet rs=stat.executeQuery("select count(\*) from [Employ].[dbo].[1] where username='"+username+"' and password='"+password+"' ");

rs.next();

**int** count=rs.getInt(1);

**if**(count==1)

{

response.sendRedirect("welcome.jsp");

}

**else**

{

response.sendRedirect("login.jsp");

}

%>

</body>

**3.使用jsp+servlet完成一个java web项目，实现用户登录功能，两个页面，一个servlet。**

**登录页面login.jsp,欢迎页面welcome.jsp,登录逻辑loginservlet。用户在登录页面输入用户名和密码，如果用户名为”admin”，密码为”123”，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。**

步骤：建一个web project类，建两个个jsp页面：welcome.jsp和login.jsp（form action=“loginservlet”），建一个名为servlet的包里面建名为loginservlet的servlet

Welcome和login代码同一

<form action=*"/2/servlet/loginservlet"* method=*"post"*>

1. loginservlet.Java代码：

在dopost代码内写：

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

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

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

//验证用户名和密码

if(("admin").equals(username) && ("123").equals(password)){

response.sendRedirect("http://localhost:8080/2/welcome.jsp");//地址写死

}

else{

response.sendRedirect("http://localhost:8080/2/login.jsp");

}}}

**4.使用jsp+servlet完成一个java web项目，实现用户登录功能，两个页面，一个servlet。**

登录页面login.jsp,欢迎页面welcome.jsp,登录逻辑loginservlet。用户在登录页面输入用户名和密码，如果数据库中存在，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。（须连接数据库，sqlserver和access均可）

1. loginservlet.Java

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html；charset=utf-8");

response.setCharacterEncoding("utf-8");

request.setCharacterEncoding("utf-8");

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

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

try {

//1.加载驱动

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

//2.得到连接

Connection conn = DriverManager.getConnection("jdbc:odbc:Employ","sasa","123456");

//3.创建PreparedStatment 用于传送sql查询语句

Statement stat = conn.createStatement();

ResultSet rs1=stat.executeQuery("select count(\*) from [Employ].[dbo].[1] where userName='"+username+"' and password='"+password+"' ");

rs1.next();

int count=rs1.getInt(1);

if(count==1)

{

response.sendRedirect("http://localhost:8080/4/welcome.jsp");

}

else

{

response.sendRedirect("http://localhost:8080/4/login.jsp");

}

} catch (Exception e) {

e.printStackTrace();

// TODO: handle exception

}}

}

**5.使用jsp完成一个java web项目，实现用户登录功能，三个页面**

**登录页面login.jsp,登录逻辑处理页面loginSubmit.jsp，欢迎页面welcome.jsp。**

**用户在登录页面输入用户名和密码，前台页面使用js进行验证，如果用户名或密码为空，则提示用户输入用户名和密码；如果用户名为”admin”，密码为”123”，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。（无须连接数据库）**

步骤：loginSubmit.jsp和welcome.jsp代码同1

Login.jsp：

<body>

<script type=*"text/javascript"*>

**function** checkform(){

**if** (login.username.value==""){

alert("请输入账号。");

**return**;

}

**if**(login.password.value==""){

alert("请输入密码。");

**return**;

}

document.login.submit();

}

</script>

欢迎来到登录页面！

<form name=*"login"* action=*"loginSubmit.jsp"* method=*"post"*>

请输入你的账号:<input name=*"username"*type=*"text"*><BR>

请输入你的密码:<input name=*"password"*type=*"password"*><BR>

<input type=*"button"* onclick="checkform()"value=*"登录"*>

</form>

</body>

**6.使用jsp完成一个java web项目，实现用户登录功能，三个页面**

**登录页面login.jsp,登录逻辑处理页面loginSubmit.jsp，欢迎页面welcome.jsp。**

**用户在登录页面输入用户名和密码，前台页面使用js进行验证，如果用户名或密码为空，则提示用户输入用户名和密码；如果密码小于6位，则提示输入的密码长度不够；如果用户名为”admin”，密码为”12345678”，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。（无须连接数据库）**

步骤同上，就是login.jsp代码在请输入密码下加上

**if** (login.password.value.length < 8)

{

alert("密码至少输入8个字符。");

**return**;

}

**7.使用jsp完成一个java web项目，实现用户登录功能，三个页面**

**登录页面login.jsp,登录逻辑处理页面loginSubmit.jsp，欢迎页面welcome.jsp。**

**用户在登录页面输入用户名和密码，如果用户名为”admin”，密码为”123”，则登录成功跳转到欢迎页面，欢迎页面上输出：“欢迎你！”+用户名。如果失败，停留在登录页面。（无须连接数据库）**

（1）Login.jsp代码同1；

（2）LoginSubmit.jsp：<body>

<%

//从表单中获取提交的用户名密码

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

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

//验证用户名和密码

**if**(username.equals("admin") && password.equals("123")){

%>

<jsp:forward page=*"welcome.jsp"*></jsp:forward>

<%}

**else**{

%>

<jsp:forward page=*"login.jsp"*></jsp:forward>

<%

}

%>

</body>

1. welcome.jsp：

<body>

<%

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

out.println("欢迎你！" + username);

