**实验六 JavaBean应用**

## 一、实训目标

1、学习在JSP文件中编写JavaBean的方法；

2、掌握JavaBean的编写、set和get方法、JavaBean与普通Java类的区别与联系。

## 二、实训内容

1. 在IDEA中创建JavaBean。创建一个名为Course的JavaBean，放在名为com.mymodel.course的包中，成员变量有cno、cname、cterm、cmajor、chour、ccredit、tno，对应每个成员变量需要提供set()和get()方法。

具体创建步骤如下：

（1）在IDEA中，新建Web Project，项目名为ch07；

（2）右击项目中的src->main->java目录，依次选择New—>Package，在弹出的“New Java Package”对话框中输入包名com.mymodel.course，单击“Finish”按钮完成包的创建。

（3）右击新建的包com.mymodel.course，依次选择New—>Class，弹出“New Java Class”对话框。输入要创建的类名Course。单击“Finish”按钮。

（4）在类中添加cno、cname、cterm、cmajor、chour、ccredit、tno等属性。

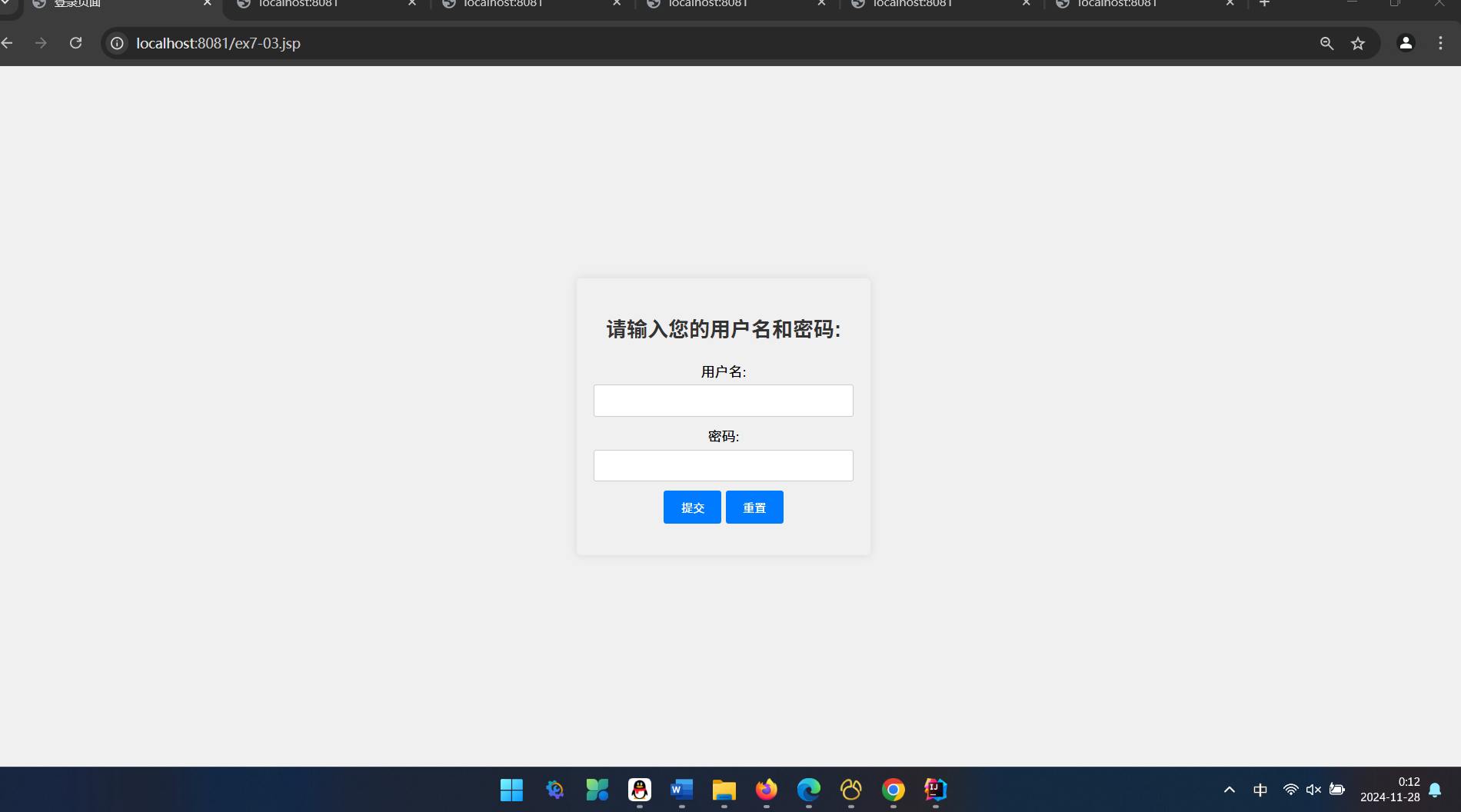
（6）在属性下方的位置处按鼠标右键，依次选择Generate ->Getter and Setter，弹出对话框。选中属性，单击“OK”按钮，这样就生成了所有成员变量的set和get方法。

