实验五 选课管理

设计数据库，基本信息包括学生、课程、教师，学生从已分配教师的课程列表中选择相应的课程（课程列表项信息包括：课程名称、学分、授课教师、授课时间等，可以手工录入已分配教师、上课时间的课程列表信息）。

选课规则：每个学生最少选择3门课程，学分高于8学分，最多可以选择5门课程，学分不超过12学分。

实现下列功能：

本次实验使用工具：

SQL SERVER2012

VISUAL STUDIO 2019

POWER DESIGNER

本次实验使用语言：

C语言

C++语言

C#语言

1.录入学号，自动显示学生姓名。



学生的姓名是事先就已经录入到数据库当中的，因此只要在学号栏里输入相应的学号，就可以通过学号来获得该学生的姓名，如果没有点击左边的姓名查找按钮来查询姓名的话则会跳出窗口来提醒你需要先获取姓名才能进入选课系统。



而当你输入了一个不存在的学号时，系统则会自动判断该学号不存在于数据库内，姓名那一块地方则会显示出NULL值，同时点击开始选课按钮的话会跳出窗口显示登陆失败，一定要输入正确的学号和姓名才能登陆成功。



如上图所示，在输入正确的学号后点击查找按钮，成功显示该学生姓名，然后点击右边的选课按钮，跳出窗口显示成功登录，进入选课系统。

这一块的函数功能代码实现如下：

private void button1\_Click(object sender, EventArgs e)

{

if(textBox1.Text=="300001")

{

label11.Text ="路明非";

}

else if (textBox1.Text == "300002")

{

label11.Text = "楚子航";

}

else if (textBox1.Text == "300003")

{

label11.Text = "陈墨瞳";

}

else if (textBox1.Text == "300004")

{

label11.Text = "夏弥";

}

else if (textBox1.Text == "300005")

{

label11.Text = "上杉绘梨衣";

}

else if (textBox1.Text == "300006")

{

label11.Text = "恺撒·加图索 ";

}

label11.Visible = true;

//sjk.SJKclose();

cnt = 1;

}

private void button2\_Click(object sender, EventArgs e)

{

void login()

{

SJK sjk = new SJK();

string sql = "select \* from 学生 where 学号='" + textBox1.Text + "'";

IDataReader bb = sjk.Read(sql);

if (bb.Read())

{

MessageBox.Show("成功登录！");

zhuye zy = new zhuye();

this.Hide();

zy.ShowDialog();

this.Show();

}

else

{

MessageBox.Show("登录失败！");

}

}

if (textBox1.Text != "" && cnt == 1)

{

cnt = 0;

login();

}

else

{

MessageBox.Show("请输入学号并点击查找学生姓名之后再点击这个按钮");

}

}

2.在可选择课程列表上方显示已选择的课程列表，可以从已选择的课程列表中删除已选择的课程。



进入选课系统后，可以看到在上面的菜单里有课程，这些课程都是可选择的课程，而下面则是已选课程的菜单表，菜单表里有课程编号，课程名，教师编号，教师名，学分，上课时间这六个属性，左边有四个按钮分别是课表，选课，退课和确认，其中课表点击之后会弹出新的窗体显示你提交之后的课程，而确认则是在你满足选课要求之后，会将已选课程里的课程数据全部传输到课表当中。

这一块的函数功能代码实现如下：

private void zhuye\_Load(object sender, EventArgs e)

{

function();

function2();

}

public void function()

{

dataGridView1.Rows.Clear();

SJK sjk = new SJK();

string sql = "select \* from 课程";

IDataReader cc = sjk.Read(sql);

while (cc.Read())

{

dataGridView1.Rows.Add(cc[0].ToString(), cc[1].ToString(), cc[2].ToString(), cc[3].ToString(), cc[4].ToString(), cc[5].ToString());

}

cc.Close();

sjk.SJKclose();

}

public void function2()

{

dataGridView2.Rows.Clear();

SJK sjk = new SJK();

string sql = "select \* from 当前课程";

IDataReader dd = sjk.Read(sql);

while (dd.Read())

{

dataGridView2.Rows.Add(dd[0].ToString(), dd[1].ToString(), dd[2].ToString(), dd[3].ToString(), dd[4].ToString(), dd[5].ToString());

}

dd.Close();

sjk.SJKclose();

