# 实验一 JSP+Servlet+Javabean+jdbc-基于MVC模式的数据库的应用

**一．实验目的：**

**1.掌握MVC模式的基本原理**

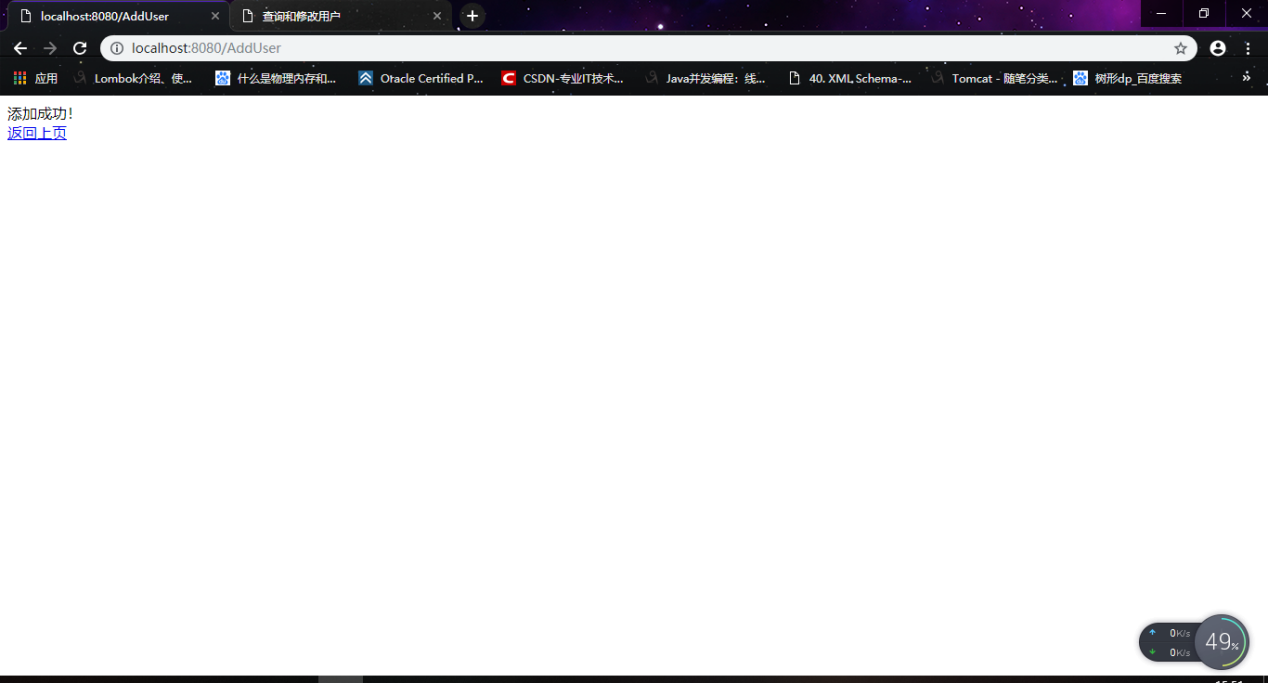
2.掌握JSP+Servlet+JavaBean架构连接、查询、删除数据库的方法；