%>

</body>

**8.使用jsp完成一个java web项目，实现用户注册功能，两个页面**

**注册页面register.jsp,注册信息页面registerinfo.jsp。**

**用户在注册页面输入用户名、密码，学号、姓名、年龄、专业后点击注册按钮，页面跳转到注册信息页面并显示注册信息。要求姓名、专业为中文输入、中文显示。（在jsp页面中解决中文乱码问题）**

1. register.jsp：

<body>

欢迎注册系统！

<form name=*"register"* action=*"registerinfo.jsp"* method=*"post"*>

账号:<input name=*"account"*type=*"text"*><BR>

密码:<input name=*"password"*type=*"password"*><BR>

学号:<input name=*"stunum"*type=*"text"*><BR>

姓名:<input name=*"realname"*type=*"text"*><BR>

年龄:<input name=*"age"*type=*"text"*><BR>

专业:<input name=*"major"*type=*"text"*><BR>

<input type=*"submit"* value=*"注册"*>

</form>

</body>

1. Registerinfo.jsp

<body>

<h2>注册成功！你的注册信息如下：</h2>

<%

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

out.println("账号:" + account);

out.println("<BR/>");

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

out.println("密码:" +password);

out.println("<BR/>");

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

out.println("学号:" + stunum);

out.println("<BR/>");

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

out.println("年龄:" + age);

out.println("<BR/>");

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

realname = **new** String(realname.getBytes("ISO-8859-1"),"gb2312");

out.println("姓名:" + realname);

out.println("<BR/>");

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

major = **new** String(major.getBytes("ISO-8859-1"),"gb2312");

out.println("专业:" + major);

%>

</body>

**9.使用jsp完成一个java web项目，实现用户注册功能，两个页面**

**注册页面register.jsp,注册信息页面registerinfo.jsp。**

**用户在注册页面输入用户名、密码，学号、姓名、年龄、专业后点击注册按钮，页面跳转到注册信息页面并显示注册信息。要求姓名、专业为中文输入、中文显示。（使用过滤器解决中文乱码问题）**

步骤：建一个包filter在建一个servlet名为EncodingFilter；建两个jsp：register.jsp和registerinfo.jsp；配置web.xml

（1）web.xml代码：<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app version=*"3.0"*

xmlns=*"http://java.sun.com/xml/ns/javaee"*

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

xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"*>

<servlet>

<description>This is the description of my J2EE component</description>

<display-name>This is the display name of my J2EE component</display-name>

<servlet-name>EncodingFilter</servlet-name>

<servlet-class>filter.EncodingFilter</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>EncodingFilter</servlet-name>

<url-pattern>/servlet/EncodingFilter</url-pattern>

</servlet-mapping>

<filter>

<filter-name>EncodingFilter</filter-name>

<filter-class>filter.EncodingFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>EncodingFilter</filter-name>

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

</filter-mapping>

</web-app>

（2）register.jsp代码：

<body>

欢迎注册系统！

<form name=*"register"* action=*"registerinfo.jsp"* method=*"post"*>

账号:<input name=*"account"*type=*"text"*><BR>

密码:<input name=*"password"*type=*"password"*><BR>

学号:<input name=*"stunum"*type=*"text"*><BR>

姓名:<input name=*"realname"*type=*"text"*><BR>

年龄:<input name=*"age"*type=*"text"*><BR>

专业:<input name=*"major"*type=*"text"*><BR>

<input type=*"submit"* value=*"注册"*>

</form>

</body>

（3）registerinfo代码：

<body>

<h2>注册成功！你的注册信息如下：</h2>

<%

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

out.println("账号:" + account);

out.println("<BR/>");

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

out.println("密码:" +password);

out.println("<BR/>");

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

out.println("学号:" + stunum);

out.println("<BR/>");

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

out.println("年龄:" + age);

out.println("<BR/>");

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

out.println("姓名:" + realname);

out.println("<BR/>");

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

out.println("专业:" + major);

%>

</body>

（4）EncodingFilter代码：**package** filter;

**import** java.io.IOException;

**import** javax.servlet.Filter;

**import** javax.servlet.FilterChain;

**import** javax.servlet.FilterConfig;

**import** javax.servlet.ServletException;

**import** javax.servlet.ServletRequest;

**import** javax.servlet.ServletResponse;

**public** **class** EncodingFilter **implements** Filter {

**public** **void** init(FilterConfig config) **throws** ServletException {}

**public** **void** destroy() {}

**public** **void** doFilter(ServletRequest request, ServletResponse response,

FilterChain chain) **throws** IOException, ServletException {

request.setCharacterEncoding("gb2312");

chain.doFilter(request, response);

}

}

**10.使用jsp完成一个java web项目，实现用户登录功能，三个页面**

**登录页面login.jsp,登录逻辑处理页面loginSubmit.jsp，欢迎页面welcome.jsp。**

**用户在登录页面输入用户名和密码，如果用户名为”admin”，密码为”123”，则登录成功跳转到欢迎页面，如果失败，停留在登录页面。使用过滤器完成只有登录用户才能访问welcome页面的功能，非登录用户直接输入welcome.jsp的url将跳转到登陆页面。**

