# 实验九 补充练习

**一、实验报告要求**

1、实验报告提交word文档，命名为“学号+姓名+实验9补充练习”。

2、本次实验共2题。本题应提供**完整代码**、以及**运行截图**。

（1）第1题的运行截图，应至少包含无敏感词、有敏感词的两种情况。其中，**无敏感词的评论内容**显示为**本人的学号和姓名**。

（2）第2题的运行截图，应至少包含使用本人的学号和姓名进行登录的**登录页面**、以及登录成功后的**欢迎页面**。

3、附上调试程序的过程中，出现的问题、以及解决方法。【可选】

**二、实验内容**

1、实现敏感词过滤。若输入的评论内容中没有敏感词，则返回标签“good guy”以及原评论内容；若评论内容中有敏感词，则返回标签“bad guy”以及使用“\*\*”替换后的评论内容。

|  |  |
| --- | --- |
|  |  |
| 无敏感词 | 有敏感词 |

【实验步骤】

（1）新建一个名为”web9\_extra”的项目（Web应用程序）。在项目中新建“JSP”文件，命名为“comment”。comment.jsp提供评论界面。

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<html>

<head>

<title>评论</title>

</head>

<body>

<h1>输入评论内容</h1>

<form action="<%= request.getContextPath()%>/comment" method="post">

<textarea name="comment" cols="30" rows="10"></textarea>

<input type="submit" value="提交">

</form>

<p><%= request.getAttribute("tag") == null ? "" : request.getAttribute("tag")%>

<span style="color:red"><%=request.getAttribute("comment") == null ? "" : request.getAttribute("comment")%></span>

</p>

</body>

</html>

（2）新建“it.servlet”包，在该包中新建java类，命名为“CommentServlet”。

CommentServlet.java的功能：根据评论中是否有敏感词，转发不同的数据给评论界面。

package it.servlet;

import javax.servlet.\*;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

import java.io.IOException;

import java.util.ArrayList;

import java.util.List;

import java.util.Objects;

@WebServlet("/comment")

public class CommentServlet extends HttpServlet {

private List<String> sensitiveWords = new ArrayList<>();

@Override

public void init() throws ServletException {

sensitiveWords.add("山寨");

sensitiveWords.add("盗版");

sensitiveWords.add("水货");

}

@Override

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

//【代码一】获取评论内容（注意：post请求体中的数据的中文处理）

request.setCharacterEncoding("utf-8");

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

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

String originalComment = comment; //保存原评论

for (String sensitiveWord : sensitiveWords) {

//对所有敏感词汇进行过滤

if (comment.contains(sensitiveWord)) {

//替换敏感词汇

comment = comment.replace(sensitiveWord, "\*\*");

}

}

if (/\*【代码二】\*/Objects.equals(originalComment, comment)) {

//没有敏感词，设置tag为good guy

request.setAttribute("tag", "good guy：");

} else {

//有敏感词，设置tag为bad guy

request.setAttribute("tag", "bad guy：");

}

//【代码三】将comment保存到request中

request.setAttribute("comment", comment);

//【代码四】跳转到comment.jsp页面（请求转发？还是重定向？）

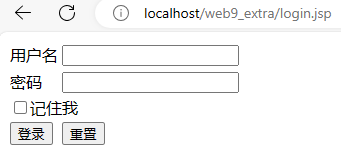
request.getRequestDispatcher("comment.jsp").forward(request, response);

}

}

2、使用cookie、session实现用户访问**welcome.jsp**页面时的7天免密登录的功能。具体要求如下：

（1）login.jsp：登录页面。



（2）LoginServlet：进行登录是否成功的判断。

当用户登录成功，将用户名保存到session中。

当用户登录成功，且在login.jsp中选择了“记住我”，将用户名和密码在cookie中保存7天。

（3）welcome.jsp：欢迎页面，显示“\*\*\*，欢迎您！”。

先从session中读取用户名。

如果session中无用户名，则从cookie中读取用户名和密码。如果从cookie中读到了正确的用户名和密码，则说明用户曾经在login.jsp中选择了“记住我”，那么把从cookie中读到的用户名显示在页面中，而无需用户重新登录。这样，就实现了welcome.jsp页面的7天免密登录。



（4）LogoutServlet：清除session以及cookie中的数据，实现“退出登录”的功能。

【实验步骤】：

（1）在项目中，新建JSP，命名为“login”。

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Title</title>

</head>

<body>

<div style="color:red"><%= request.getAttribute("error") == null ? "" : request.getAttribute("error")%></div>

