Jsp课程设计报告

课程： jsp程序设计

学号： 2016531121

姓名：强尊慧

班级：

教师：

学生选课管理 系统

一．系统需求分析

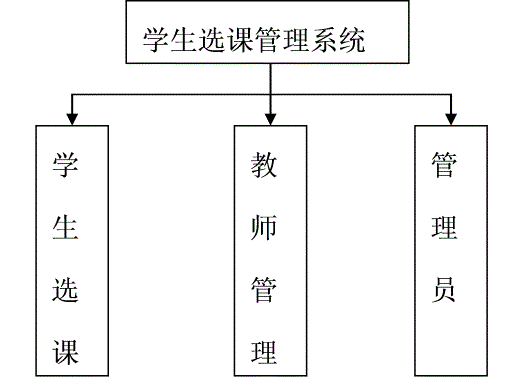
学生选课系统是一个学校不可缺少的部分 , 它的内容对于学校的决策者和管理者来说都至关重要 , 所以学生选课系统应该能够为用户提供充足的信息和快捷的查询手段。随着科学技术的不断提高 , 计算机科学日渐成熟 , 其强大的功能已为人们深刻认识 , 它已进入人类社会的各个领域并发挥着越来越重要的作用。  
 作为计算机应用的一部分 , 使用计算机对选课信息进行管理 , 有着手工管理所无法比拟的优点。这些优点能够极大地提高人事劳资管理的效率 , 也是企业的科学化、正规化管理 , 与世界接轨的重要条件。  
 因此，开发这样一套软件成为很有必要的事情 , 在下面的各章中我们将以开发一套学生选课系统为例 , 谈谈其开发过程和所涉及到的问题及解决方法。  
 本实验实现的学生选课管理系统主要包括以下功能：

1 ．完成数据的录入和修改，并提交数据库保存。其中的数据包括：学生信息、课程设置、学生成绩以及操作员信息等；  
 2. 设计实现学生信息查询，选课管理、退选管理、分数查询，课程的浏览教师信息查询、教室空闲时间查询等 JSP 页面 。  
 3 ．具备一定的安全性管理功能。包括操作员的权限设置、修改密码设置、注销和等功能。

二．系统总概括

1. 系统的总设计

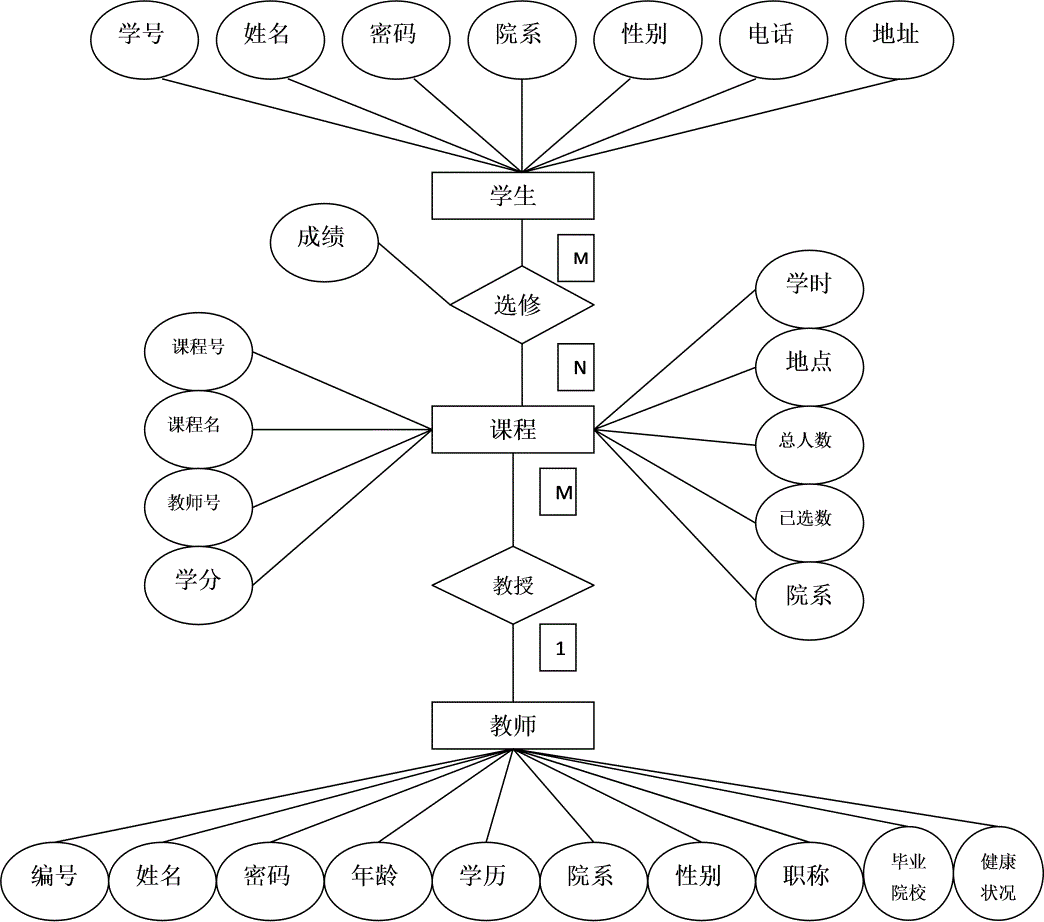
本系统结合数据库和 JSP 编程实现了学生选课管理系统。学生选课管理系统分为三个子模块：第一模块是学生选课的页面操作，包括个人信息，密码修改，查询成绩，选课，退选五个功能；第二模块是教师操作页面，包含个人信息，密码修改，录入成绩，查询选课情况，查看教室信息；第三模块是管理员，包含学生信息管理（增加、修改、删除、查询），教师信息管理（增加、修改、删除、查询），管理员信息管理（增加、修改、删除、查询）。

