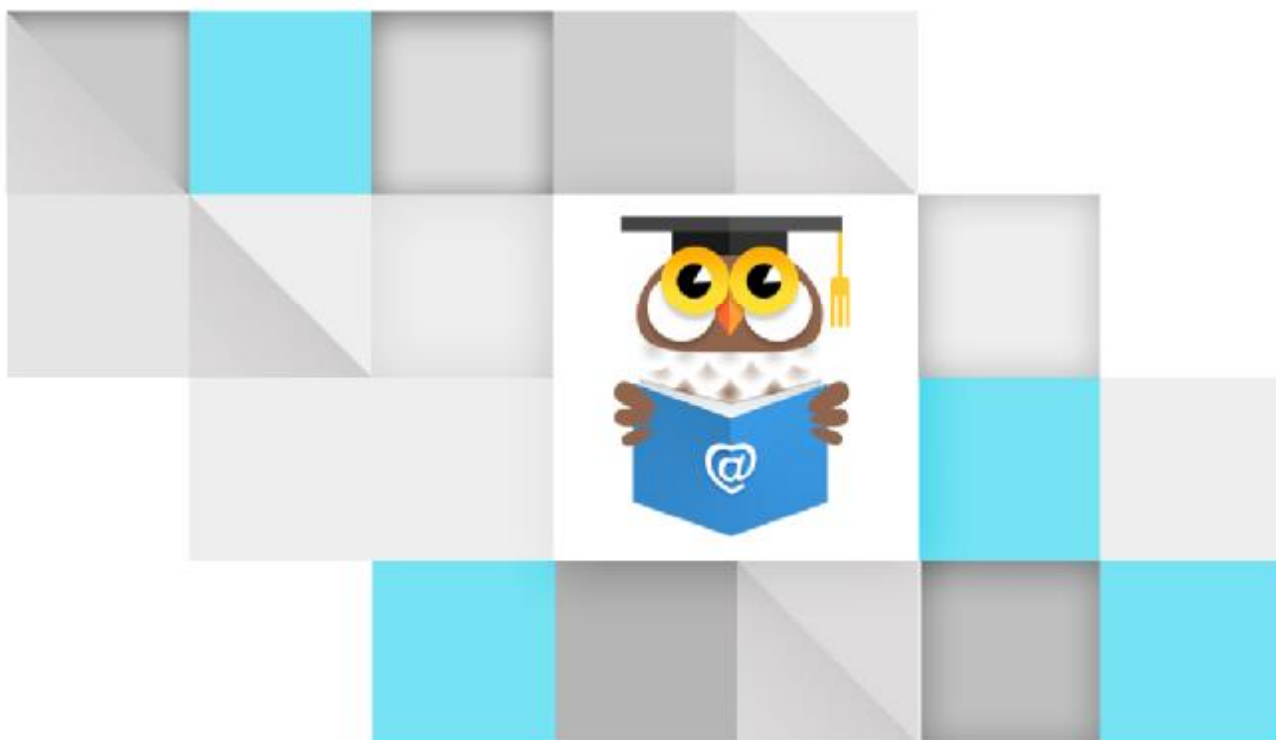


去哪玩旅游网（动态 WEB）

教学指导手册

PRJ-WTP-JEE-007 错误登录提示



Campus Solution Group

目 录

一、场景说明	1
1、完成效果	1
2、业务描述	1
3、实现思路	1
4、核心组件	2
二、实训技能	3
1、重点演练	3
2、相关技能	3
3、相关知识点	3
4、前置条件	4
5、搭建环境	4
三、场景任务	5
任务 1、login.html à login.jsp	5
任务 2、验证真实的用户名和密码	8
任务 3、页面动态显示错误信息	11