}

3.显示可选择的课程列表（不包含已选择的课程），可以按规则选择3-5门课程，选择的课程自动从可选课程列表中移至已选课程列表。



选择可选课程菜单栏当中的一门课程，点击选课按钮，该课程将会移动到已选课程菜单栏，同时将上面的该课程信息删除。选择已选课程菜单栏当中的一门课程，点击退课按钮，该课程将会返回到可选课程菜单栏，将下面已选菜单栏的该课程信息删除。

这一块的函数功能代码实现如下：

private void button1\_Click(object sender, EventArgs e)

{

try

{

string id = dataGridView1.SelectedCells[0].Value.ToString();

string tid = dataGridView1.SelectedCells[1].Value.ToString();

string tname = dataGridView1.SelectedCells[2].Value.ToString();

string cname = dataGridView1.SelectedCells[3].Value.ToString();

string credit = dataGridView1.SelectedCells[4].Value.ToString();

string time = dataGridView1.SelectedCells[5].Value.ToString();

string sql = "insert into 当前课程 (课程编号,教师编号,教师名,课程名,学分,时间) " +

"values ('" + id + "','" + tid + "','" + tname + "','" + cname + "','" + credit + "','" +time + "')";

SJK sjk = new SJK();

sjk.Execute(sql);

sql = "delete from 课程 where 课程编号 ='"+id+"'and 教师编号='"+tid+"'";

sjk.Execute(sql);

int A = int.Parse(credit);

cnt += A;

A = 0;

cnt2+=1;

MessageBox.Show("选课成功,目前总学分是："+cnt);

MessageBox.Show("总共选课数：" + cnt2);

function();

function2();

sjk.SJKclose();

}

catch

{

MessageBox.Show("我是BUG测试");

}

}

private void button2\_Click(object sender, EventArgs e)

{

try

{

string id = dataGridView2.SelectedCells[0].Value.ToString();

string tid = dataGridView2.SelectedCells[1].Value.ToString();

string tname = dataGridView2.SelectedCells[2].Value.ToString();

string cname = dataGridView2.SelectedCells[3].Value.ToString();

string credit = dataGridView2.SelectedCells[4].Value.ToString();

string time = dataGridView2.SelectedCells[5].Value.ToString();

string sql = "insert into 课程 (课程编号,教师编号,教师名,课程名,学分,时间) " +

"values ('" + id + "','" + tid + "','" + tname + "','" + cname + "','" + credit + "','" + time + "')";

SJK sjk = new SJK();

sjk.Execute(sql);

sql = "delete from 当前课程 where 课程编号 ='" + id + "'and 教师编号='" + tid + "'";

sjk.Execute(sql);

int A = int.Parse(credit);

cnt -= A;

A = 0;

cnt2 -= 1;

MessageBox.Show("退课成功,目前总学分是：" + cnt);

MessageBox.Show("总共选课数：" + cnt2);

function();

function2();

sjk.SJKclose();