Welcome.jsp代码同1；login.jsp同1

LoginSubmit.jsp同7

步骤：建一个filter的包，建一个LoginFilter的servlet，配置web.xml

1. LoginFilter代码：

**package** filter;

**import** java.io.IOException;

**import** javax.servlet.Filter;

**import** javax.servlet.FilterChain;

**import** javax.servlet.FilterConfig;

**import** javax.servlet.ServletException;

**import** javax.servlet.ServletRequest;

**import** javax.servlet.ServletResponse;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** javax.servlet.http.HttpSession;

**public** **class** LoginFilter **implements** Filter {

**public** **void** init(FilterConfig filterConfig) **throws** ServletException {

// **TODO** Auto-generated method stub

}

**public** **void** doFilter(ServletRequest request, ServletResponse response,

FilterChain chain) **throws** IOException, ServletException {

// 获得在下面代码中要用的request,response,session对象

HttpServletRequest servletRequest = (HttpServletRequest) request;

HttpServletResponse servletResponse = (HttpServletResponse) response;

HttpSession session = servletRequest.getSession();

// 获得用户请求的URI

String path = servletRequest.getRequestURI();

String userName = (String) session.getAttribute("userName");

/\*创建类Constants.java，里面写的是无需过滤的页面

for (int i = 0; i < Constants.NoFilter\_Pages.length; i++) {

if (path.indexOf(Constants.NoFilter\_Pages[i]) > -1) {

chain.doFilter(servletRequest, servletResponse);

return;

}

}\*/

// 登陆页面无需过滤

**if**(path.indexOf("/login.jsp") > -1) {

chain.doFilter(servletRequest, servletResponse);

**return**;

}

// 判断如果没有取到信息,就跳转到登陆页面

**if** (userName == **null** || "".equals(userName)) {

// 跳转到登陆页面

servletResponse.sendRedirect("/caozuo10/login.jsp");

} **else** {

// 已经登陆,继续此次请求

chain.doFilter(request, response);

}

}

**public** **void** destroy() {

// **TODO** Auto-generated method stub

}}

（2）web.xml代码：...

*http://java.sun.com/xml/ns/j2ee/web-app\_2\_4.xsd"*>

<!-- 配置登录过滤器 -->

<filter>

<filter-name>login</filter-name>

<filter-class>filter.LoginFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>login</filter-name>

<url-pattern>/welcome.jsp</url-pattern>

</filter-mapping>

<welcome-file-list>

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

</welcome-file-list>

</web-app>

**11.页面1中表单内输入一个数字N,提交，能够在另一个页面打印N个“欢迎”字符**

（1）Resigter.jsp：<body>

请输入一个数字！

<form action=*"registerinfo.jsp"* method=*"post"*>

<input name=*"number"*type=*"text"*><BR>

<input type=*"submit"* value=*"提交"*>

</form>

</body>

1. resigterinfo.jsp：

<body>

<%

String str = request.getParameter("number");

**int** number = Integer.parseInt(str);

**for** (**int** i = 1; i <= number; i++) {

out.println("欢迎<br>");

}

%>

</body>

**13.用户访问首页，用一个下拉菜单选择背景颜色，提交，到达欢迎页面，背景颜色**

(1)Color.jsp:<body>

<%

**if**(request.getCookies()!=**null**)

{

Cookie[] cookies = request.getCookies();

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

{

**if**(cookies[i].getName().equals("color\_test"))

{

%>

<jsp:forward page = *"color1.jsp"*></jsp:forward>

<%

}

}

}

%>

<form action = *"color1.jsp"*>

<select name = *"color"*>

<option value = *"pink"*>pink</option>

<option value = *"black"*>black</option>

<option value = *"blue"*>blue</option>

<option value = *"gray"*>gray</option>

<option value = *"green"*>green</option>

<option value = *"navy"*>navy</option>

<option value = *"purple"*>purple</option>

<option value = *"maroon"*>maroon</option>

</select>

<input type=*"submit"* value=*"提交"*>

</form>

</body>

(2)color1.jsp:

<%

String color = **null**;

Cookie[] cookies = request.getCookies();

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

{

**if**(cookies[i].getName().equals("color\_test"))

{

color = cookies[i].getValue();

out.println("颜色:"+color);

}

}

**if**(request.getParameter("color")!=**null**)

{

color = request.getParameter("color");

Cookie cookie = **new** Cookie("color\_test",color);

cookie.setMaxAge(600);

response.addCookie(cookie);

}

%>

<body bgcolor = <%=color %>>

<a href = *"clearcookie.jsp"*>清除cookie</a>

</body>

(3)clearcookie:<body>

<%

Cookie[] cookies = request.getCookies();

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

{

**if**(cookies[i].getName().equals("color\_test"))

{

cookies[i].setMaxAge(-1);

response.addCookie(cookies[i]);

}

}

%>

</body>

**14.编写图书模糊界面，输入图书名称的模糊资料，界面下方显示图书信息，提交给servlet完成**

建一个servlet类建loginservlet

（1）Select.jsp:<body>

<form action=*"bookservlet"* method=*"post"*>

输入图书名称:<input type=*"text"* name=*"name"*><BR>

<input type=*"submit"* value=*"查询"*>

</form>

</body>

1. loginservlet.java代码：

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

response.setCharacterEncoding("utf-8");

request.setCharacterEncoding("utf-8");