一、场景说明

1、完成效果



图 1-1-1

2、业务描述

PRJ-WTP-JEE-003场景实现了，固定用户名与密码的登录业务，本场景在此基础上，实现《去哪玩》旅游网的真实登录业务。

2-1. 从数据库获取真实用户数据，代替原本固定的用户数据进行登录判断（无需实现）。

2-2. 在页面上动态显示登录失败的错误提示信息。

3、实现思路

3-1. 本场景建议将“真实登录业务”分为三个任务依次实现：

3-1.1. 任务1. 将login.html转成JSP文件，为显示登录业务中的错误信息做准备。

3-1.2. 任务2. 修改原固定用户与密码的判断逻辑，实现真实登录业务流程。

3-1.3. 任务3. 在任务2的基础上，确保登录数据无效时，登录页面呈现错误信息。

4、核心组件

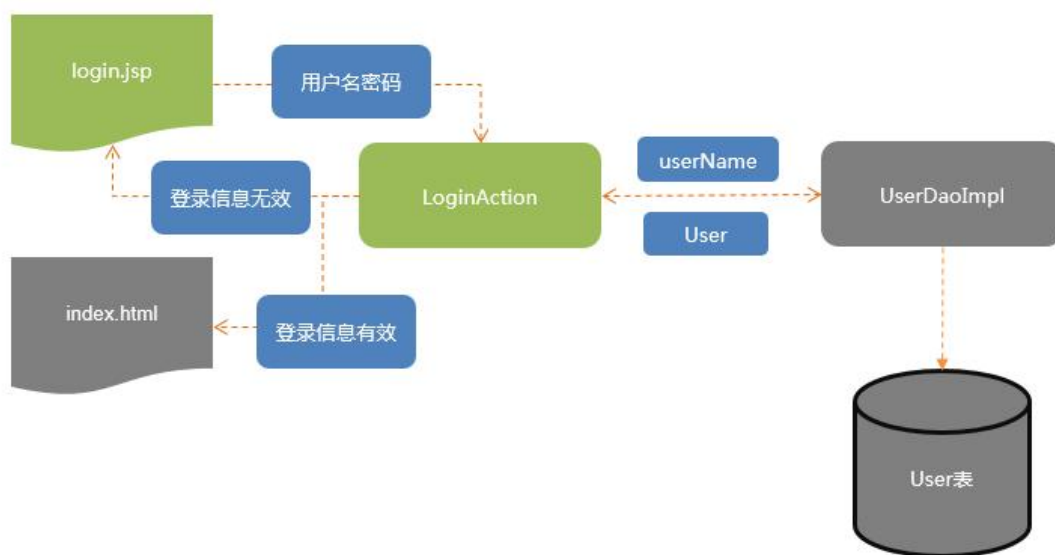


图 1-4-1

4-1. login.jsp (需实现) :

4-1.1. 提供用户登录表单，在登录失败时负责动态显示错误提示。

4-2. index.html (已提供) :

4-2.1. 网站首页，登录成功时跳转到该页面。

4-3. LoginAction (需实现) :

4-3.1. 负责处理用户登录请求，并根据校验结果做出不同跳转响应的Servlet。

4-4. UserDaoImpl (已提供) :

4-4.1. 《去哪玩旅游网》数据库 – User表的数据访问类。

4-4.2. 用于访问和读取数据库中的User表数据。

4-5. UserName :