图 1 系统的功能模块图

2 ．数据库设计

2.1 数据库概念结构设计

本系统主要有三个实体，其中有学生和课程两个实体集，一个学生可以选修若干门课程，一门课程也可以被多个学生选修，这两个实体集之间是多对多联系，还有教师和课程两个实体，一个老 师可以 教授若干门课程，一门课程却只能有一个老 师 教授，这两个实体集之间是一对多联系。通过选课 成绩将 老师和学生联系在一起。为保证系统的安全性，设置了操作员这一实体集，用来存放合法用户的编号、姓名、密码。本系统的 E-R 图如下：



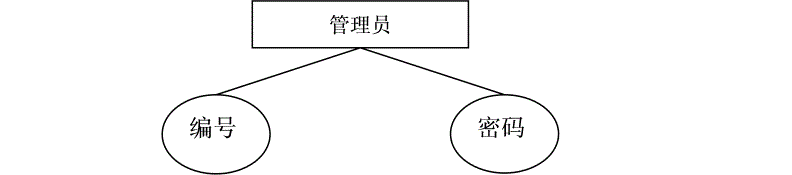


图 2 系统的 E-R 图

2.2 数据库逻辑结构设计

根据关系模型的转换原则，上面的 E—R 图可转换为如下所示的关系模型：

student( sId , sName,sPass , dId)

含义分别为：学生（ 学号 , 姓名，密码，所在系的编号）

stuDetail （ userID ， userSex ， userMobil ， useraddress ）

含义分别为：学生信息（ 学号 , 性别，电话，地址）

teacher （ tId ， tPame ， tPass ， dId ）

含义分别为：教师（ 学号 , 姓名，密码，所在系的编号）

teaDetail （ userID ， userSex ， userAge ， useredu ， usertitle ， usersch ， userhealth ）

含义分别为：教师信息（ 学号 , 性别，年龄，学历，职称，毕业院校，健康状况）

course( cId , cName,tId , credit ， period ， cPlace ， cNumall,cNum,dId)

含义分别为：课程（ 课程号 , 名称 , 教师号 , 学分，学时，上课地点，总人数，已选人数，院系号）

c hoosecourse( cId ， sId , Grade)

含 义分别为：选修（ 学号 , 课程代码 , 成绩）

room （ roomname ， roomnum ， roomtime ）

含义分别为： 教室 （ 教室名 ，容纳人数，空闲时间 ）

admin ( aI d , aPass )

含义分别为： 管理员 （ 管理 员编号 ， 姓名）

d epartment （ dId ， dName ）

含义分别为： 院系 （ 院系 编号 ，院系 名）

根据上面的 E-R 图设计及关系模型的设计，得出数据表的设计，在本系统中主要的数据表如下：

表 1 学生表 Student 的表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 小数位 | 空否 | 备 注 |
| Pk | sId | Char | 10 |  | N | 学号 |
|  | sName | Char | 20 |  | Y | 姓名 |
|  | sPass | Char | 2 |  | Y | 密码 |
|  | d Id | Char | 20 |  | Y | 所在系号 |
|  |  |  |  |  |  |  |

表 2 学生信息表 StuDetail 的表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 小数位 | 空否 | 备 注 |
| Pk | userID | Char | 10 |  | N | 学号 |
|  | userSex | Char | 10 |  | Y | 性别 |
|  | userMobil | Char | 30 |  | Y | 电话 |
|  | userAddre ss | Char | 5 0 |  | Y | 地址 |
|  |  |  |  |  |  |  |

表 3 教师 表 teacher 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | t Id | Char | 10 | N | 编 号 |
|  | t Name | Char | 20 | Y | 姓名 |
|  | t Pass | Char | 2 | Y | 密码 |
|  | d Id | Char | 20 | Y | 所在系号 |
|  |  |  |  |  |  |

表 4 教师 信息表 teadetail 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | userID | Char | 10 | N | 学号 |
|  | userSex | Char | 10 | Y | 性别 |
|  | user Age | Char | 5 0 | Y | 电话 |
|  | user edu | Char | 3 0 | Y | 学历 |
|  | usertitle | Char | 3 0 | Y | 职称 |
|  | usersch | Char | 3 0 | Y | 毕业院校 |
|  | Userhealth | Char | 50 | Y | 健康状况 |
|  |  |  |  |  |  |

表 5 课程信息表 Course 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | cId | Char | 10 | N | 课程 号 |
|  | coName | Char | 20 | Y | 课程名称 |
|  | tId | Char | 10 | Y | 教师 号 |
|  | credit | Char | 8 | Y | 学分 |
|  | period | Char | 8 | Y | 学时 |
|  | cPlace | Char | 10 | Y | 上课地点 |
|  | cNumAll | Char | 4 | Y | 总人数 |
|  | cNum | Char | 4 | Y | 已选人数 |
|  | dId | Char | 10 | Y | 院系号 |
|  |  |  |  |  |  |

表 6 选课成绩表 choosecourse 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | cId | Char | 5 | N | 课程代码 |
| sId | Char | 3 | N | 学号 |  |
|  | g rade | Decimal | 5 | Y | 成绩 |
|  |  |  |  |  |  |