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

**try** {

//1.加载驱动

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

//2.得到连接

Connection conn = DriverManager.*getConnection*("jdbc:odbc:Employ","sasa","123456");

//3.创建PreparedStatment 用于传送sql查询语句

Statement stat = conn.createStatement();

String sql = "SELECT ID,bookname,bookprice FROM [Employ].[dbo].[book] WHERE bookname like '%"+name+"%'";

ResultSet rs1 = stat.executeQuery(sql);

**while**(rs1.next()){

String ID = rs1.getString("ID");

String bookname = rs1.getString("bookname");

String bookprice = rs1.getString("bookprice");

System.*out*.println("书号:"+ ID + " 书名:" + bookname + " 价格:"+ bookprice);

}

} **catch** (Exception e) {

e.printStackTrace();

// **TODO**: handle exception

}

}

**15.为网站配置欢迎页面“index.html”，如果找不到则为“indes.jsp”,进行测试。**

步骤：建一个jsp和html，并在各自的body中写index.html和index.jsp

**17.（jsp，javabean，servlet(Dao)合一起做登录界面）编写一个登陆页面，输入学号和姓名，在数据库中进行验证，如果验证通过,则在另一个页面中显示顾客的姓名。要求使用javabean来封装顾客信息，使用dao来查询数据库。**

步骤：建一个包名为dao里面建User和UserDao两个class；在建一个servlets包里面建loginservlet的servlet；两个jsp页面：login.jsp和welcome.jsp代码同一

（1）User.Java：userName和password及他们的get，set函数

（2）UserDao.Java代码：**package** dao;

**import** java.sql.\*;

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** UserDao **extends** User{

**public** ResultSet quaryAllUser()**throws** Exception{

Connection conn = **null**;

ArrayList Users = **new** ArrayList();

**try**{

//获取连接

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

String url="jdbc:odbc:Employ";

conn=DriverManager.*getConnection*(url,"sasa", "123456");

//运行sql语句

String sql="SELECT \* FROM [Employ].[dbo].[Users]";

Statement stat=conn.createStatement();

ResultSet rs=stat.executeQuery(sql);

**return** rs;

}**catch**(SQLException e){

e.printStackTrace();

}

**return** **null**;

}

**public** **static** **void** main(String args[]) **throws** Exception{

UserDao dao=**new** UserDao();

ResultSet rs=dao.quaryAllUser();

**while**(rs.next()){

System.*out*.println(rs.getString("userName"));

}

}

}

1. loginservlet代码：写在dopost里面

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

**int** i=0;

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

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

UserDao dao=**n ew** UserDao();

ResultSet rs = **null**;

**try** {

rs = dao.quaryAllUser();

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**try** {

**while**(rs.next()){

String ss=rs.getString("userName");

String aa=rs.getString("password");

**if**(username.equals(ss)&&password.equals(aa)){

i++;

response.sendRedirect("http://localhost:8080/prj17/welcome.jsp");

}

}

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**if**(i==0)

response.sendRedirect("http://localhost:8080/prj17/login.jsp");

}

1. 注册功能

有zhuce.jsp和welcome.jsp同登录；加过滤器还有配置同20；

（1）UserDao代码：登录UserDao代码下加**public** **int** inserts(String userName,String password){

Connection conn = **null**;

ArrayList LoginForm = **new** ArrayList();

//获取连接

**try** {

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

String url="jdbc:odbc:Employ";

**try** {

conn=DriverManager.*getConnection*(url,"sasa", "123456");

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

//运行sql语句

String sql="insert into [Employ].[dbo].[Users] values ('"+userName+"','"+password+"')";

Statement stat = **null**;

**try** {

stat = conn.createStatement();

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**int** rs = 0;

**try** {

rs = stat.executeUpdate(sql);

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**return** rs;

}

**public** **static** **void** main(String args[]) **throws** Exception{

UserDao dao=**new** UserDao();

ResultSet rs=dao.quaryAllUser();

**while**(rs.next()){

System.*out*.println(rs.getString("userName"));

}

}

1. zhuceservlet.Java代码：在Dopost里面写

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

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

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

UserDao dao=**new** UserDao();

**int** i=dao.inserts(userName, password);

**if**(i==1){

response.sendRedirect("http://localhost:8080/prj17/welcome.jsp");

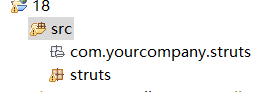
}

**else** response.sendRedirect("http://localhost:8080/prj17/zhuce.jsp");

}

**18.使用struts1.x技术，实现登录功能（不连接数据库）**

先按书186-192页配置：建一个web project，点上面Myeclipse —第一个—add struts Capailities—点finish把名字改为struts即可；工程名和包名弄一样的。



Web.xml不用配置，lib下的第二个struts-config.xml代码如下：

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

<!DOCTYPE struts-config PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 1.2//EN" "http://struts.apache.org/dtds/struts-config\_1\_2.dtd">