4-5.1. 用户输入的用户名：

4-5.2.本场景通过该属性查询用户数据，UserName一般是手机号码。

4-6. User：

4-6.1. UserDaoImpl的getUserByCondition函数查询得到用户数据。

4-6.2. 查询获得的数据会被封装入该对象。

二、实训技能

1、重点演练

1-1. HTML转JSP。

1-2. 写入与读取Web作用域（Request）。

1-3. JSP指令标签的应用。

1-4. JSP服务端脚本标记的应用。

1-5. JSP服务端输出标记的应用。

1-6. JSP隐式对象（request）的应用。

2、相关技能

I JSP 技术应用

3、相关知识点

I JSP 指令元素脚本

I JSP 代码脚本

I JSP 输出脚本

I Servlet 访问 JSP

I 访问安全性

I 服务端跳转

4、前置条件

3-1. 前置场景：PRJ-WTP-JEE-005 – 校验验证码

3-2. 已学技能：

3-2.1. Java开发工具（Eclipse）。

3-2.2. Tomcat服务器。

3-2.3. MySQL数据库。

3-2.4. Servlet应用技术。

3-2.5. JSP应用技术。

5、搭建环境



三、场景任务

任务 1、login.html à login.jsp

1. 任务说明：

1-1. 完成效果：



图 3-1-1

1-2. 任务目标：

1-2.1. 将login.html转成JSP文件，为显示登录业务中的错误信息做准备。

1-3. 任务要求：

1-3.1. 为保证页面数据的动态呈现，将登录页login.html转换成JSP文件。

1) 限制1. 转换后的文件名，固定为：login.jsp

2) 限制2. login.jsp文件需存储在：站点根目录下（与html同级目录）。

1-3.2. 为JSP设置必要的头参数，确保JSP文件、响应数据不能出现乱码。

2. 实现思路：

2-1. 实现登录页HTMLàJSP：

2-1.1. 复制login.html文件，在同级目录粘贴一个副本，命名为login.jsp。

2-2. 设置JSP文件的编码集：

2-2.1. 将login.jsp文件的编码集设置为：UTF-8。

2-3. 为JSP设置必要的头参数（<%@page %>）：

2-3.1. 设置language属性，值为：java。

2-3.2. 设置contentType属性，值为：text/html；charset=UTF-8

2-3.3. 设置pageEncoding属性，值为：UTF-8

2-4. 任务开发流程见下图：

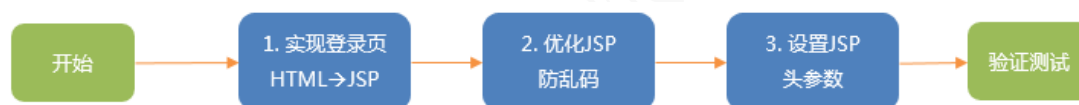


图 3-1-2

3. 推荐步骤：

2-1. 实现登录页HTMLàJSP

+ 提示：

1) 重命名文件的方法：在文件上单击右键，选择Refactor à Rename

2-2. 优化JSP文件，防止中文乱码。

+ 提示：

1) 右键文件login.jsp选择【Properties】，选择左侧的【Resource】选项。

2) 右侧资源对话框中，定位【Text file encoding】中选择Other单选框。

3) 将字符集下拉框设置为【UTF-8】编码集。

2-3. 设置JSP头参数

4. 验证与测试：

4-1. 运行项目工程。

4-2. 并在浏览器地址栏输出 http://localhost/PRJ_WTP_JEE_007/ 进行访问。

任务 2、验证真实的用户名和密码

1. 任务说明：

1-1. 完成效果：

用户名或密码错误
用户名和密码输入正确

图 3-2-1

1-2. 任务目标：

1-2.1. 修改原固定用户与密码的判断逻辑，实现真实登录业务流程。

1-3. 任务要求：

1-3.1. 访问《去哪玩》数据库：qunawan中的用户表User。

1-3.2. 根据输入的用户名（手机号），判断User表内是否存在匹配的用户信息。

+ 业务说明：

- 1) 本场景不要编写数据库的访问代码。
- 2) 获取匹配记录，调用：`UserDaoImpl`类的`getUserByCondition`函数。
- 3) 有匹配记录，`getUserByCondition`返回`User`对象，该对象保存着完整的用户信息。
- 4) 无匹配记录，`getUserByCondition`返回`null`。

1-3.3. 如果存在匹配记录则判断，`User`对象中的密码与用户输入的密码是否一致：

+ 业务说明：

- 1) `User`对象中的密码可通过`User`类的`getPassword`函数获取。
- 2) 判断密码是否一致，需要对用户输入的密码进行MD5加密。

说明：为确保密码的安全性，`User`表中的密码经过了MD5加密。

说明：当前任务需对用户输入的密码进行MD5加密后，才可判断两者是否一致。

3) MD5加密, 可调用: `campsg.qunawan.utils.Utils`类的静态函数`toMD5`。

1-3.4. 如果密码一致, 则向控制台输出: “用户名和密码输入正确”。

1-3.5. 如无匹配记录, 或者密码不一致, 则向控制台输出: “用户名或密码错误”。

2. 实现思路:

2-1. 在LoginAction的doGet方法中创建UserDaoImpl类的实例。

2-2. 调用getUserByCondition函数, 将从请求中获取的用户名(手机号)传入。

2-3. 判断1: getUserByCondition函数返回结果是否为null。

2-4. 判断2: 对用户输入的密码进行MD5加密, 判断该密码与User表中的密码是否一致。

2-5. 判断1和判断2均满足, 则向控制台输出: “用户名和密码输入正确”。

2-6. 判断1和判断2任意一项不满足, 则向控制台输出: “用户名或密码错误”。

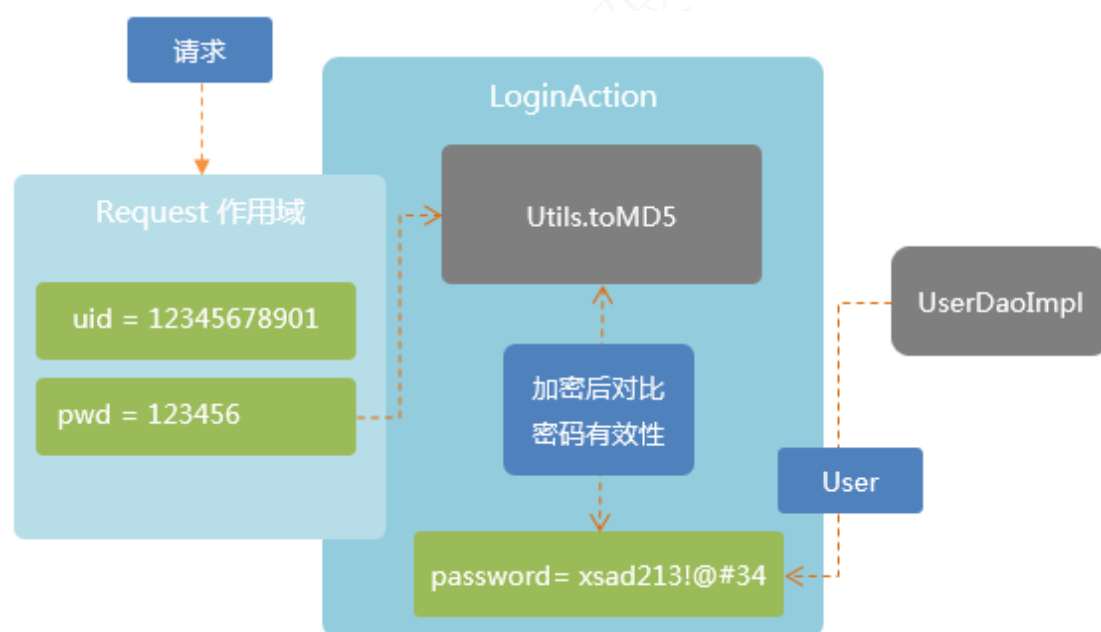


图 3-2-2

2-7. 任务开发流程见下图:

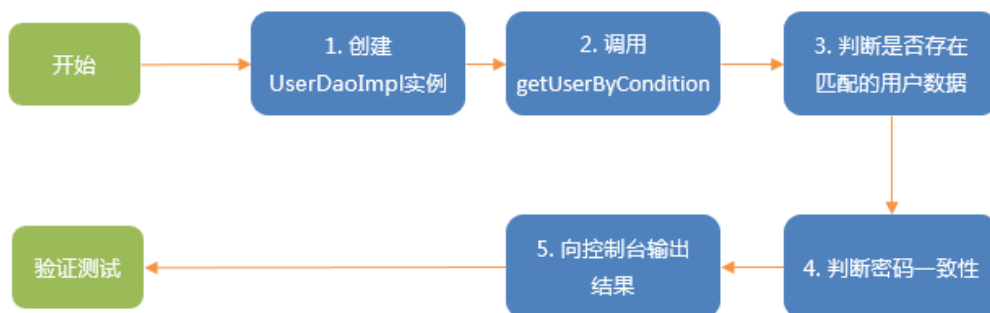


图 3-2-3

3. 推荐步骤：

3-1. 按实现思路依次完成。

+ 业务说明：

1) 密码为何需要做MD5加密：

实际生产环境中，密码是非常重要的字段，一般都不会做明码保存，MD5属于一种不可逆加密算法，安全性级别较高，当数据表遭非法入侵时，入侵者即便获取了数据表访问权限，也无法获取真实的密码。