表 7 教室信息 表 room 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | roomname | Char | 10 | N | 教室名 |
| roomnum | Char | 10 | N | 容纳人数 |  |
|  | roomtime | Char | 10 | Y | 空闲时间 |
|  |  |  |  |  |  |

表 8 管理 员表 admin 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | aId | Char | 10 | N | 管理员 编号 |
|  | aPass | Char | 10 | Y | 管理员 姓名 |
|  |  |  |  |  |  |

表 9 院系 表 department 的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 主码 | 列 名 | 数据类型 | 宽度 | 空否 | 备 注 |
| Pk | dId | Char | 10 | N | 院系 号 |
|  | dName | Char | 10 | Y | 院系 名 |
|  |  |  |  |  |  |

下面在系统实施阶段就可以物理实现数据库了，本系统使用前面实验已经建好的服务器端 SQL Server 数据库 — xuanke ，具体创建方法见前面实验。

三．系统实现

本实验是用 JSP 结合数据库实现的，其中包含了三个主页面，一是学生登录页面，二是教师登录页面，三是管理员登录页面。其中每个页面中都会有菜单进行相应的操作。

3.1 JSP 文件代表的操作

(1) 主界面操作 :

Index.jsp// 登录界面 （包含学生、教师、管理员三个选项）

Login.jsp// 密码验证

(2) 学生选课

(a) 学生选课界面 ： student .jps// 菜单显示

studentIndex .html// 欢迎界面

studentMessage .jsp

(b) 个人信息： studMessSel .jsp// 个人信息的页面显示

(c) 密码修改： stuM .jsp// 在数据库中修改密码

stuMessMod .jsp// 修改密码界面

stuMessModSuc .jsp// 修改成功

(d) 成绩查询： gradeselect.jsp// 相应课程的成绩查询

(e) 选课： choosecourse.jsp// 所有课程显示

choose.jsp// 在数据库中插入数据

choosesuccess.jsp// 选课成功

(f) 选课情况查询及退选： courseselect.jsp// 选课情况查询

ccdel.jsp// 数据库中删除选课记录

(g) 注销 : zhuxiao.jsp// 注销

( 3 ) 教师管理

(a) 教师管理 界面 ： teacher.jps// 菜单显示

teacherI ndex .html// 欢迎界面

teacher Message .jsp

(b) 个人信息： teacher MessSel .jsp// 个人信息的页面显示

(c) 密码修改： teacher M .jsp// 在数据库中修改密码

teacher Mod .jsp// 修改密码界面

tea MessModSuc .jsp// 修改成功

(d) 提交成绩： grade.jsp// 选课学生信息显示

gradeIns.jsp// 在数据库中插入数据

(e) 课程信息： lookcourse.jsp// 选课情况查询

(f) 教室信息： classroom.jsp// 查看可用教室的空闲时间

(g) 注销 : zhuxiao.jsp// 注销

3.2 数据库的连接

举例：验证用户类型和密码时的数据库连接

<%

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

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

String s=(String)request.getParameter("userselect");

Connection conn=null;

Statement stmt=null;

ResultSet rs=null;

String loginNum="0";

session.setAttribute("lnum",loginNum);

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

if(s.equals("1"))

{

String sql="select \* from student where sId='"+name+"' and sPass='"+pass+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

loginNum="1";

session.setAttribute("lnum",loginNum);

session.setAttribute("name",name);

out.print(name+" 登陆成功 , 页面跳转 ........");

response.sendRedirect("student/studentIndex.html");

}

else

{

out.print(" 用户名或密码有误 ..");

response.sendRedirect("index.jsp");

}

}

if(s.equals("2"))

{

String sql="select \* from teacher where tId='"+name+"' and tPass='"+pass+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

loginNum="2";

session.setAttribute("lnum",loginNum);

session.setAttribute("name",name);

out.print(name+" 登陆成功 , 页面跳转 ........");

response.sendRedirect("teacher/teacherIndex.html");

}

else

{

out.print(" 用户名或密码有误 ..");

response.sendRedirect("index.jsp");

}

}

if(s.equals("3"))

{

String sql="select \* from admin where aId='"+name+"' and aPass='"+pass+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

loginNum="3";

session.setAttribute("lnum",loginNum);

session.setAttribute("name",name);

out.print(" 管理员 "+name+" 登陆成功 , 页面跳转 ........");

response.sendRedirect("admin/adminIndex.html");

}

else

{

out.print(" 用户名或密码有误 ..");

out.print(" 登陆页面 ");

}

}

}

catch(Exception e){out.print(e);

//response.sendRedirect("index.jsp");