<form action="<%= request.getContextPath()%>/login" method="post">

<table>

<tr>

<td>用户名</td><td><input type="text" name="username"></td>

</tr>

<tr>

<td>密码</td><td><input type="password" name="userpass"></td>

</tr>

<tr>

<td colspan="2"><input type="checkbox" name="remember" id="remember" value="yes"><label for="remember">记住我</label> </td>

</tr>

<tr>

<td><input type="submit" value="登录"></td>

<td><input type="reset" value="重置"></td>

</tr>

</table>

</form>

</body>

</html>

（2）在“it.servlet”包中新建java类，命名为“LoginServlet”。

LoginServlet.java的功能：登录验证以及7天免登录的判断。

package it.servlet;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

import java.io.IOException;

import java.util.Objects;

@WebServlet("/login")

public class LoginServlet extends HttpServlet {

@Override

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

//接收前端数据

request.setCharacterEncoding("utf-8");

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

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

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

if (/\*【代码一】登录成功的用户名为本人姓名，密码为本人学号\*/"江炜".equals(username) &&

"226709033".equals(userpass)) {

//登录成功

//勾选了“记住我”，remember的值为“yes”；如果没有勾选,remember的值为null

if (/\*【代码二】\*/Objects.equals("yes", remember)) {

Cookie cookie1 = new Cookie("username", username);

Cookie cookie2 = new Cookie("userpass", userpass);

//设置cookie携带路径为整个项目都携带

cookie1.setPath(request.getContextPath());

cookie2.setPath(request.getContextPath());

//设置cookie有效期(7天),cookie的过期时间是基于秒设置的

cookie1.setMaxAge(60 \* 60 \* 24 \* 7);

cookie2.setMaxAge(60 \* 60 \* 24 \* 7);

response.addCookie(cookie1);

response.addCookie(cookie2);

} else {

Cookie cookie1 = new Cookie("username", "");

Cookie cookie2 = new Cookie("userpass", "");

cookie1.setMaxAge(0);

cookie2.setMaxAge(0);

response.addCookie(cookie1);

response.addCookie(cookie2);

}

// 用户名存入session

HttpSession session = request.getSession();

session.setAttribute("username", username);

//重定向到welcome.jsp

response.sendRedirect(request.getContextPath() + "/welcome.jsp");

return;

}

//【代码三】登录失败，将错误信息“用户名或密码错误”转发给login.jsp

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

response.getWriter().println("用户名或密码错误");

}

}

（3）新建JSP，命名为“welcome”。welcome.jsp为欢迎页面。

<%@page contentType="text/html" pageEncoding="UTF-8" %>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<%

// 先判断session中是否有用户名

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

String userpass = null;

if (username == null) {

/\* session中没有用户名，即：用户没有经过LoginServlet的登录成功的验证（用户没有登录过）,

则判断Cookie中是否有用户名，如果有，则说明用户之前登录时选择了“记住我”\*/

//【代码四】从cookie中读取用户名和密码

Cookie[] cookies = request.getCookies();

for (Cookie cookie : cookies) {

if (cookie.getName().equals("username"))

username = cookie.getValue();

else if (cookie.getName().equals("userpass"))

userpass = cookie.getValue();

}

if (username != null && userpass != null) { //用户选择过“记住我”（免密登录）

if (/\*这里要修改被比较的用户名为本人姓名，被比较的密码为本人学号\*/"江炜".equals(username) && "226709033".equals(userpass)) {

//【代码五】cookie中的用户名和密码正确，将用户名存储到session

/\* 存储到session的目的是，之后只要还是在一次会话中访问welcome.jsp，则直接从session中读取用户名，而不再访问cookie\*/

session.setAttribute("username", username);

} else {// cookie中的用户名或密码不正确，跳转到登录页面

response.sendRedirect(request.getContextPath() + "/login.jsp");

return;

}

} else {//session中无用户名；cookie中无用户名或密码（用户没有选择免密登录），则跳转到登录页面

response.sendRedirect(request.getContextPath() + "/login.jsp");

return;

}

}

%>

<%--session中有用户名；或者：session中无用户名，但cookie中存储了正确的用户名和密码，则显示“\*\*\*，欢迎您！”--%>

<%=username%>，欢迎您！<a href="<%=request.getContextPath()%>/logout">退出登录</a>

</body>

</html>

（4）在java包“it.servlet”中新建java类，命名为“LogoutServlet”。

LogoutServlet.java的功能：退出登录。

package it.servlet;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

@WebServlet("/logout")

public class LogoutServlet extends HttpServlet {

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

/\* 【代码六】清除当前登录用户名（session中的username）以及cookie中的username和userpass \*/

req.getSession().removeAttribute("username");

Cookie username = new Cookie("username", null);

username.setMaxAge(0);