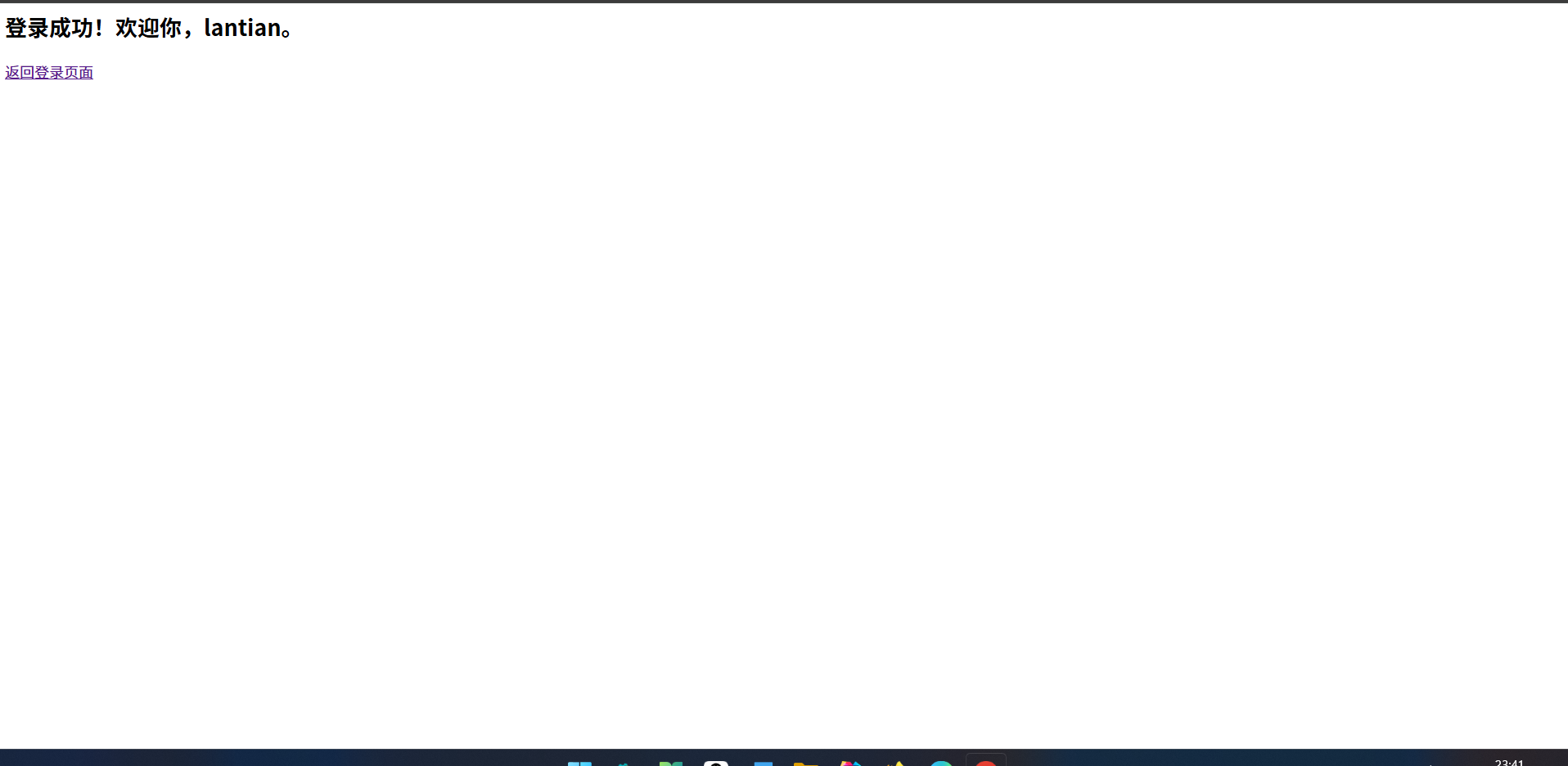
|  |
| --- |
| 此处插入代码。  package com.mymodel.course;  public class Course {  private String cno;  private String cname;  private String cterm;  private String cmajor;   public String getCno() {  return cno;  }   public void setCno(String cno) {  this.cno = cno;  }   public String getCname() {  return cname;  }   public void setCname(String cname) {  this.cname = cname;  }   public String getCterm() {  return cterm;  }   public void setCterm(String cterm) {  this.cterm = cterm;  }   public String getCmajor() {  return cmajor;  }   public void setCmajor(String cmajor) {  this.cmajor = cmajor;  }   public int getChour() {  return chour;  }   public void setChour(int chour) {  this.chour = chour;  }   public double getCcredit() {  return ccredit;  }   public void setCcredit(double ccredit) {  this.ccredit = ccredit;  }   public String getTno() {  return tno;  }   public void setTno(String tno) {  this.tno = tno;  }   private int chour;  private double ccredit;  private String tno; } |