**实验要求**：

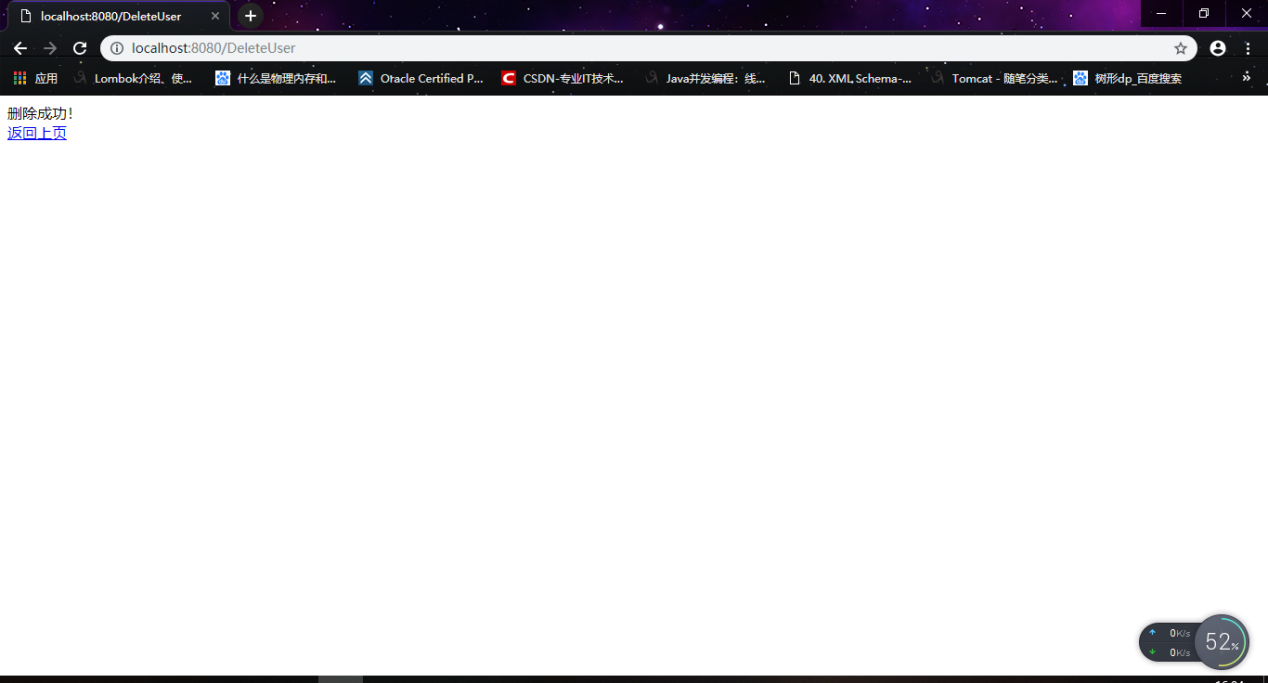
3.采用JSP+Servlet+JavaBean架构，设计通过下列JSP页面访问数据库(如表5-1)，具体要求如下：