Cookie userpass = new Cookie("userpass",null);

userpass.setMaxAge(0);

resp.addCookie(username);

resp.addCookie(userpass);

resp.sendRedirect(req.getContextPath() + "/login.jsp");

}

}

调试过程：

实验二在欢迎界面没有正确显示中文，这是因为在LoginServlet.java文件中没有添加上代码response.setContentType("text/html;charset=utf-8");

实验一结果：



图 1无敏感词



图 2有敏感词

实验二结果

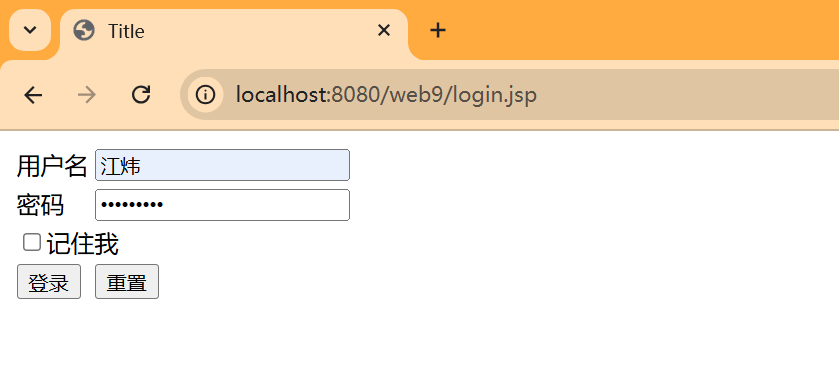


图 3不选记住我登录

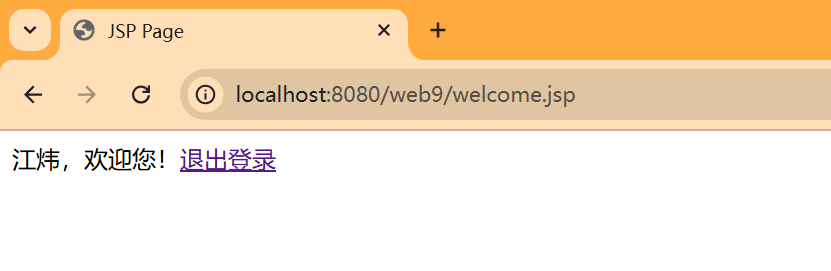


图 4登录成功页面

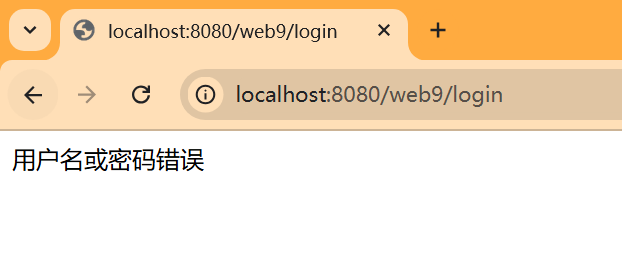


图 5登录失败页面

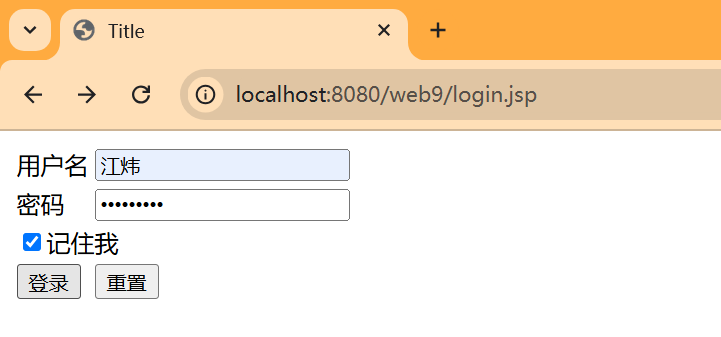


图 6选择记住我登录

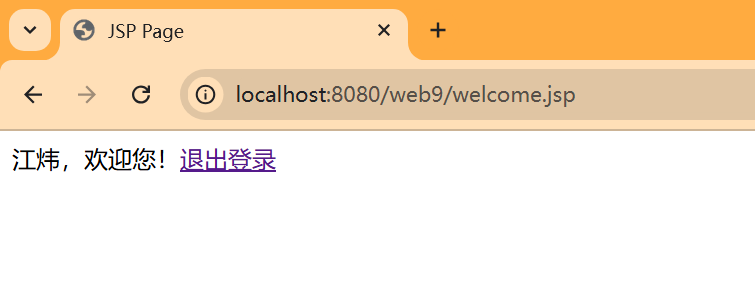


图 7关闭浏览器后仍然可以重新访问该页面