<struts-config>

<data-sources />

<form-beans >

<form-bean name=*"LoginForm"* type=*"struts（包*

*名）.LoginForm"*></form-bean>

</form-beans>

<global-exceptions />

<global-forwards />

<action-mappings >

<action path=*"/Login"* name=*"LoginForm"* type=*"struts(包名).LoginAction"*></action>

</action-mappings>

<message-resources parameter=*"18.struts.ApplicationResources"* />

</struts-config>

1. login.jsp代码：

<%@ page language=*"java"* contentType=*"text/html; charset=gb2312"*%>

<head>

<title>登陆留言板</title>

</head>

<body bgcolor=*"#33FF66"*>

<h3>

欢迎登陆留言板

</h3>

<form action=*"./Login.do"* method=*"post"*>

<br>

用户：

<input name=*"userName"* type=*"text"*>

<br>

<br>

密码：

<input name=*"password"* type=*"password"*>

<br>

<br>

<input type=*"submit"* value=*"登陆"*>

<input type=*"reset"* value=*"重置"*>

</form>

</body>

1. Welcome.jsp代码：

<head>

<title>登陆留言板</title>

</head>

<body bgcolor=*"#33FF66"*>

<h3>

欢迎登陆留言板

</h3>

<form action=*"./Login.do"* method=*"post"*>

<br>

用户：

<input name=*"userName"* type=*"text"*>

<br>

<br>

密码：

<input name=*"password"* type=*"password"*>

<br>

<br>

<input type=*"submit"* value=*"登陆"*>

<input type=*"reset"* value=*"重置"*>

</form>

</body>

1. Src建包然后里面建两个class

LoginForm.Java：

**package** struts;

**import** org.apache.struts.action.ActionForm;

**public** **class** LoginForm **extends** ActionForm {

**private** String userName;

**private** String password;

**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;

}

}

1. LoginAction.Java：

package struts;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

public class LoginAction extends Action {

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response)

throws Exception {

LoginForm loginform=(LoginForm)form;

String userName=loginform.getUserName();

String password=loginform.getPassword();

if(userName.equals("admin") && password.equals("123")) {

return new ActionForward("/welcome.jsp");

}

return new ActionForward("/login.jsp");

}

}

1. **P196（1）编写学生资料模糊界面，输入学生姓名的模糊资料，在另一个界面中显示所有的学生信息。要求使用struts框架来实现。**

**书P124**

**20.使用struts1.x技术，实现登录功能（连接数据库）**

一.登录功能：

配置同18，建一个包dao建一个UserDao的class；在struts下建LoginAction和LoginForm；login.jsp和welcome。Jsp代码同18

（1）LoginAction代码：package com.yourcompany.struts;

import java.util.ArrayList;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import dao.UserDao;

public class LoginAction extends Action {

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response)

throws Exception {

LoginForm loginForm = (LoginForm) form;

String userName = loginForm.getUserName();

String password = loginForm.getPassword();

UserDao dao = new UserDao();

ArrayList list1 = dao.quaryAllUser();

//request.setCharacterEncoding("UTF-8");

for (int i = 0; i < list1.size(); i++)

{

loginForm = (LoginForm) list1.get(i);

if((loginForm.getUserName()).equals(userName) && (loginForm.getPassword()).equals(password)){

return new ActionForward("/welcome.jsp");

}}

return new ActionForward("/login.jsp");

}

}

1. loginform代码同18
2. 建一个包dao里面的UserDao代码：

**package** dao;

**import** java.sql.\*;

**import** java.util.ArrayList;

**import** java.util.List;

**import** com.yourcompany.struts.LoginForm;

**public** **class** UserDao {

**public** ArrayList quaryAllUser()**throws** Exception{

Connection conn = **null**;

ArrayList LoginForm = **new** ArrayList();

**try**{

//获取连接

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

String url="jdbc:odbc:Employ";

conn=DriverManager.*getConnection*(url,"sasa", "123456");

//运行sql语句

String sql="SELECT TOP 1000 [userName],[password] FROM [Employ].[dbo].[Users]";

Statement stat=conn.createStatement();

ResultSet rs=stat.executeQuery(sql);

**while**(rs.next()){

//实例化VO

LoginForm loginform=**new** LoginForm();

loginform.setUserName(rs.getString("userName"));

loginform.setPassword(rs.getString("password"));

LoginForm.add(loginform);

}

rs.close();

stat.close();

}**catch**(SQLException e){

e.printStackTrace();

}**finally**{

**try**{//关闭连接

**if**(conn!=**null**){

conn.close();

conn=**null**;

}

}**catch**(Exception ex){

}

}

**return** LoginForm;

}

}

1. 注册功能：

点dao右键建class名为EncodingFilter，然后点add javax.servlet.Filter就行，在配置web.xml

在struts-config的空白处点右键new—jsp，action and Form然后点zhuce

1. zhuce1Action.java代码：

/\*

\* Generated by MyEclipse Struts

\* Template path: templates/java/JavaClass.vtl

\*/

package com.yourcompany.struts.action;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

import com.yourcompany.struts.form.Zhuce1Form;

import dao.UserDao;

/\*\*

\* MyEclipse Struts

\* Creation date: 05-08-2017

\*

\* XDoclet definition:

\* @struts.action path="/zhuce1" name="zhuce1Form" input="/form/zhuce1.jsp" scope="request" validate="true"

\* @struts.action-forward name="success" path="/welcome.jsp"

\* @struts.action-forward name="fail" path="/login.jsp"

\*/

public class Zhuce1Action extends Action {

/\*

\* Generated Methods

\*/

/\*\*

\* Method execute

\* @param mapping

\* @param form

\* @param request

\* @param response

\* @return ActionForward

\*/

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) {

Zhuce1Form zhuce1Form = (Zhuce1Form) form;// TODO Auto-generated method stub

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

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

UserDao dao=new UserDao();

int i=dao.inserts(userName, password);

if(i==1){

return mapping.findForward("success");

}

else return mapping.findForward("fail");

}

}

（2）UserDao：上面同登录，下面加

**public** **int** inserts(String userName,String password){

Connection conn = **null**;

ArrayList LoginForm = **new** ArrayList();

//获取连接

**try** {

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

String url="jdbc:odbc:Employ";

**try** {

conn=DriverManager.*getConnection*(url,"sasa", "123456");

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

String sql="insert into [Employ].[dbo].[Users] values ('"+userName+"','"+password+"')";

Statement stat = **null**;

**try** {

stat = conn.createStatement();

} **catch** (SQLException e) {

e.printStackTrace();

}

**int** rs = 0;

**try** {

rs = stat.executeUpdate(sql);

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**return** rs;

}

login.jsp和welcome.jsp同登录

（3）EncodingFilter代码：package dao;

import java.io.IOException;

import javax.servlet.Filter;

import javax.servlet.FilterChain;

import javax.servlet.FilterConfig;

import javax.servlet.ServletException;

import javax.servlet.ServletRequest;

import javax.servlet.ServletResponse;

public class EncodingFilter implements Filter {

public void destroy() {

// TODO Auto-generated method stub

}

public void doFilter(ServletRequest request, ServletResponse response,

FilterChain chain) throws IOException, ServletException {

// TODO Auto-generated method stub

request.setCharacterEncoding("gb2312");

chain.doFilter(request, response);

}

public void init(FilterConfig arg0) throws ServletException {

// TODO Auto-generated method stub

}

}

（4）login.jsp代码：

<body>

欢迎来到登录页面！

<form action=*"/prj17/servlet/loginservlet"* method=*"post"*>

请输入你的账号:<input name=*"username"*type=*"text"*><BR>

请输入你的密码:<input name=*"password"*type=*"password"*><BR>

<input type=*"submit"* value=*"登录"*>

</form>

<form action=*"zhuce.jsp"* method=*"post"*>

<input type=*"submit"* value=*"登录"*>

</body>

1. web.xml代码：...

<filter>

<filter-name>EncodingFilter</filter-name>

<filter-class>dao.EncodingFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>EncodingFilter</filter-name>

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

</filter-mapping>

.....

1. **（1）编写一个登陆界面，有账号和密码两个表单元素，控制用户输入，是用户输入的账号和密码必须不为空；账号必须在5~8之间，密码必须在6~10位之间；账号必须都是数字题。要求所有的提示信息和错误信息都从资源文件中得到并且是中文。**

**见书P209**

**24.使用struts2技术，实现登录功能（不连接数据库）**

步骤：按照书P223-228

在lib下引进五个包，建一个web project建一个名为struts2的包建一个LoginAction，Java的class，写完loginaction代码后点struts右键建struts.xml文件配置，再配置web。Xml文件

（1）struts.xml代码：<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE struts PUBLIC

"-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"

"http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>

<package name=*"prj66"* extends=*"struts-default"*>

<action name=*"login"* class=*"prj66.LoginAction"*>

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

<result name=*"fail"*>/loginFail.jsp</result>

</action>

</package>

</struts>

（2）web.xml代码：<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app version=*"2.4"*

xmlns=*"http://java.sun.com/xml/ns/j2ee"*

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

xsi:schemaLocation=*"http://java.sun.com/xml/ns/j2ee*

*http://java.sun.com/xml/ns/j2ee/web-app\_2\_4.xsd"*>

<display-name>prj66</display-name>

<filter>

<filter-name>struts2</filter-name>

<filter-class>org.apache.struts2.dispatcher.FilterDispatcher</filter-class>

</filter>

<filter-mapping>

<filter-name>struts2</filter-name>

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

</filter-mapping>

</web-app>

（3）LoginAction。Java代码：**package** prj66;

**public** **class** LoginAction {

**private** String account;

**private** String password;

**public** String getAccount() {

**return** account;

}

**public** **void** setAccount(String account) {

**this**.account = account;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

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

**if**(account.equals("admin")&&password.equals("123")){

**return** "success";

}

**return** "fail";

}

}

（4）login.jsp代码：

<body>

<form action=*"./login.action"*method=*"post"*>

请您输入账号:<input name=*"account"*type=*"text"*><BR>

请您输入密码:<input name=*"password"*type=*"password"*>

<input type=*"submit"* value=*"登录"*>

</form>

</body>

（5）loginSucess.jsp代码：

<body>

登陆成功 <br>

</body>

（6）loginFail.jsp代码：登录成功改为登陆失败

**25.使用struts2技术，实现登录功能（连接数据库）**

一.登录功能：

步骤：配置同上，三个jsp页面同上，两个xml文件代码同上；

（1）LoginAction.jsp代码上面同上一样，最后一个函数代码：

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

UserDao dao=**new** UserDao();