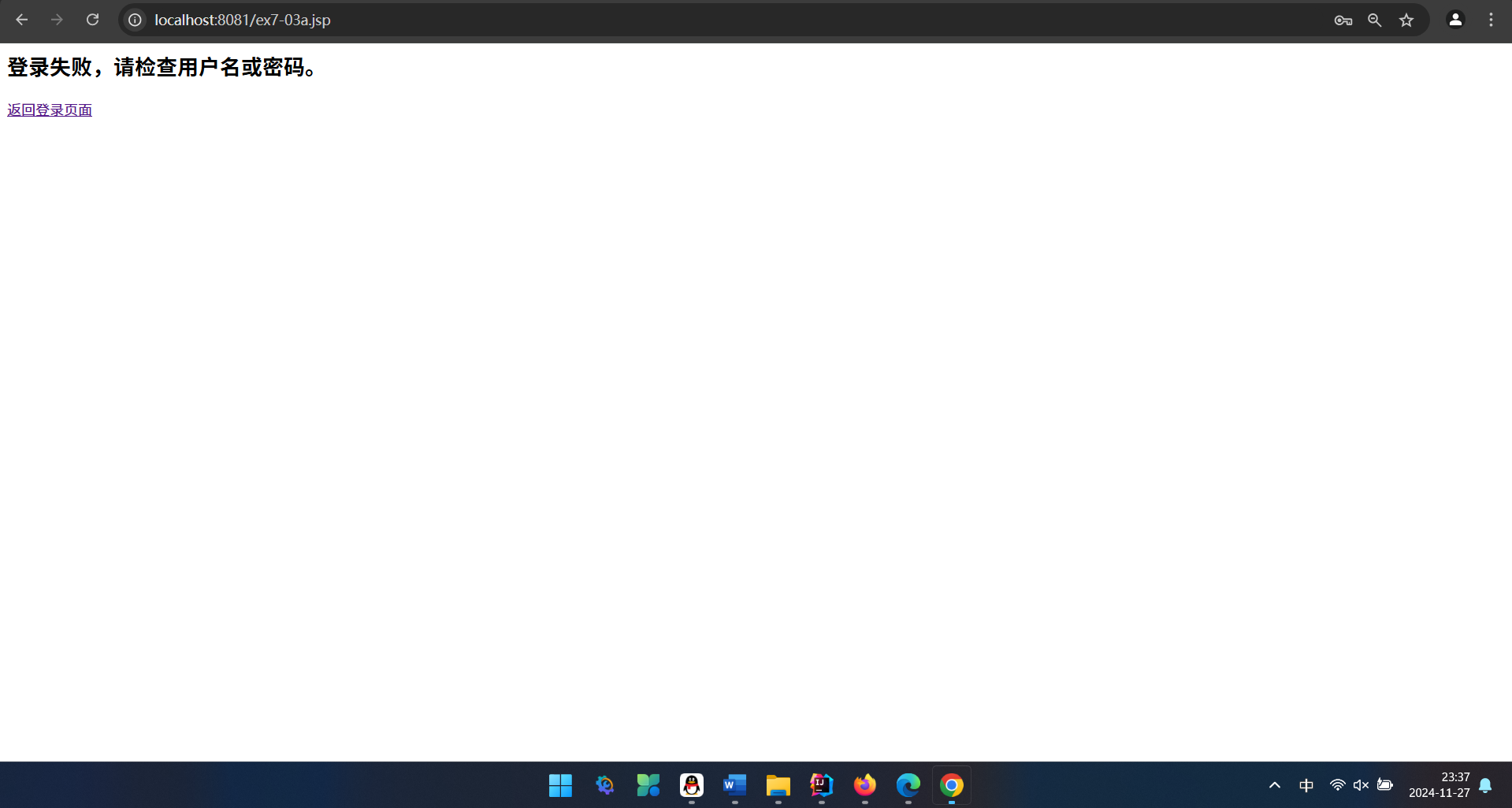
2. JavaBean与HTML表单的交互。首先在com.mymodel.user包中建立一个类Login，用来封装用户表单提交的数据，并且编写布尔类型的check方法，判断用户输入的用户名是不是”lantian”,密码是不是”123456”,如果是，网页提示用户，登录成功，否则登录失败。编写文件ex7-03.jsp，页面运行如图1所示，代码中文本框和密码框的name属性必须分别为userName和password。输入用户名和密码，点击“提交”按钮，将页面提交给ex7-03a.jsp。

