# **《Servlet实验》**

## 一、实验目的

1. 理解 Servlet 的基本概念和工作原理，掌握其在 Web 开发中的作用。
2. 学会创建、配置和部署 Servlet，能够处理简单的 HTTP 请求和响应。
3. 掌握 Servlet 与 HTML 表单的交互方式，实现用户数据的接收和处理。

## 实验内容

### **简单的用户信息显示 Servlet** 创建一个 Servlet，接收来自 HTML 表单的用户姓名和年龄信息（表单使用 POST 方法提 交），然后在 Servlet 中获取这些信息并在页面上显示 “姓名：[用户姓名]，年龄：[用户年龄]”。

1. 在 Eclipse（或其他 IDE）中创建一个新的 Java 类，继承javax.servlet.http.HttpServlet。

package com;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

@WebServlet("/UserInfoServlet")

public class UserInfoServlet extends HttpServlet {

    private static final long serialVersionUID = 1L;

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

        request.setCharacterEncoding("UTF-8");

        response.setCharacterEncoding("UTF-8");

        // 获取表单提交的姓名和年龄参数

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

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

        // 设置响应内容类型

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

        // 输出用户信息到页面上

        response.getWriter().append("姓名: ").append(name).append(", 年龄: ").append(age);

    }

}

1. 重写doPost方法，在方法内使用HttpServletRequest的getParameter方法获取表单提交的姓名和年龄参数。
2. 设置HttpServletResponse的内容类型为text/html，并通过getWriter方法获取输出流，将用户信息输出到页面上。

<h1>请输入您的信息</h1>

<form action="UserInfoServlet" method="post">

    姓名: <input type="text" name="name"><br>

    年龄: <input type="text" name="age"><br>

    <input type="submit" value="提交">

</form>

1. 在web.xml文件中配置 Servlet 的名称和映射路径。

  <servlet>

    <servlet-name>UserInfoServlet</servlet-name>

    <servlet-class>com.UserInfoServlet</servlet-class>

</servlet>

<servlet-mapping>

    <servlet-name>UserInfoServlet</servlet-name>

    <url-pattern>/UserInfoServlet</url-pattern>

</servlet-mapping>

### **Servlet 实现简单的用户登录验证**

创建一个模拟用户登录的 Servlet。假设存在一个固定的用户名 “admin” 和密码 “123456”。接收来自 HTML 表单的用户名和密码（POST 方法提交），如果用户名和密码正确，显示 “登录成功”，否则显示 “用户名或密码错误”。

1. 创建LoginServlet类继承HttpServlet，重写doPost方法。

package com;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

@WebServlet("/LoginServlet")

public class LoginServlet extends HttpServlet {

    private static final long serialVersionUID = 1L;

    // 预设的用户名和密码

    private static final String USERNAME = "admin";

    private static final String PASSWORD = "123456";

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

        request.setCharacterEncoding("UTF-8");

        response.setCharacterEncoding("UTF-8");

        // 获取表单提交的用户名和密码

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

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

        // 设置响应内容类型

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

        // 验证用户名和密码

        if (USERNAME.equals(username) && PASSWORD.equals(password)) {

            response.getWriter().append("登录成功");

        } else {

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

        }

    }

}

1. 获取表单提交的用户名和密码，与预设的用户名和密码进行比较。
2. 根据比较结果设置不同的响应内容并输出到页面。

<form action="LoginServlet" method="post">

    用户名: <input type="text" name="username"><br>

    密码: <input type="password" name="password"><br>

    <input type="submit" value="登录">

</form>

1. 在web.xml中配置 Servlet 的相关信息

<servlet>

    <servlet-name>LoginServlet</servlet-name>

    <servlet-class>com.LoginServlet</servlet-class>

</servlet>

<servlet-mapping>

    <servlet-name>LoginServlet</servlet-name>

    <url-pattern>/LoginServlet</url-pattern>

</servlet-mapping>

## 三、实验环境

1. 操作系统：windows操作系统
2. Web服务器：Tomcat8/Tomcat9
3. 开发工具:Eclipse 或其他 IDE