ArrayList list=dao.quaryAllUser();

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

{

LoginAction loginaction = (LoginAction) list.get(i);

System.*out*.println(loginaction.getAccount());

**if**(account.equals(loginaction.getAccount()) && password.equals(loginaction.getPassword())){

**return** "success";

}

}

**return** "fail";

}

（2）UserDao代码：

**package** dao;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**import** java.util.ArrayList;

**import** struts2.LoginAction;

**public** **class** UserDao {

**public** ArrayList quaryAllUser()**throws** Exception{

Connection conn = **null**;

ArrayList LoginAction = **new** ArrayList();

**try**{

//获取连接

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

String url="jdbc:odbc:Employ";

conn=DriverManager.*getConnection*(url,"sasa", "123456");

//运行sql语句

String sql="SELECT TOP 1000 [account],[password] FROM [Employ].[dbo].[User1]";

Statement stat=conn.createStatement();

ResultSet rs=stat.executeQuery(sql);

**while**(rs.next()){

//实例化VO

LoginAction loginaction=**new** LoginAction();

loginaction.setAccount(rs.getString("account"));

loginaction.setPassword(rs.getString("password"));

LoginAction.add(loginaction);

}

rs.close();

stat.close();

}**catch**(SQLException e){

e.printStackTrace();

}**finally**{

**try**{//关闭连接

**if**(conn!=**null**){

conn.close();

conn=**null**;

}

}**catch**(Exception ex){

}

}

**return** LoginAction;

}}

二． 注册功能：

两个xml页面：web.xml同登录；三个jsp页面：zhuce，welcome，fail页面

（1）struts.Xml代码为<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE struts PUBLIC

"-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"

"http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>

<package name=*"struts2"* extends=*"struts-default"*>

<action name=*"login"* class=*"struts2.LoginAction"*>

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

<result name=*"fail"*>/loginFail.jsp</result>

</action>

<action name=*"zhuce"* class=*"sturts2.ZhuceAction"*>

<result name=*"zhucesuccess"*>/welcome.jsp</result>

<result name=*"zhucefail"*>/fail.jsp</result>

</action>

</package>

</struts>

1. UserDao代码：在登陆下加

**public** **int** inserts(String userName,String password){

Connection conn = **null**;

ArrayList LoginForm = **new** ArrayList();

//获取连接

**try** {

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

String url="jdbc:odbc:Employ";

**try** {

conn=DriverManager.*getConnection*(url,"sasa", "123456");

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

//运行sql语句

String sql="insert into [Employ].[dbo].[User1] values ('"+userName+"','"+password+"')";

Statement stat = **null**;

**try** {

stat = conn.createStatement();

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**int** rs = 0;

**try** {

rs = stat.executeUpdate(sql);

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**return** rs;

}

1. 三个页面代码同登录

（4）ZhuceAction.Java代码：package struts2;

import java.util.ArrayList;

import dao.UserDao;

public class ZhuceAction {

public class LoginAction {

private String account;

private String password;

public String getAccount() {

return account;

}

public void setAccount(String account) {

this.account = account;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public String registered(){

UserDao dao=new UserDao();

int i=dao.inserts(account, password);

if(i==1){

return "zhucesuccess";

}

else return "zhucefail";

}

}

}

（5）login.jsp代码：

<body>

<form action=*"./login.action"*method=*"post"*>

请您输入账号:<input name=*"account"*type=*"text"*><BR>

请您输入密码:<input name=*"password"*type=*"password"*>

<input type=*"submit"* value=*"登录"*>

</form>

<form action=*"zhuce.jsp"* method=*"post"*>

<input type=*"submit"* value=*"注册"*>

</form>

</body>

**27.在struts2里实现国际化，完成登录页面的国际化（英文和中文）**

步骤：加五个包，建action和local两个包;action里有LoginAction；local里有两个包；建struts.xml；配置web.xml；建两个jsp

（1）Struts.xml代码：<?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="struts2" namespace="/" extends="struts-default">

<action name="login" class="action.loginAction">

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

<result name="fail">/login.jsp</result>

</action>

</package>

</struts>

(2)login.jsp代码：<%@ 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>

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

</head>

<body>

<s:i18n name="local.message">

<s:form action="login.action" method="POST" namespace="/">

<s:textfield name="username" label="%{getText('login.username')}"/>

<s:password name="password" label="%{getText('login.password')}"/>

<s:submit value="提交"/>

</s:form>

</s:i18n>

</body>

</html>

（3）welcome.jsp代码：<%@ page language="java" import="java.util.\*" pageEncoding="UTF-8"%>

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

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

<html>

<head>

<title>My JSP 'welcome.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>

<s:i18n name="local.message">

<s:label value="%{getText('login.welcome')}"></s:label>

</s:i18n>

<body>

</body>

</html>

（4）web.xml代码：<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5"

xmlns="http://java.sun.com/xml/ns/javaee"