****

图1 登录页面







|  |
| --- |
| 此处插入代码。  package com.mymodel.user;  public class Login {  private String userName;  private String password;   // 构造函数  public Login() {}   // Getter 和 Setter 方法  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;  }   // 检查用户名和密码的方法  public boolean check() {  if ("lantian".equals(userName) && "123456".equals(password)) {  return true; // 登录成功  } else {  return false; // 登录失败  }  } }  <%@ **page** contentType="text/html;charset=UTF-8" language="java" %> <html> <head>  <meta charset="UTF-8">  <title>登录页面</title>  <style>  body {  font-family: Arial, sans-serif;  margin: 0;  padding: 0;  display: flex;  justify-content: center;  align-items: center;  height: 100vh;  background-color: #f0f0f0;  }   .container {  width: 300px;  padding: 20px;  border-radius: 5px;  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  text-align: center;  }   h2 {  color: #333;  }   label {  display: block;  margin-bottom: 5px;  }   input[type="text"], input[type="password"] {  width: 100%;  padding: 10px;  margin-bottom: 10px;  border: 1px solid #ccc;  border-radius: 3px;  }   button {  padding: 10px 20px;  background-color: #007BFF;  color: white;  border: none;  cursor: pointer;  border-radius: 3px;  }   button:hover {  background-color: #0056b3;  }  </style> </head> <body> <div class="container">  <h2>请输入您的用户名和密码:</h2>  <form id="loginForm" action="ex7-03a.jsp" method="post">  <label for="username">用户名:</label>  <input type="text" id="username" name="userName" required>  <label for="password">密码:</label>  <input type="password" id="password" name="passWord" required>  <button type="submit">提交</button>  <button type="reset">重置</button>  </form> </div>  <script>  ***document***.getElementById('loginForm').addEventListener('submit', function(event) {  var username = ***document***.getElementById('username').value;  var password = ***document***.getElementById('password').value;   if (!username || !password) {  alert('用户名和密码不能为空！');  event.preventDefault();  }  }); </script> </body> </html> |

3．基于数据库的登录验证。

（1）在ch07项目下定义名为com.mymodel.user的包。在该包中创建类User，用来封装用户表单提交的数据。

（2）新建com.db包，在包中创建一个类DBConn，作为用来封装数据库连接的JavaBean。

（3）在com.mymodel.user包中再创建一个类UserOper，用来进行逻辑处理。添加一个login()方法，用来实现基于数据库的登录验证。

（4）编写文件ex7-04.jsp，运行ex7-04.jsp，输入用户名和密码，点击“提交”按钮。将页面提交给ex7-04a.jsp。只有当输入的用户名和密码与login表中的数据相同时，才会显示欢迎信息。这就实现了基于数据库的登录验证。

