

序号: 24

学号: 16430121



常州大学

实 验 报 告

课 程 名 称: Java EE 程序设计与应用开发

实 验 题 目: Ajax 程序设计

实 验 次 数: 第 三 次实验 实验时间: 2019 年 4 月 23 日

学 生 姓 名: 林锦雄

学 院: 信息数理 专 业 班 级: 计算机 162

指 导 教 师: 陆洁茹 评 阅 成 绩:

一、实验目的：

1. 了解 Ajax 的基本构成和作业，掌握简单的 Ajax 程序的编写方法。
2. 要求： 实现 Ajax 程序，并使用 Ajax 技术访问数据库。

二、实验仪器、设备

1、硬件环境

PC 微机；2G 以上内存；VGA 显示格式。

2、软件环境

Windows XP 以上操作系统，JDK，Tomcat 服务器等。

三、实验内容与要求

在数据库中建立表格 T_CUSTOMER(ACCOUNT,PASSWORD,CNAME)，插入一些记录。如图 1 所示。

```
Database changed
MariaDB [books]> select * from T_CUSTOMER;
+-----+-----+-----+
| ACCOUNT | PASSWORD | CNAME |
+-----+-----+-----+
| admin   | admin    | 管理员 |
| asdfgh  | hgfdsa   | 用户 B |
| qwe     | qwel23456 | 用户 A |
| youke   | youkel   | 游客   |
+-----+-----+-----+
4 rows in set (0.000 sec)

MariaDB [books]> 
```

图 1. books 数据库中的 T_CUSTOMER 表

1. 编写注册页面，输入账号、密码、确认密码、姓名、完成注册的功能，并能够判断是否重复注册。提交后界面不刷新。

register.jsp 为注册页面，上方显示“欢迎注册”信息，下方是注册信息表，需用户填写注册相关信息来进行注册。如图 2 所示。

该注册网页使用 Ajax 技术实现表单异步提交，即在注册并等待后台响应的时候，网页并不会发生改变（变成一片空白），后台响应的结果会在网页接收到后显示到指定位置（注册页面的下方）。

系统会检查用户填写的注册信息，如果没有填写完整的注册信息便进行注册，系统会弹窗提示“请填写完整的注册信息!”，如图 3 所示。



图 2. 注册页面

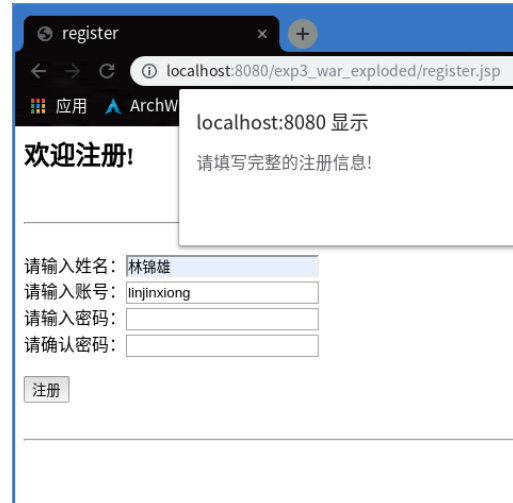


图 3. 注册信息不完整

```

1. <!-- register.jsp -->
2. <script language="JavaScript">
3.     function register() {
4.         var xmlhttp = false;
5.         if(window.XMLHttpRequest) {
6.             xmlhttp = new XMLHttpRequest();
7.         } else if(window.ActiveXObject) {
8.             xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
9.         }
10.        var account = document.registerForm.account.value;
11.        var password = document.registerForm.password.value;
12.        var confirmPassword = document.registerForm.confirmPassword.value;
13.        var name = document.registerForm.name.value;
14.        if(account==" " | password==" " | confirmPassword==" " | name==" ") {
15.            window.alert("请填写完整的注册信息!");
16.        } else {
17.            var url = "registerServlet?account=" + account + "&password=" + pas
18.            sword + "&confirmPassword=" + confirmPassword + "&name=" + name;
19.            xmlhttp.open("post",url,true);
20.            xmlhttp.onreadystatechange = function () {
21.                if(xmlhttp.readyState == 4) {
22.                    resultInfo.innerHTML = xmlhttp.responseText;
23.                } else {
24.                    resultInfo.innerHTML += "正在注册,请稍后...";
25.                }
26.            }
27.            xmlhttp.send();
28.        }
29.    </script>
30.    <h2>欢迎注册!</h2><br><hr><br>
31.    <form name="registerForm">
32.        请输入姓名: <input type="text" name="name"><br>
33.        请输入账号: <input type="text" name="account"><br>
34.        请输入密码: <input type="password" name="password"><br>
35.        请确认密码: <input type="password" name="confirmPassword"><br><br>
36.        <input type="button" value="注册" onclick="register()"><br><br>
37.    </form><hr>
38.    <div id="resultInfo"></div>