}

%>

3.3 具体操作

(1) 学生选课操作

(a) 登录页面



图 1.1 学生登录页面

主要代码：

欢迎您使用学生选课系统

用户名：

密码：

用户类型：

学生

教师

管理员

(b) 菜单操作

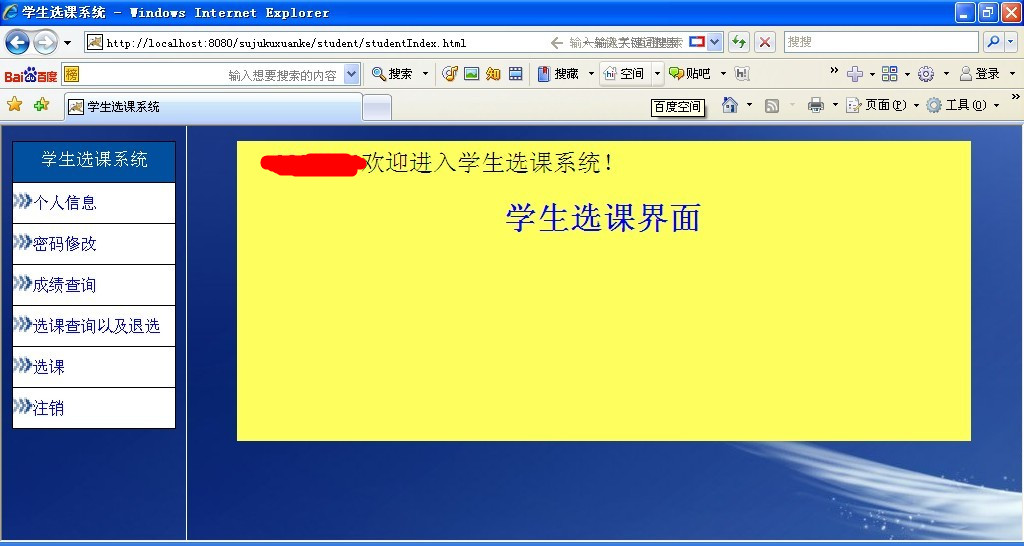


图 1.2 学生选课系统的界面

主要代码：

学生选课系统

[个人信息](studMessSel.jsp)

[密码修改](stuMessMod.jsp)

[成绩查询](gradeSelect.jsp)

[选课查询以及退选](courseSelect.jsp)

[选课](choosecourse.jsp)

[注销](zhuxiao.jsp)

(c) 个人信息显示

学生进入后可以在菜单里进行相应的操作 , 点击个人信息时 , 就会出现该学生的信息 . 如下图所示 :



图 1.2 学生信息显示的界面

主要代码：

从数据库中读出相应的信息：

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql="select \* from student where student.sId='"+name+"'";

System.out.println(sql);

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

sId=rs.getString("sId");

sName=rs.getString("sName");

dId=rs.getString("dId");

}

String sql1="select \* from stuDetail where userID ='"+name+"'";

System.out.println(sql1);

stmt=conn.createStatement();

rs=stmt.executeQuery(sql1);

if(rs.next())

{

Sex=rs.getString("userSex");

Mobil=rs.getString("userMobil");

Addr=rs.getString("userAddress");

}

Connection connd=null;

Statement stmtd=null;

ResultSet rsd=null;

connd=DriverManager.getConnection("jdbc:odbc:xk","sa","");

stmtd=connd.createStatement();

rsd=stmtd.executeQuery("select dName from department where dId='"+dId+"'");

if(rsd.next())

{

dName=rsd.getString("dName");

}

}

catch(Exception e){out.println(e);

}

%>

**学生个人信息**

size="4" color="#FFFFFF"> 学号

size="3" color="#FFFFFF"><%=sId%>

size="4" color="#FFFFFF"> 姓名

size="3" color="#FFFFFF"><%=sName%>

size="4" color="#FFFFFF"> 性别

size="3" color="#FFFFFF"><%=Sex%>

size="4" color="#FFFFFF"> 院系

size="3" color="#FFFFFF"><%=dName%>

size="4" color="#FFFFFF"> 电话

size="3" color="#FFFFFF"><%=Mobil%>

size="4" color="#FFFFFF"> 地址

size="3" color="#FFFFFF"><%=Addr%>

(d) 密码修改

举例 :

将 0826006 学号的密码该为： 6666

其显示如下：

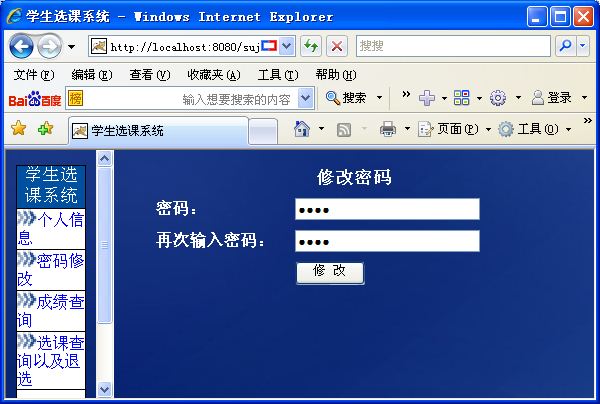


图 1.3 修改密码

修改提交后数据库中的结果如下，可见密码已修改

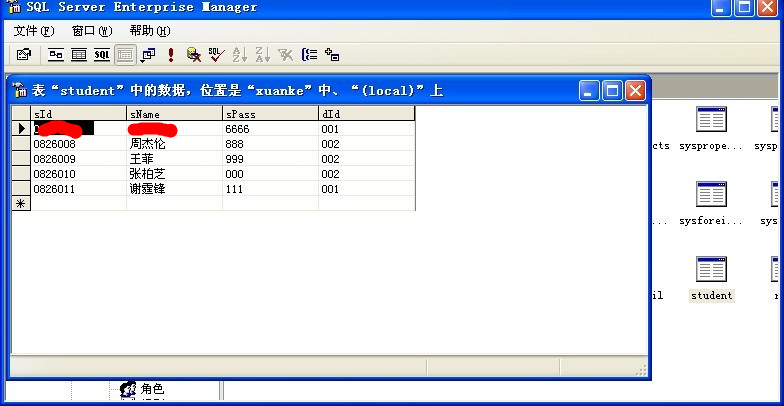


图 1. 4 数据库中 修改 后的 student 表

主要代码：

数据 库 中修改该学生的密码 的代码

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

Connection conn=null;

Statement stmt=null;