1)．添加用户页面





1. . 删除用户页面



3)．查找和修改用户页面



**二 .实验步骤：**

1．准备需要访问的数据库

下载并安装MySQL5.7.1

Copy MySQL5.7.1驱动mysql-connector-java-5.0.8-bin.jar到%TOMCAT%\lib

文件夹下创建数据库及表：

数据库的设计如下表：

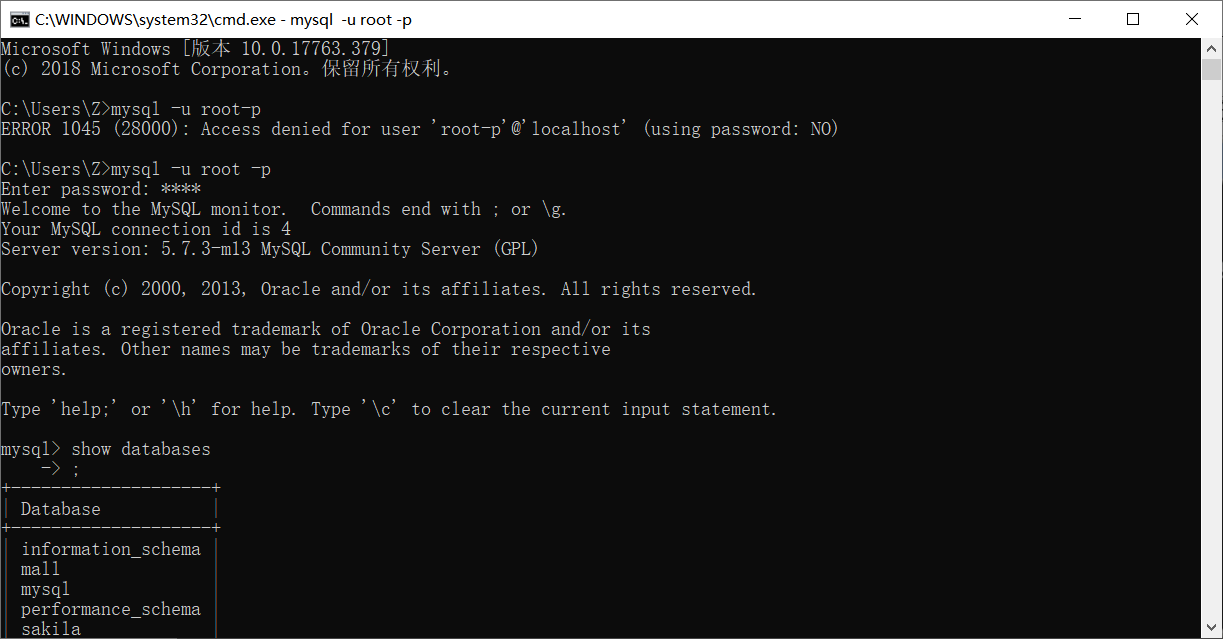
数据库采用MySQL5.0，数据库名：db，表名：user（用户表）

数据库的用户表

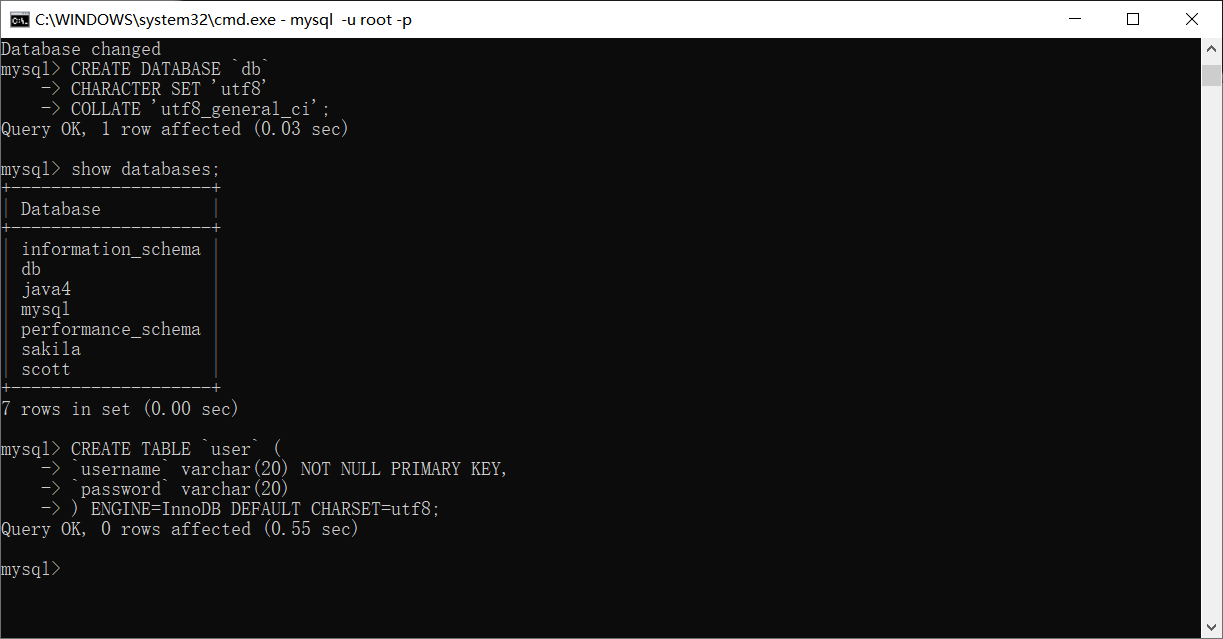
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 主键 | 允许空 | 默认值 | 说明 |
| 1 | username | varchar | 40 | 是 | 否 |  | 用户名 |
| 2 | password | varchar | 40 |  | 是 |  | 口令 |

创建过程如下：(可选择使用MySQL的数据库管理和开发工具Navicat 和 MySQL-Front等)

Step1:进入MySQL程序的MySQL Command Line Client界面，如下图所示



Step2:分别执行下列命令：

1. 创建数据库：
2. 建表:

2.开发过程如下：

Step1: 创建值JavaBean: User.java,内容如下：

package valueBean;

public class VOUser {

private String username;//用户名

private String password;//口令

public VOUser()

{

this.username="";

this.password="";

}

public void setUsername(String username) {

this.username = username;

}

public String getUsername() {

return username;

}

public void setPassword(String password) {

this.password = password;

}

public String getPassword() {

return password;