```

registerServlet 是注册处理类，把由前端 register.jsp 传来的注册信息来进行账号注册。

如果用户输入的注册密码和确认密码不一致，则返回“确认密码不一致,请重新输入!”信息，提示用户重新输入。如图 4 所示。



图 4. 密码和确认密码不一致

registerServlet 检查用户输入的账号，如果注册信息无误，并且通过 registerDao 调用数据库查询注册账号没有存在于数据库中，则把该新账号存入数据库，并返回“注册成功!”，提示用户注册成功。如图 5 所示。

调用 registerDao 调用数据库查询注册账号，发现其在数据库中已存在，则返回“注册失败!账号已存在!”，提示用户注册失败。如图 6 所示。



图 5. 注册成功



图 6. 重复注册

```
1. @WebServlet(name = "registerServlet")
2. public class registerServlet extends HttpServlet {
3.     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
4.         request.setCharacterEncoding("UTF-8");
```

```

5.         response.setCharacterEncoding("UTF-8");
6.         PrintWriter pw = response.getWriter();
7.         String account = request.getParameter("account");
8.         String password = request.getParameter("password");
9.         String confirmPassword = request.getParameter("confirmPassword");
10.        String name = request.getParameter("name");
11.        if(!password.equals(confirmPassword)) {
12.            pw.println("确认密码不一致,请重新输入!");
13.            return;
14.        }
15.        RegisterDao registerDao = new RegisterDao();
16.        try {
17.            boolean result = registerDao.confirmRegister(account,password,name)
;
18.            if(result == true) {
19.                pw.println("注册成功!");
20.            } else {
21.                pw.println("注册失败!账号已存在");
22.            }
23.        } catch (ClassNotFoundException e) {
24.            e.printStackTrace();
25.        } catch (SQLException e) {
26.            e.printStackTrace();
27.        }
28.        pw.close();
29.    }

```

```

1. //RegisterDao.java
2. public class RegisterDao {
3.
4.     private String driver = "com.mysql.jdbc.Driver";
5.     private String url = "jdbc:mysql://localhost:3306/books?characterEncoding=UTF-8";
6.     private String user = "root";
7.     private String passwd = "bingjie123";
8.
9.     public boolean confirmRegister(String account, String password, String name) thro
ws ClassNotFoundException, SQLException {
10.        Class.forName(driver);
11.        Connection conn = DriverManager.getConnection(url,user,passwd);
12.        String sql = "select * from T_CUSTOMER where ACCOUNT = '" + account + "'";
13.        Statement stat = conn.createStatement();
14.        ResultSet rs = stat.executeQuery(sql);
15.        if(rs.next()) {
16.            return false;
17.        }
18.
19.        sql = "insert into T_CUSTOMER (ACCOUNT, PASSWORD, CNAME) value (?,?,?)";
20.        PreparedStatement ps = conn.prepareStatement(sql);
21.        ps.setString(1,account);
22.        ps.setString(2,password);
23.        ps.setString(3,name);
24.        ps.executeUpdate();
25.
26.        stat.close();
27.        rs.close();
28.        ps.close();
29.        conn.close();

```

```

30.     return true;
31. }
32. }

```

2. 制作一个登录页面，输入账号和密码，如果匹配，则显示“登录成功”，否则显示“登录失败”。界面不刷新。

login.jsp 为登录页面，上方显示“欢迎登录”信息，下方是登录信息表，需用户填写自己的账号和密码来进行登录，如果没有账号，可以点击注册按钮进入注册页面注册新账号。如图 7 所示。

该登录网页使用 Ajax 技术实现表单异步提交，即在注册并等待后台响应的时候，网页并不会发生改变（变成一片空白），后台响应的登录结果会在网页接收到后显示到指定位置（登录页面的下方）。