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql2="update student set sPass='"+sPass+"' where sId='"+name+"'";

stmt=conn.createStatement();

stmt.executeUpdate(sql2);stmt.close();

response.sendRedirect("stuMessModSuc.jsp");

}

function checkBig()

{

if (document.form1.sPass.value=="")

{

alert(" 密码不能为空！ ");

document.form1.sPass.focus();

return false;

}

else if (document.form1.sPassagn.value=="")

{

alert(" 再次输入密码！ ");

document.form1.sPassagn.focus();

return false;

}

else if (document.form1.sPass.value!=document.form1.sPassagn.value)

{

alert(" 两次密码不同！ ");

document.form1.sPass.focus();

return false;

}

}

out.print(" 修改成功 ");

(e) 成绩查询

根据教授该门课的老师是否提交成绩，学生可以查到自己相应课程的成绩 ，其查询结果如下：

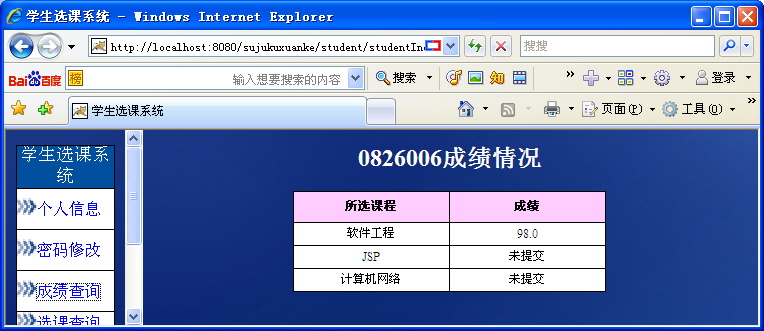


图 1.5 查询成绩的页面

主要代码：

try

{

Statement stmtc=null;

ResultSet rsc=null;

Connection connc=null;

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

connc=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlc="select \* from chooseCourse where sId='"+name+"'";

stmtc=connc.createStatement();

rsc=stmtc.executeQuery(sqlc);

while(rsc.next())

{

String coId=rsc.getString("coId");

float grade=rsc.getFloat("grade");

String grademess=null;

Statement stmt=null;

ResultSet rs=null;

Connection conn=null;

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql="select coName from course where coId='"+coId+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{ if(grade>0)out.print(grade);

else out.print(" 未提交 "); }

}

(f) 课程查询及退选

可以通过对应课程后面的退选选项进行课程的退选

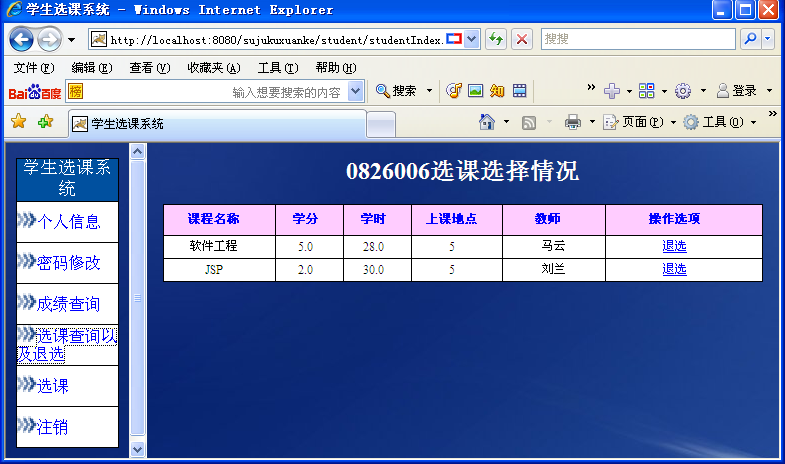


图 1.6 查询选课情况的页面

主要代码：

<%=name%> 选课选择情况

**课程名称**

**学分**

**学时**

**上课地点**

**教师**

bgcolor="#FFCCFF"> **操作选项**

while(rsc.next())

{

String coId=rsc.getString("coId");

float grade=rsc.getFloat("grade");

String grademess=null;

Statement stmt=null;

ResultSet rs=null;

Connection conn=null;

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql="select \* from course where coId='"+coId+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

String coName=rs.getString(2);

String tId=rs.getString(3);

float credit=rs.getFloat(4);

float period=rs.getFloat(5);

String cPlace=rs.getString(7);

Connection connt=null;

Statement stmtt=null;

ResultSet rst=null; connt=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlt="select tName from teacher where tId='"+tId+"'";

stmtt=connt.createStatement();

rst=stmtt.executeQuery(sqlt);

if(rst.next())out.print(rst.getString(1));

}

}

rsc.close();stmtc.close();

%>

(f) 选课

界面中已通过数据库将所有的课程列出来，学生可以通过前面的选择标志进行选课。



图 1.7 选课信息的页面及选课操作

选课后，相应数据库中的 choosecourse 表增加了一行选课记录

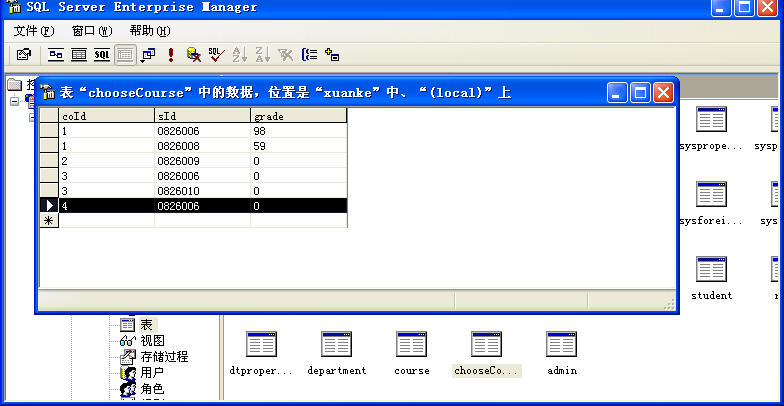


图 1.8 选课后数据库中的 chooseCourse 表

主要代码：

**选择**

**课程名称**

**学分**

**学时**

**上课地点**

**教师**

**剩余可选人数**

<%

while(rs.next())