}

}

Step2: 创建连接数据库类DBUtil

package toolBean.db;

import java.sql.\*;

import valueBean.User;

public class DBUtil{

private static Connection conn=null;

public static Connection getConnection()

{

String driverClass="com.mysql.jdbc.Driver";

String url=//与创建数据库的代码结合，保证访问数据库时不出现乱码

"jdbc:mysql://localhost:3306/db?useUnicode=true&characterEncoding=UTF-8";

String username = "root";

String password = "123";

try {

Class.forName(driverClass);// 加载数据库驱动

} catch (ClassNotFoundException e) {

e.printStackTrace();

}

if (conn==null)

try {

conn = DriverManager.getConnection(url, username, password);//建立连接

} catch (SQLException e) {

e.printStackTrace();

System.out.print("数据库连接失败！");

}

return conn;

}

Step3: 创建访问数据库的类—DAOUser.java,内容如下：

package toolBean.db;

import java.sql.\*;

import valueBean.User;

public class UserDAO {

private static Connection conn=null;

private Statement stmt=null;

//构造函数中完成对数据库进行初始化

public UserDAO() {

//创建连接

conn=DBUtil.getConnection();

try {

//创建Statement

stmt=conn.createStatement();

} catch (SQLException e) {

e.printStackTrace();

}

}

//将对象user添加到表中

public void Add(VOUser user) throws SQLException

{

String sql="INSERT INTO user(username,password) VALUES('"

+user.getUsername()+"','"+user.getPassword()+"')";

stmt.executeUpdate(sql);

//此处可以使用预定义语句PreparedStatement

}

//根据用户名（关键字）从数据库删除相应的记录

public void Delete(String username) throws SQLException

{

String sql="Delete From user Where username='"+username+"'";

stmt.executeUpdate(sql);

}

//修改数据库中的user

public void Update(User user) throws SQLException

{

String sql="UPDATE user set password='"

+user.getPassword()+"' where username='"+user.getUsername()+"'";

stmt.executeUpdate(sql);

}

//根据用户名称（关键字）从数据库中查找记录，并将找到的记录写入对象user中返回

public User searchByUsername(String username) throws SQLException

{

User user=new User();

String sql="select \* from user where username='"+username+"'";

ResultSet rs=stmt.executeQuery(sql);

while(rs.next())

{

String name=rs.getString("username");

if(name.equals(username))

{

//将从数据库查找得到的记录存入对象user中

user.setUsername(name);

user.setPassword(rs.getString("password"));

if(rs!=null)

rs.close();

return user;

}

}

return null;//查找不到返回null

}

//根据用户判断该记录是否存在

public boolean exists(String username)

{

boolean rtn=false;

try {

if(searchByUsername(username)!=null)

rtn=true;

} catch (SQLException e) {

e.printStackTrace();

}

return rtn;

}

//关闭与访问数据库有关连接

public void finalize()

{

try {

if(stmt!=null)

stmt.close();

if(conn!=null)

conn.close();

}

catch (SQLException e) {

e.printStackTrace();

}

}

}

Step4: 创建JSP文件以及相应的Servlet

创建 添加用户页面—AddUser.jsp,代码如下：

<%@ page contentType="text/html; charset=gb2312" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>添加用户</title>

</head>

<body>

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

<div align="center"><font face="宋体" size="6"><strong>

添加用户</strong></font><br/><hr/>

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

<br/>

<br/>

密&nbsp;&nbsp;码：

<input name="password" type="text"/>

<br />

<br />

<input type="submit" value="添加" />

<input type="reset" value="取消" />

</div>

</form>

</body>

</html>

相应的添加用户Servlet(AddUserServlet.java),代码如下：

package servlet.user;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import toolBean.db.AccessUserFromDB;

import valueBean.User;

public class AddUserServlet extends HttpServlet {

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

//返回上页

String goBack="<br><a href='javascript:window.history.go(-1);'>返回上页</a>";

response.setContentType("text/html;charset=gb2312");

PrintWriter out = response.getWriter();

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

if(username==null||username.equals(""))

{

out.print("用户名不能为空！");

out.print(goBack);

}

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

User user=new User();

user.setUsername(username);

user.setPassword(password); //注入属性

UserDAO userDB=new UserDAO();

if(userDB.exists(user.getUsername()))