登录时如果账号或密码信息不完整，网页会弹出提示框“请填写完整的登陆信息!”提示用户填写完整的登录资料进行登录。如图 8 所示。

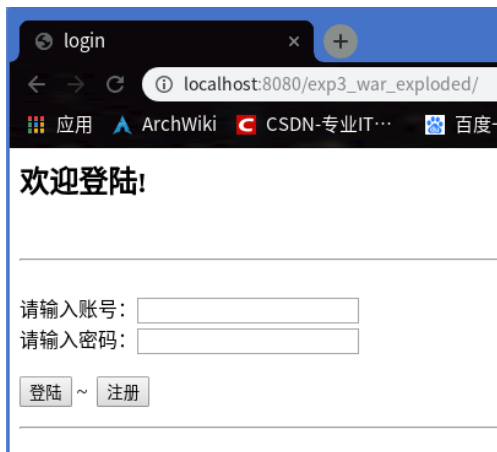


图 7. 登录页面

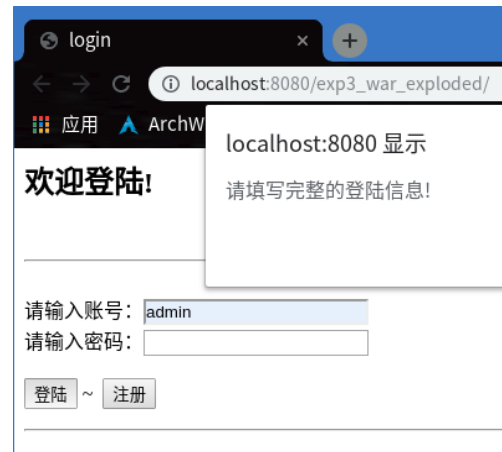


图 8. 登录信息不完整

```

1. 1. -- login.jsp -->
2. <script language="JavaScript">
3.     function login() {
4.         var xmlhttp = false;
5.         if(window.XMLHttpRequest) {
6.             xmlhttp = new XMLHttpRequest();
7.         } else if(window.ActiveXObject) {
8.             xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
9.         }
10.        var account = document.loginForm.account.value;
11.        var password = document.loginForm.password.value;
12.        if(account == "" || password == "") {
13.            window.alert("请填写完整的登陆信息!");
14.        } else {
15.            var URL = "loginServlet?account=" + account + "&password=" + password;
16.            xmlhttp.open("post",URL,true);
17.            xmlhttp.onreadystatechange = function () {
18.                if(xmlhttp.readyState == 4) {
19.                    resultInfo.innerHTML = xmlhttp.responseText;
20.                } else {

```

```

21.         resultInfo.innerHTML += "正在登陆,请稍候...";
22.     }
23. }
24. xmlhttp.send();
25. }
26. }
27. </script>
28. <h2>欢迎登陆!</h2><br><hr><br>
29. <form method="post" action="register.jsp" name="loginForm">
30.     请输入账号: <input type="text" name="account"><br>
31.     请输入密码: <input type="password" name="password"><br><br>
32.     <input type="button" value="登陆" onclick="login()"> ~
33.     <input type="submit" value="注册"><br>
34. </form><hr>
35. <div id="resultInfo"></div>

```

loginServlet 检查用户输入的账号和密码,通过 loginDao 类调用数据库查询用户输入的账号是否存在,如果不存在,则返回“登陆失败,账号不存在!”,提示用户输入的账号不存在。如图 9 所示。