{

String coId=rs.getString(1);

String coName=rs.getString(2);

String tId=rs.getString(3);

float credit=rs.getFloat(4);

float period=rs.getFloat(5);

String cPlace=rs.getString(6);

int cNumAll=rs.getInt(7);

int cNum=rs.getInt(8);

int num=cNumAll-cNum;

%>

Connection connc=null;

Statement stmtc=null;

ResultSet rsc=null;

connc=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlc="select \* from chooseCourse where sId='"+name+"'";

stmtc=connc.createStatement();

rsc=stmtc.executeQuery(sqlc);

if(rsc.next())out.print(" ");

else if(num<=0)out.print("disabled");

%>>

<%=coName%>

<%=credit%>

<%=period%>

<%=cPlace%>

<%

Connection connt=null;

Statement stmtt=null;

ResultSet rst=null;

connt=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlt="select tName from teacher where tId='"+tId+"'";

stmtt=connt.createStatement();

rst=stmtt.executeQuery(sqlt);

if(rst.next())out.print(rst.getString(1));

%>

<%if(num>0)out.print(num);

else out.print(" 已选满 ");%>

<%

}

%>

(2) 教师操作

(a) 登录页面

输入教师的教师编号和密码，选择用户类型为教师，即可登录

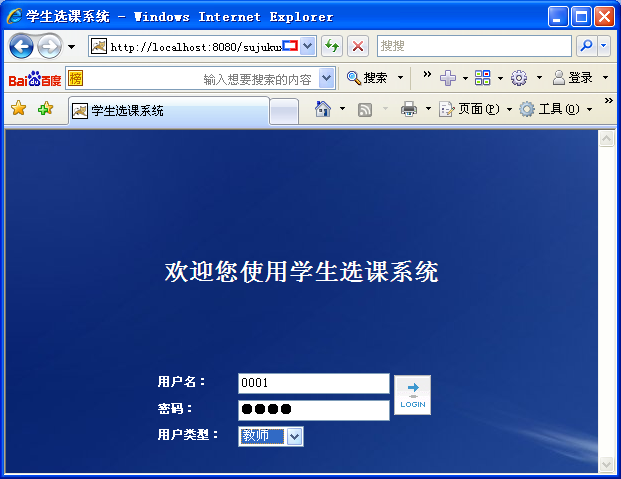


图 2.1 教师登录页面

主要代码：

欢迎您使用学生选课系统

用户名：

height="40" onClick="loginclick()">

密码：

用户类型：

学生

教师

管理员

(b) 菜单操作

登录之后进入教师操作页面，如下图所示，左边是菜单，可以进行相应的操作

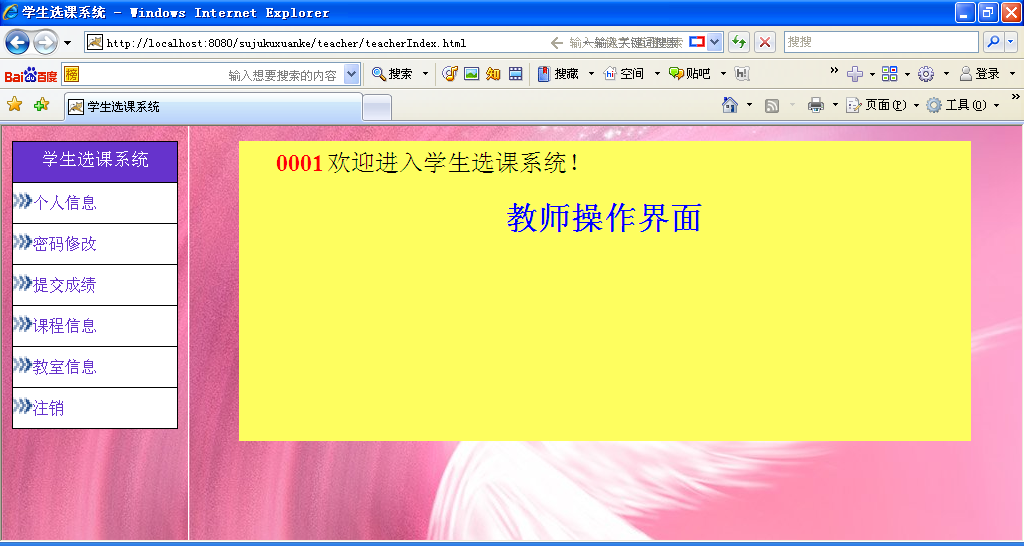


图 2.2 教师操作页面

主要代码：

学生选课系统

[个人信息](teacherMessSel.jsp)

[密码修改](teacherMessMod.jsp)

[提交成绩](grade.jsp)

[课程信息](lookcourse.jsp)

[教室信息](classroom.jsp)

[注销](zhuxiao.jsp)

<%}

else

{

out.print(" 请先登陆 .....");out.print(" 登陆页面 ");