4. 验证与测试：

4-1. 运行项目工程。

4-2. 错误登录验证：

4-2.1. 务必确保输入的验证码错误。

4-2.2. 输入错误的用户名、密码【用户名：12345678901 密码：111】。

4-2.3. 浏览器无显示，控制台打印：用户名或密码错误。

4-3. 正确登录验证：

4-3.1. 务必确保输入的验证码正确。

4-3.2. 输入正确的用户名、密码【用户名：18701721202 密码：123456】

4-3.3. 浏览器无显示，控制台打印：用户名和密码输入正确。

任务 3、页面动态显示错误信息

1. 任务说明：

1-1. 完成效果：

1-1.1. 用户输入的用户名或密码不正确（数据表中无可匹配的用户名和密码）：



图 3-3-1

1-1.2. 用户输入的用户名或密码正确（数据表中有可匹配的用户名和密码）：

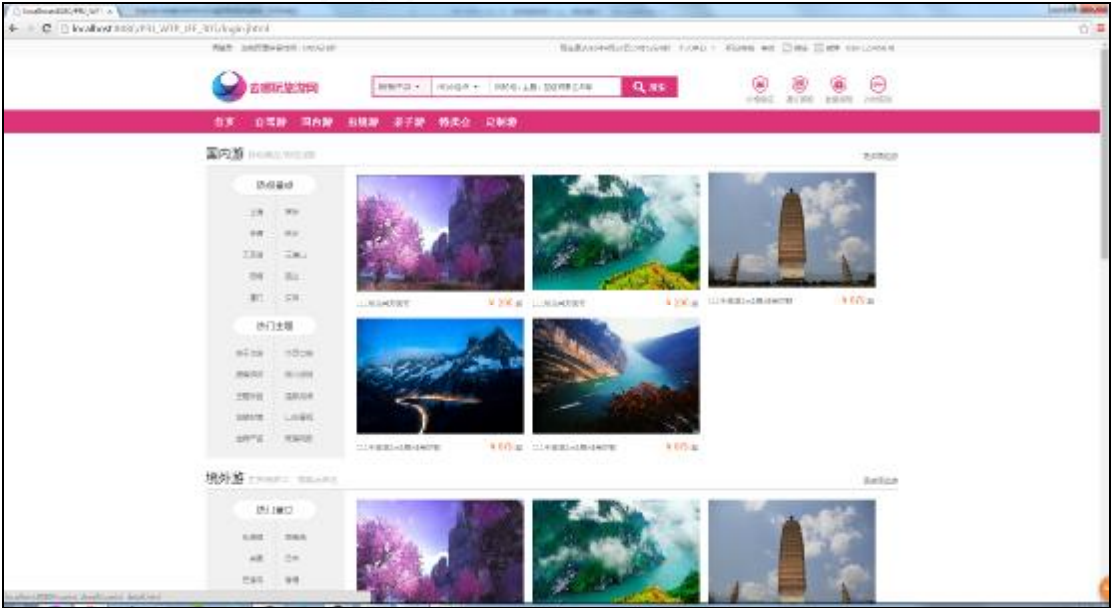


图 3-3-2

1-2. 任务目标：

1-2.1. 在任务2的基础上，确保登录数据无效时，登录页面呈现错误信息。

1-3. 任务要求：

1-3.1. 当用户输入的用户名或密码不正确时，将错误信息保存于Request作用域中：

1) 限制1. 错误信息：用户名或密码错误

2) 限制2. 保存入Request的key，固定使用：**error_msg**。

1-3.2. 根据登录信息验证的有效性，实现页面跳转：

1) 登录信息有效：跳转到**index.html**，使用常量：OK_URL

2) 登录信息无效：跳转回**login.jsp**，使用常量：ERROR_URL

1-3.3. 当登录信息无效时，登录界面上显示：“**用户名或密码错误**”