{

out.print("用户名已存在！");

out.print(goBack);

return;

}

try {

userDB.Add(user);

} catch (SQLException e) {

e.printStackTrace();

out.print("添加失败:"+e.getLocalizedMessage());

out.print(goBack);

return;

}

out.print("添加成功！");

out.print(goBack);

out.close();

}

}

创建删除用户页面--Delete.jsp,代码如下：

<%@ page contentType="text/html; charset=gb2312" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>删除用户</title>

</head>

<body>

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

<div align="center"><font face="宋体" size="6"><strong>

删除用户</strong></font><br/>

<hr/>

<p>

</p>

<p>输入删除的用户名 <input name="username" type="text"/><br/><br/>

<input type="submit" name="delete" value="删除用户" />

<br />

<br />

</p>

</div>

</form>

</body>

</html>

相应的删除用户Servlet (DeleteUserServlet.java)的代码，如下：

package servlet.user;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import toolBean.db.AccessUserFromDB;

import valueBean.User;

public class DeleteUserServlet extends HttpServlet {

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

//返回上页

String goBack="<br><a href='javascript:window.history.go(-1);'>返回上页</a>";

response.setContentType("text/html;charset=gb2312");

PrintWriter out = response.getWriter();

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

if(username==null||username.equals(""))

{

out.print("用户名不能为空！");

out.print(goBack);

}

AccessUserFromDB userDB=new AccessUserFromDB();

if(!userDB.exists(username))

{

out.print("用户名不存在！");

out.print(goBack);

return;

}

try {

userDB.Delete(username);

} catch (SQLException e) {

e.printStackTrace();

out.print("删除失败:"+e.getLocalizedMessage());

out.print(goBack);

return;

}

out.print("删除成功！");

out.print(goBack);

out.close();

}

}

创建查找和修改页面(SearchAndModify.jsp),代码如下：

<%@ page contentType="text/html; charset=gb2312" language="java" import="java.sql.\*" errorPage="" %>

<%@page import="toolBean.db.AccessUserFromDB"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<%

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

valueBean.User user=new valueBean.User();

if(username!=null&&username!="")

{

AccessUserFromDB dbUser=new AccessUserFromDB();

valueBean.User temp=dbUser.searchByUsername(username);

if(temp==null)

{

out.print("用户名不存在！");

}

else

user=temp;//防止user为null，导致下文的user.getUsername()产生异常

}

%>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

<title>查询和修改用户</title>

</head>

<body>

<div align="center">

<form name="query" method="post" action="">

<font face="宋体" size="6"><strong>查询和修改用户</strong></font><br/><hr/><br/>

输入查询的用户名： <input name="queryName" type="text"/> <input type="submit" name="query" value="查询用户" /><br/><br/><hr/><br/>

</form>

<form name="modify" method="post" action="ModifyUser">

用户名：

<input name="modfyName" type="text" value="<%=user.getUsername()%>" readonly="readonly" />

<br/><br/>

密&nbsp;&nbsp;码：

<input name="password" type="text" value="<%=user.getPassword()%>"/>

<br /><br />

<input type="submit" name="modify" value="修改用户" />

</form>

<hr/>

<a href="DeleteUser.jsp">删除用户</a><br/>

<a href="AddUser.jsp">添加用户</a><br/>

</div>

</body>

</html>

修改用户的Servlet (ModifyUser.java),代码如下：

package servlet.user;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import valueBean.User;

public class ModifyUserServlet extends HttpServlet {

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

//返回上页

String goBack="<br><a href='javascript:window.history.go(-1);'>返回上页</a>";

response.setContentType("text/html;charset=gb2312");

PrintWriter out = response.getWriter();

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

if(name!=null) //单击了‘修改用户’

{

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

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

User user=new User();

user.setUsername(username);

user.setPassword(password);

toolBean.db.AccessUserFromDB dbUser=new toolBean.db.AccessUserFromDB();

try {

dbUser.Update(user);

} catch (SQLException e) {

e.printStackTrace();

out.print("修改失败："+e.getLocalizedMessage());

out.print(goBack);

return;

}

}

out.print("修改成功！");

out.print(goBack);

out.close();

}

}