}

%>

(c) 个人信息

点击个人信息的菜单，会在右面的框里显示教师的个人信息 ，教师信息页面显示如下图：



图 2. 3 教师 信息显示

主要代码：

Connection conn=null;

Statement stmt=null;

ResultSet rs=null;

String tId=null;

String tName=null;

String dId=null;

String dName=null;

String sex=null;

String age=null;

String edu=null;

String title=null;

String school=null;

String health=null;

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql="select \* from teacher where teacher.tId='"+name+"'";

System.out.println(sql);

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

tId=rs.getString("tId");

tName=rs.getString("tName");

dId=rs.getString("dId");

}

String sql1="select \* from teaDetail where teaDetail.userID ='"+name+"'";

System.out.println(sql);

stmt=conn.createStatement();

rs=stmt.executeQuery(sql1);

if(rs.next())

{

sex=rs.getString("usersex");

age=rs.getString("userage");

edu=rs.getString("useredu");

title=rs.getString("usertitle");

school=rs.getString("usersch");

health=rs.getString("userhealth");

}

Connection connd=null;

Statement stmtd=null;

ResultSet rsd=null;

connd=DriverManager.getConnection("jdbc:odbc:xk","sa","");

stmtd=connd.createStatement();

rsd=stmtd.executeQuery("select dName from department where dId='"+dId+"'");

if(rsd.next())

{

dName=rsd.getString(1);

}

}

catch(Exception e){out.println(e);}

(c) 密码修改

将原来的密码 111 改成了 11111 1

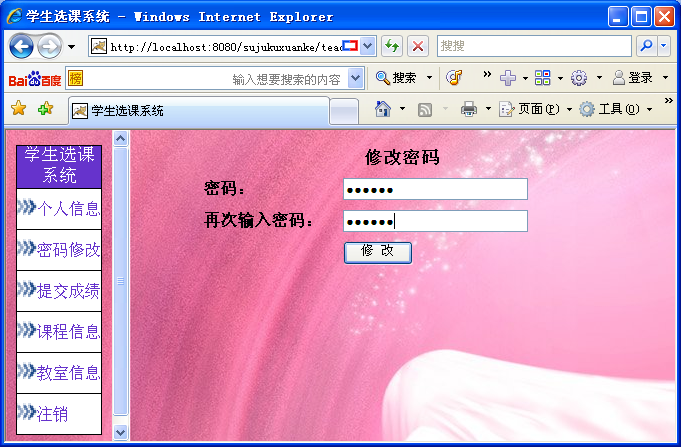


图 2. 4 教师 密码修改界面

修改后，可以查询数据库中的 teacher 表，可以看到马云教师的密码被修改成了 111111 。

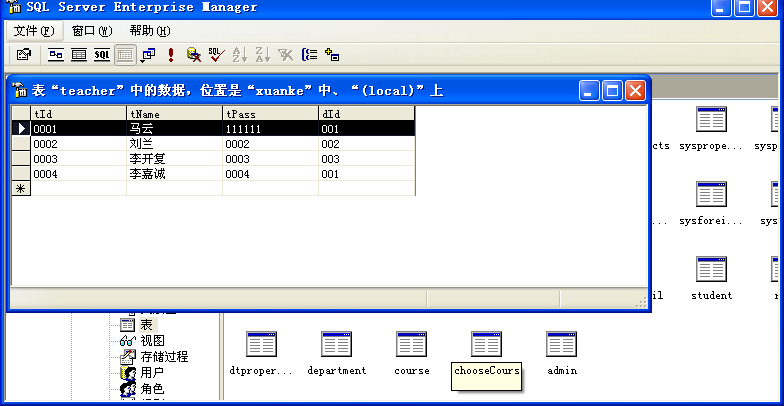


图 2. 5 修改密码后的 teacher 表

主要代码：

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

Connection conn=null;

Statement stmt=null;

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql="update teacher set tPass='"+tPass+"' where tId='"+name+"'";

stmt=conn.createStatement();

stmt.executeUpdate(sql);stmt.close();

response.sendRedirect("teaMessModSuc.jsp");