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

xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<display-name></display-name>

<welcome-file-list>

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

</welcome-file-list>

<filter>

<filter-name>struts2</filter-name>

<filter-class>

org.apache.struts2.dispatcher.ng.filter.StrutsPrepareAndExecuteFilter

</filter-class>

</filter>

<filter-mapping>

<filter-name>struts2</filter-name>

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

</filter-mapping>

</web-app>

（5）LoginAction.Java代码：package action;

import com.opensymphony.xwork2.ActionSupport;

import org.apache.struts2.interceptor.\*;

public class loginAction extends ActionSupport {

private String username;

private String password;

public String execute(){

if("admin".equals(username) && "123".equals(password))

{

return "success";

}

return "fail";

}

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;

}

}

1. **在struts2里，完成登录页面，使用validate对输入数据进行校验，用户名必填，密码必填且长度大于6。**

步骤：建Action包建login的class，建struts.xml；两个jsp页面；配置web.xml

（1）login.Java代码：package Action;

import com.opensymphony.xwork2.ActionSupport;

public class login extends ActionSupport {

private String username;

private String password;

public String execute(){

System.out.println(SUCCESS);

return SUCCESS;

}

public void validate(){

boolean ifFail = false;

if(username == null || "".equals(username)){

addFieldError("username", "用户名不能为空！");

ifFail = true;

}

if(password == null || "".equals(password)){

addFieldError("password", "密码不能为空！");

ifFail = true;

}

else if(password.length() <3){

addFieldError("password", "密码少于3位！");

ifFail = true;

}

if(ifFail){

this.addActionMessage("提交失败");

return ;

}

else{

this.addActionMessage("提交成功");

return ;

}

}

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;

}

}

1. struts.xml代码：<?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="login" extends="struts-default">

<action name="validate" class="Action.login">

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

<result name="input">/login.jsp</result>

</action>

</package>

</struts>

（3）web.xml代码：<display-name/>

<welcome-file-list>

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

</welcome-file-list>

-<filter>

<filter-name>struts2</filter-name>

<filter-class>org.apache.struts2.dispatcher.ng.filter.StrutsPrepareAndExecuteFier </filter-class>

</filter>

-<filter-mapping>

<filter-name>struts2</filter-name>

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

</filter-mapping>

</web-app>

（4）login.jsp代码：<%@ 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>

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

</head>

<body>

<s:actionerror />

<s:actionmessage />

<s:form action="validate.action">

<s:textfield name="username" label="用户名" /><br>

<s:password name="password" label="密码"/>

<s:submit />

</s:form>

</body>

</html>

（5）welcome.jsp代码：<%@ page language="java" import="java.util.\*" pageEncoding="UTF-8"%>

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

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

<html>

<head>

<title>My JSP 'welcome.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>

欢迎登录！

</body>

</html>

1. **在struts2里，完成登录页面，使用框架进行数据校验，用户名必填，密码必填且长度大于6。**

步骤：建Action包里面建class名为LoginAction；点击src右键建struts.xml；点击Action右键建loginAction-validation.xml；建两个jsp页面；配置web.xml

（1）struts.xml代码：<?xml version="1.0" encoding="UTF-8"?>

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

-<struts>

-<package extends="struts-default" name="struts2">

-<action name="login" class="Action.loginAction">

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

<result name="input">/login.jsp</result>

</action>

</package>

</struts>

1. login.Java代码：package Action;

import com.opensymphony.xwork2.ActionSupport;

public class loginAction extends ActionSupport {

private String username;

private String password;

public String execute(){

return SUCCESS;

}

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;

}

}

（3）loginAction-validation.xml代码：<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE validators PUBLIC "-//OpenSymphony Group//XWork Validator 1.0//EN" "http://www.opensymphony.com/xwork/xwork-validator-1.0.2.dtd" >

<validators>

<field name="username">

<field-validator type="requiredstring">

<message>请输入用户名！</message>

</field-validator>

</field>

<field name="password">

<field-validator type="requiredstring">

<message>请输入密码！</message>

</field-validator>

</field>

<field name="password.length()">

<field-validator type="int">

<param name="min">7</param>

<param name="max">20</param>

<message>密码长度必须大于6</message>

</field-validator>

</field>

</validators>

（4）login.jsp代码：<%@ 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>

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

</head>

<body>

<s:form action="login">

<s:textfield name="username" label="用户名" />

<s:password name="password" label="密码" />

<s:submit value="提交"/>

</s:form>

</body>

</html>

（5）welcome.jsp代码：<%@ page language="java" import="java.util.\*" pageEncoding="UTF-8"%>

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

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

<html>

<head>

<title>My JSP 'welcome.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>

欢迎登录！

</body>

</html>

（6）web.xml代码：<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5"

xmlns="http://java.sun.com/xml/ns/javaee"

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

xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<display-name></display-name>

<welcome-file-list>

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

</welcome-file-list>

<filter>

<filter-name>struts2</filter-name>

<filter-class>

org.apache.struts2.dispatcher.ng.filter.StrutsPrepareAndExecuteFilter

</filter-class>

</filter>

<filter-mapping>

<filter-name>struts2</filter-name>

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

</filter-mapping></web-app>