如果用户输入的账号在数据库中能找到,但查询数据库发现用户输入的密码不正确,则返回“登陆失败,密码错误!”,提示用户登录密码错误。如图 10 所示。

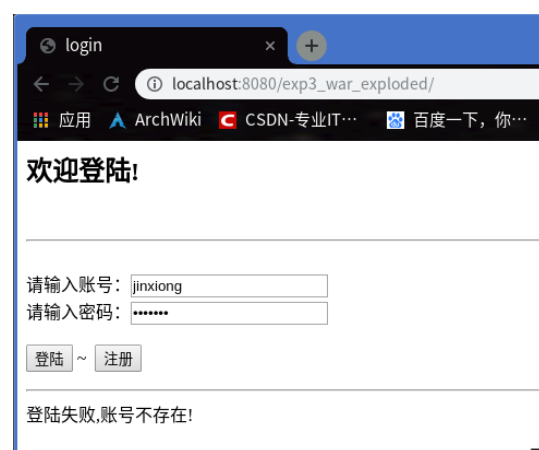


图 9. 账号不存在

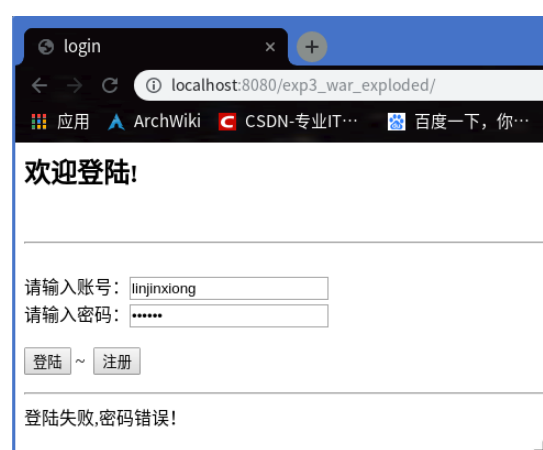


图 10. 登录密码错误

用户输入的账号在数据库中能找到并且密码也正确对应,则返回“登陆成功,欢迎某某用户!”提示信息。如图 11 所示。

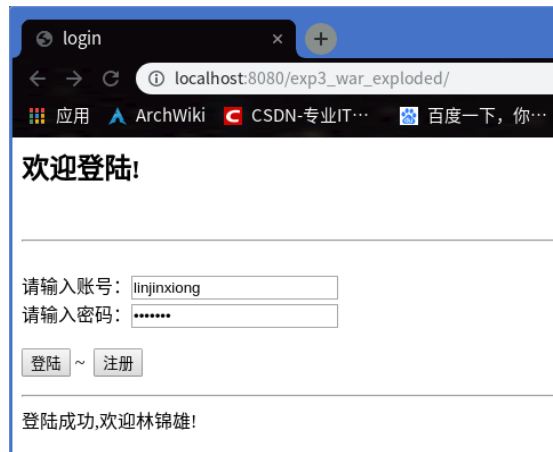


图 11. 登录成功

```

1. @WebServlet(name = "loginServlet")
2. public class loginServlet extends HttpServlet {
3.     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
4.         request.setCharacterEncoding("UTF-8");
5.         response.setCharacterEncoding("UTF-8");
6.         PrintWriter pw = response.getWriter();
7.         String account = request.getParameter("account");
8.         String password = request.getParameter("password");
9.         LoginDao loginDao = new LoginDao();
10.        try {
11.            String name = loginDao.queryAccount(account,password);
12.            if(name == null) {
13.                pw.println("登陆失败,账号不存在!");
14.            } else if(name.equals("-1")) {
15.                pw.println("登陆失败,密码错误!");
16.            } else {
17.                pw.println("登陆成功,欢迎" + name + "!");
18.            }
19.        } catch (ClassNotFoundException e) {
20.            e.printStackTrace();
21.        } catch (SQLException e) {
22.            e.printStackTrace();
23.        }
24.        pw.close();
25.    }
26. }

```

```

1. //LoginDao.java
2. public class LoginDao {
3.
4.     private String driver = "com.mysql.jdbc.Driver";
5.     private String url = "jdbc:mysql://localhost:3306/books?characterEncoding=UTF-8";
6.     private String user = "root";
7.     private String passwd = "bingjie123";
8.
9.     public String queryAccount(String account, String password) throws ClassNotFoundException, SQLException {
10.        Class.forName(driver);
11.        Connection conn = DriverManager.getConnection(url,user,passwd);

```



```

12.      String sql = "select * from T_CUSTOMER where ACCOUNT = ?";
13.      PreparedStatement ps = conn.prepareStatement(sql);
14.      ps.setString(1,account);
15.      ResultSet rs = ps.executeQuery();
16.      String ans = null;
17.      while(rs.next()) {
18.          String pw = rs.getString(2);
19.          String na = rs.getString(3);
20.          ans = "-1";
21.          if(pw.equals(password)) {
22.              ans = na;
23.              break;
24.          }
25.      }
26.      rs.close();
27.      ps.close();
28.      conn.close();
29.      return ans;
30.  }
31. }

```

四、实验心得

本次实验，让我们了解了 Ajax 的基本构成和作业，掌握简单的 Ajax 程序的编写方法，在网页表单提交时，使用 Ajax 异步提交的方式，可以在让表单在后台异步进行提交，前端网页不变，等待 servlet 传回处理好的数据，这样，用户就可以在网络拥堵的环境下，工作的同时继续浏览网页的其他信息，这样大大地改善了用户的浏览体验。

Ajax 技术是改善用户浏览网页体验的技术，熟练掌握其技术原理，以便写出更好更优良的网页，达到让用户拥有更好体验的效果。