}

**size="4" color="#000000"> 修改密码**

**密码：**

**再次输入密码：**

(d) 成绩录入

点击菜单中的提交成绩，可显示如下页面 ：

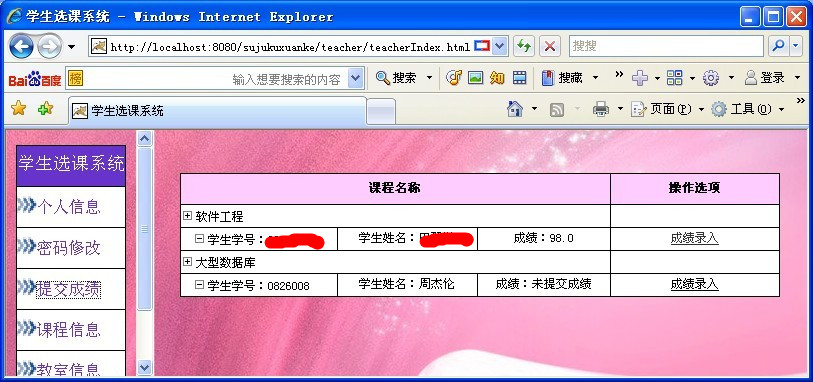


图 2. 6 学生成绩录入界面

点击相应的课程下的学生成绩录入，可进入下个页面 ：

学号为 0826006 的学生录入成绩为 59 分。



图 2. 7 成绩录入

成绩录入后可在数据中查询到相应的信息，学生也可通过学生选课菜单中的查询成绩也可看到相应的成绩 ，下图为数据库中的 choosecourse 中的信息

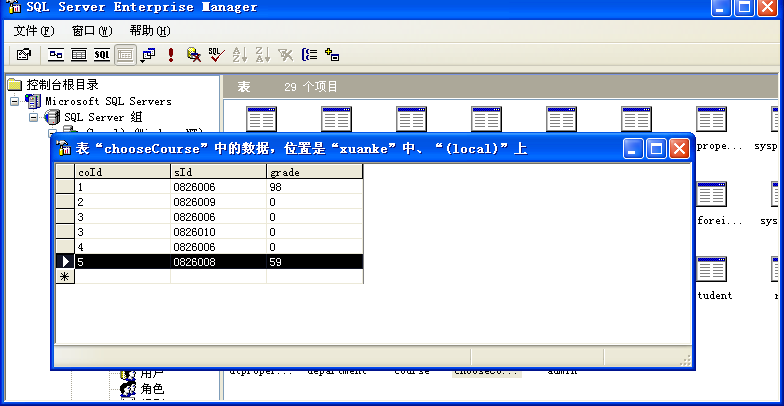


图 2. 8 录入成绩后数据库中的 choosecourse 表

主要代码：

Connection conn=null;

Statement stmt=null;

ResultSet rs=null;

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sql="select course.coId,coName,sId,grade from course,chooseCourse where course.coId=chooseCourse.coId and tId='"+name+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

while(rs.next())

{

String coId=rs.getString(1);

String coName=rs.getString(2);

String sId=rs.getString(3);

float grade=rs.getFloat(4);

Statement stmtc=null;

ResultSet rsc=null;

Connection connc=null;

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

connc=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlc="select sName from student Where sId='"+sId+"'";

stmtc=connc.createStatement();

rsc=stmtc.executeQuery(sqlc);

while(rsc.next())

{

String sName=rsc.getString(1);

**录入成绩**

**学生学号：**

" readonly="false"/>

"/>

**成绩：**

(e) 课程信息

在这里可以 查询到 老师所教的所有课程的具体信息，包括 学分，学时，上课地点， 已选的人数等 ，如下图所示。



图 2.9 教师的课程选课情况

主要代码：

Statement stmtc=null;

ResultSet rsc=null;

Connection connc=null;

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver"); connc=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlc="select distinct chooseCourse.coId from chooseCourse,course where chooseCourse.coId=course.coId and tId='"+name+"'";

stmtc=connc.createStatement();

rsc=stmtc.executeQuery(sqlc);

while(rsc.next())

{

String coId=rsc.getString("coId");

String grademess=null;

Statement stmt=null;

ResultSet rs=null;

Connection conn=null;

conn=DriverManager.getConnection("jdbc:odbc:xk","sa","")

String sql="select \* from course where coId='"+coId+"'";

stmt=conn.createStatement();

rs=stmt.executeQuery(sql);

if(rs.next())

{

String coName=rs.getString(2);

String tId=rs.getString(3);

float credit=rs.getFloat(4);

float period=rs.getFloat(5);

String cPlace=rs.getString(6);

int cNumAll=rs.getInt(7);

int cNum=rs.getInt(8);

int num=cNumAll-cNum;

if(num>0)out.print(cNum);

else out.print(" 已经选满 ");

}

rsc.close();stmtc.close();

}

(f) 教室信息

教师可以通过查询教室的空闲时间，以备调课使用。



图 2.10 教室空闲时间查询

主要代码：

Statement stmtc=null;

ResultSet rsc=null;

Connection connc=null;

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

connc=DriverManager.getConnection("jdbc:odbc:xk","sa","");

String sqlc="select \* from room";

stmtc=connc.createStatement();

rsc=stmtc.executeQuery(sqlc);

while(rsc.next())

{

String rname=rsc.getString("roomname");

String rnum=rsc.getString("roomnum");

String rtime=rsc.getString("roomtime");

}

rsc.close();stmtc.close();

<%=name%> 教室情况

**教室名称**

**可容纳人数**

**空闲时间**

<%=rname%>

<%=rnum%>

<%=rtime%>

四 . 总结

经过一段时间终于把数据库的课程设计做好了 , 基本实现了学生选课的一般功能。 通过这次课程设计，我对数据库软件 SQL Server 、 JSP 等系列知识都有了一定的了解。使用 JSP 开发页面的能力也有了很大提高。 在本次设计的全过程中，我对所学的知识有了一个比较系统的认识和理解。涉及了各方面的知识，大大扩展了我的知识面。

在整个设计阶段感受最为深刻的是掌握学习的方式和解决问题的方法。首先是解决问题的方式，就像我在解决设计中的很多问题一样，一开始总是在看书，和书上的进行比对，虽然书本上的只是大部分都是有价值，正确的，但实际上每个人编程的思路和对数据处理的方法、思想都是不同的，这就要求我们一定要通过实践才能找到解决问题的方案。在整个课程设计的过程中，我深刻的体会到了这一点的重要性，也牢记主了这一点，在今后的学习中我也会继续将这种方法延续下去。

在做课程设计的这段时间里 , 虽然也碰到了很多的问题 , 但是经过和同学的讨论和研究，问题都一一克服了得到了解决。因为我们始终坚信“只要有恒心，铁杵磨成针 !” 。经过这次课程设计，我的能力有了很大的提高，比如操作能力、分析问题的能力、合作精神、严谨的工作作风等方方面面都进步了。我要更加努力，努力学好专业知识，全面发展自己。相信经过这次的磨练，对我以后的学习肯定大有益处！