（5）启动Tomcat服务器后，在IE的地址栏中输入http://localhost:8080/ch07/ex7-04.jsp。程序运行结果如图2、3所示。

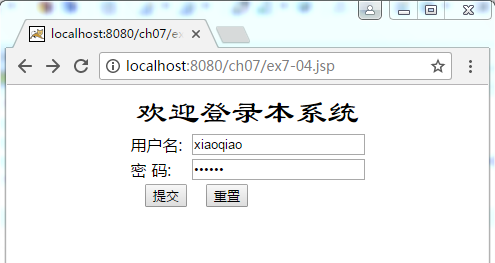
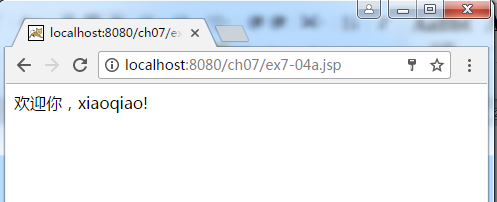
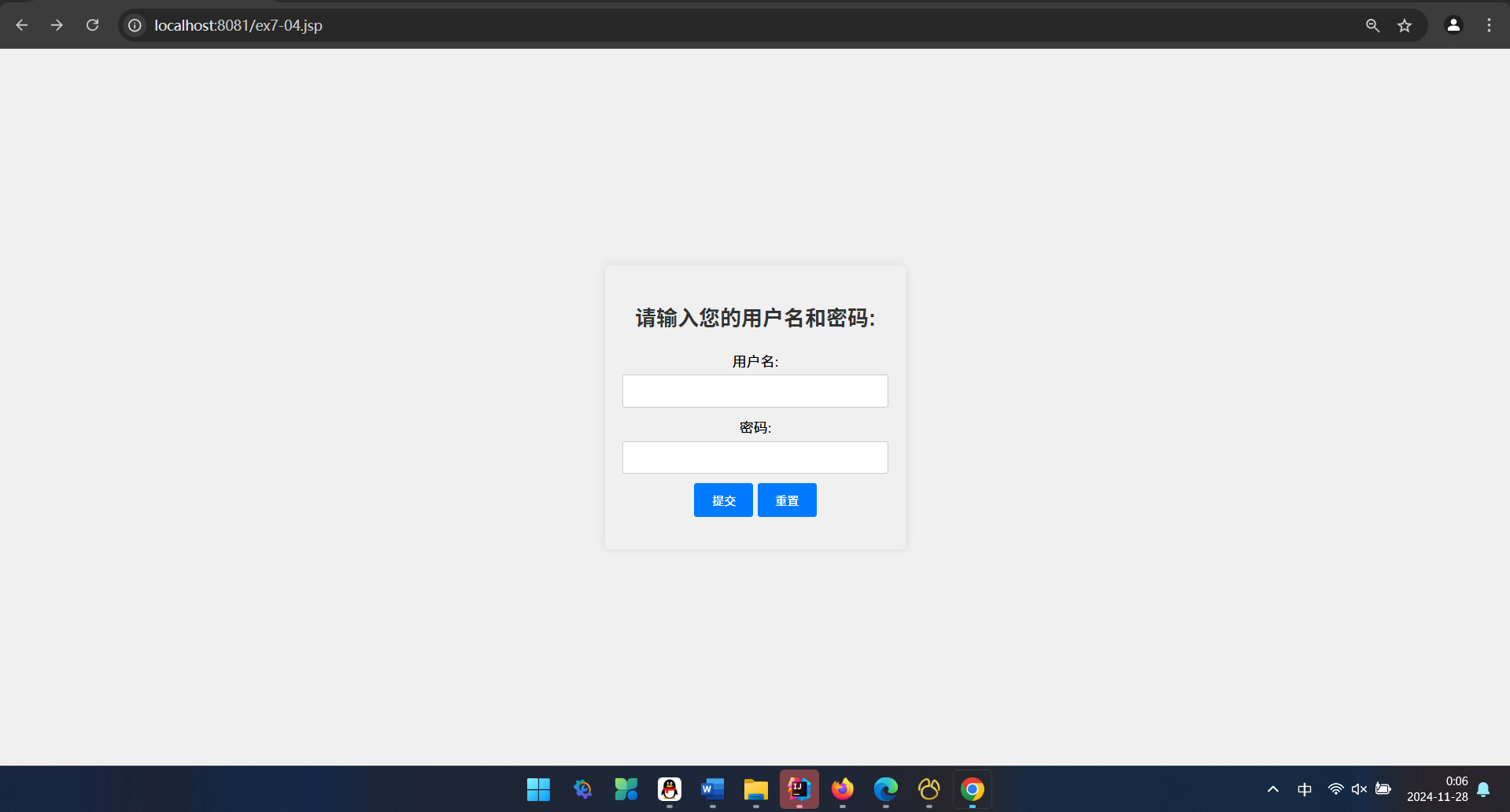