2. 实现思路：

2-1. 在LoginAction的doGet方法，找到**任务2**实现的登录信息验证代码。

2-2. 登录信息无效时的处理方法：

2-2.1. 将“用户名或密码错误”的信息保存入Request。

2-2.2. 存入Request作用域时，key = **error_msg**。

2-2.3. 为确保登录页可以显示错误信息，使用**服务端跳转**语句，跳转到**login.jsp**（使用常量：ERROR_URL）。

2-3. 登录信息有效时的处理方法：

2-3.1. 使用**客户端重定向**语句，跳转到index.html（使用常量：OK_URL）。

2-4. 在login.jsp页面中编写错误信息显示标签：

2-4.1. 利用服务端脚本标签（<% %>）从Request作用域中获取错误信息。

2-4.2. 在登录页中显示错误信息：

1) 找到id=contents所在的div标签对。

2) 通过<%= %>标记显示错误信息。

2-4.3. 在获取错误信息后，需做判null处理：

1) 如果获取的错误信息为null，则直接在登录页面中显示空数据（ "" ）；

2) 如果不为null，则直接将错误信息显示在页面上。

+ 业务说明：

1) 当login.jsp通过URL直接访问，或者第一次访问login.jsp的时候，错误信息始终为null。

2) 为防止页面直接显示null字符，需要对错误信息做判null处理。

2-5. 任务开发流程见下图：



图 3-3-3

3. 推荐步骤：

3-1. 按实现思路依次完成。

3-2. 登录信息无效时，通过服务端跳转技术返回登录页，URL为成员变量：【ERROR_URL】。

+ 提示：

1) 服务器跳转技术：

```
request.getRequestDispatcher("跳转页面").forward(request, response);
```

2) 服务器跳转技术的特点：

2-1. 客户端浏览器地址栏不会发生任何变化（不显示跳转页面的连接）。

2-2. 保存在请求 (Request) 作用域中的数据不会丢失。

3-3. 登录信息有效时,通过**客户端重定向技术**跳转到首页,URL为成员变量:【OK_URL】。

+提示:

1) 客户端重定向技术:

```
response.sendRedirect("跳转页面");
```

3-4. 在JSP中显示错误信息:

3-4.1. 获取错误提示信息。

3-4.2. 错误信息判null处理。

3-4.3. 在id=contents所在的div标签对中,显示错误提示信息:

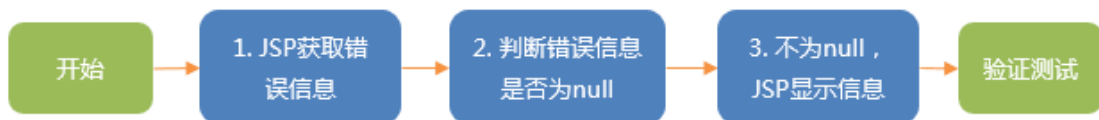


图 3-3-4

4. 验证与测试:

4-1. 运行项目工程。

4-2. 错误信息判null处理测试:

4-2.1. 输入地址: http://localhost/PRJ_WTP_JEE_007/ 访问登录页。

4-2.2. 没有看到任何错误提示。

4-2.3. 浏览器地址栏的URL应该以: login.jhtml结尾。

4-3. 错误登录验证:

4-3.1. 务必确保输入的验证码正确。

4-3.2. 输入错误的用户名、密码【用户名: 12345678901 密码: 111】。

4-3.3. 应可见【任务完成效果图1】。

4-4. 正确登录验证:

4-4.1. 务必确保输入的验证码正确。

4-4.2. 输入正确的用户名、密码【用户名：18701721202 密码：123456】。

4-4.3. 应可见【任务完成效果图2】。

4-4.4. 浏览器地址栏的URL应该以：index.html结尾。

作者：Roger.Huang