图2 登录页面 图3 登录成功页面





|  |
| --- |
| 此处插入代码。  package com.db;  import java.sql.Connection; import java.sql.DriverManager; import java.sql.SQLException;  public class DBConn {  private static final String *URL* = "jdbc:mysql://localhost:3306/test?useUnicode=true&characterEncoding=UTF-8";  private static final String *USER* = "root";  private static final String *PASSWORD* = "123456";   public static Connection getConnection() throws SQLException, ClassNotFoundException {  Class.*forName*("com.mysql.jdbc.Driver");  return DriverManager.*getConnection*(*URL*, *USER*, *PASSWORD*);  }  }  package com.mymodel.user;  import com.db.DBConn; import java.sql.Connection; import java.sql.PreparedStatement; import java.sql.ResultSet; import java.sql.SQLException; import java.util.logging.Level; import java.util.logging.Logger;  public class UserOper {  public boolean login(User user) {  try (Connection conn = DBConn.*getConnection*();  PreparedStatement ps = conn.prepareStatement("SELECT \* FROM login WHERE username=? AND password=?")) {  ps.setString(1, user.getUserName());  ps.setString(2, user.getPassword());   ResultSet rs = ps.executeQuery();  return rs.next(); // 如果有结果返回，则表示用户名和密码匹配  } catch (SQLException | ClassNotFoundException ex) {  Logger.*getLogger*(UserOper.class.getName()).log(Level.*SEVERE*, null, ex);  }  return false; // 出现异常或没有匹配的记录时返回false  } }  <%--  Created by IntelliJ IDEA.  User: Jake  Date: 2024-11-27  Time: 23:56  To change this template use File | Settings | File Templates. --%> <%@ **page** contentType="text/html;charset=UTF-8" language="java" %> <html> <head>  <meta charset="UTF-8">  <title>登录页面</title>  <style>  body {  font-family: Arial, sans-serif;  margin: 0;  padding: 0;  display: flex;  justify-content: center;  align-items: center;  height: 100vh;  background-color: #f0f0f0;  }   .container {  width: 300px;  padding: 20px;  border-radius: 5px;  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  text-align: center;  }   h2 {  color: #333;  }   label {  display: block;  margin-bottom: 5px;  }   input[type="text"], input[type="password"] {  width: 100%;  padding: 10px;  margin-bottom: 10px;  border: 1px solid #ccc;  border-radius: 3px;  }   button {  padding: 10px 20px;  background-color: #007BFF;  color: white;  border: none;  cursor: pointer;  border-radius: 3px;  }   button:hover {  background-color: #0056b3;  }  </style> </head> <body> <div class="container">  <h2>请输入您的用户名和密码:</h2>  <form id="loginForm" action="ex7-04a.jsp" method="post">  <label for="username">用户名:</label>  <input type="text" id="username" name="userName" required>  <label for="password">密码:</label>  <input type="password" id="password" name="passWord" required>  <button type="submit">提交</button>  <button type="reset">重置</button>  </form> </div>  <script>  ***document***.getElementById('loginForm').addEventListener('submit', function(event) {  var username = ***document***.getElementById('username').value;  var password = ***document***.getElementById('password').value;   if (!username || !password) {  alert('用户名和密码不能为空！');  event.preventDefault();  }  }); </script> </body> </html>  <%--  Created by IntelliJ IDEA.  User: Jake  Date: 2024-11-27  Time: 23:57  To change this template use File | Settings | File Templates. --%> <%@ **page** contentType="text/html;charset=UTF-8" language="java" %> <%@ **page** import="com.mymodel.user.User, com.mymodel.user.UserOper" %> <%@ **page** contentType="text/html;charset=UTF-8" language="java" %> <%  // 获取表单数据  String userName = request.getParameter("userName");  String password = request.getParameter("passWord");   // 创建User对象并设置属性  User user = new User(userName,password);   // 使用UserOper进行登录验证  UserOper userOper = new UserOper();  boolean result = userOper.login(user); %> <html> <head>  <title>登录结果</title> </head> <body> <h2>  <% if (result) { %>  登录成功！欢迎你，<%= userName %>。  <% } else { %>  登录失败，请检查用户名或密码。  <% } %> </h2> </body> </html> |