}

catch

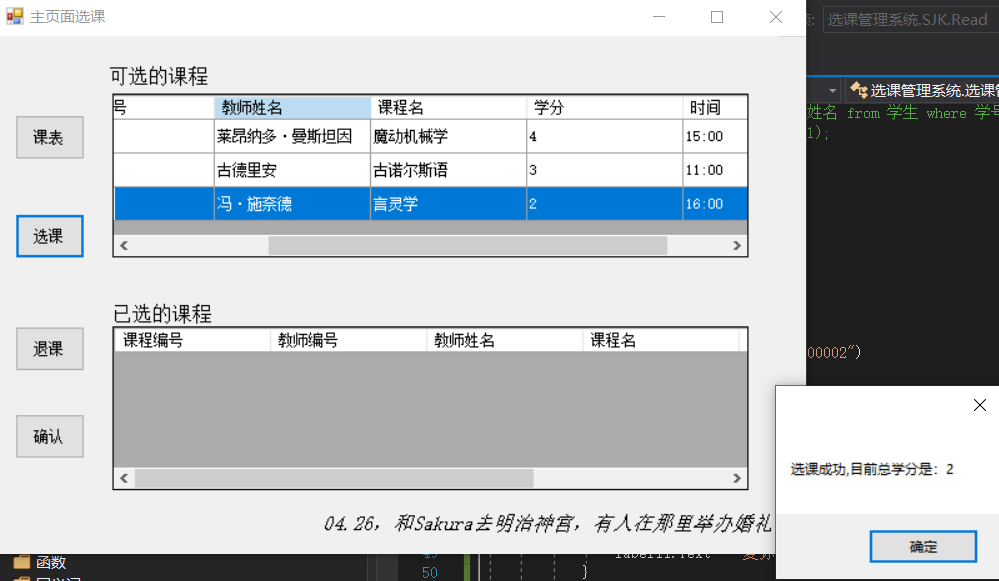
{

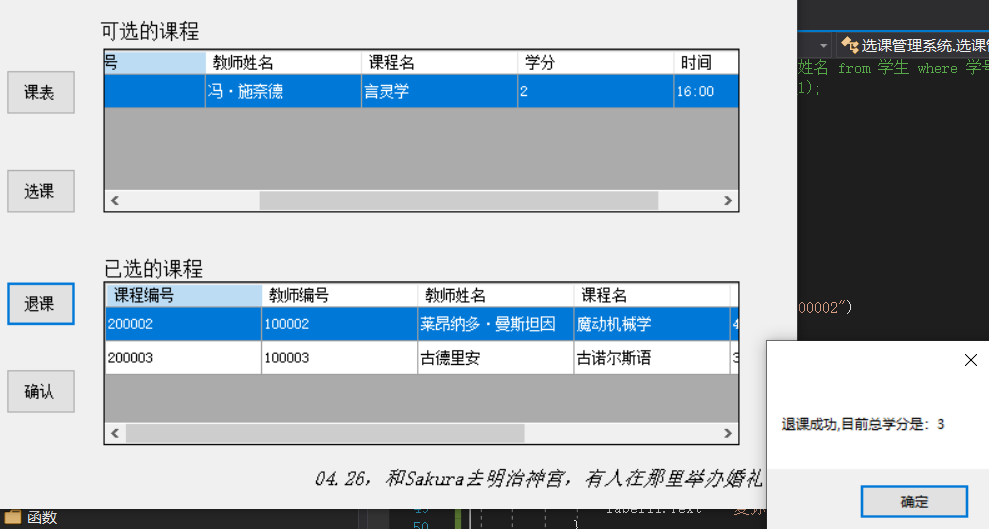
MessageBox.Show("我是BUG测试");

}

}

4.显示当前学生选择的课程数量、总学分。



当选中一门课程并将该课程添加至已选课程菜单栏时，系统会自动弹出两个窗口显示目前的总学分和已选课程数，同时退课后也会显示出退课后的学分和课程数。

这一块的函数功能代码实现如下：

选课：

Int cnt=0;

Int cnt2=0;

int A = int.Parse(credit);

cnt += A;

A = 0;

cnt2+=1;

MessageBox.Show("选课成功,目前总学分是："+cnt);

MessageBox.Show("总共选课数：" + cnt2);

function();

function2();

退课：

Int cnt=0;

Int cnt2=0;

int A = int.Parse(credit);

cnt -= A;

A = 0;

cnt2 -= 1;

MessageBox.Show("退课成功,目前总学分是：" + cnt);

MessageBox.Show("总共选课数：" + cnt2);

function();

function2();

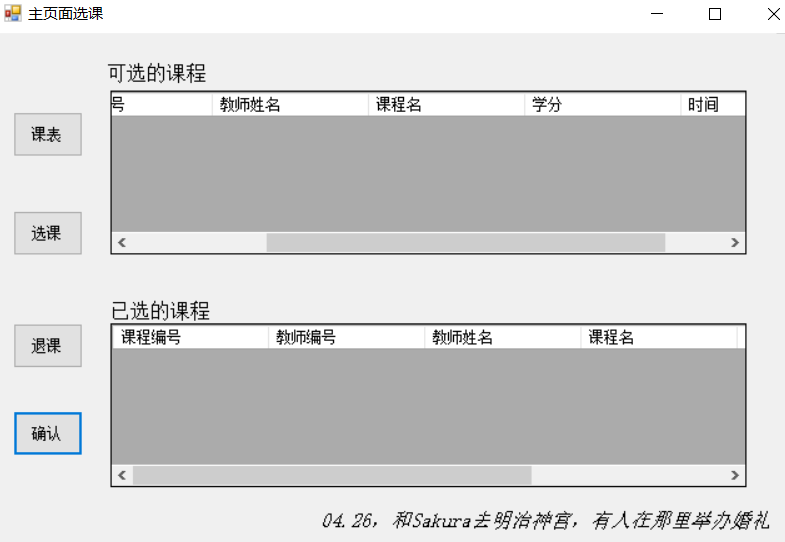
5.可以临时保存学生选择的课程信息，如果符合选课规则（3-5门、学分：8-12学分），则可提交选课结果。



如果条件不满足3-5门、学分：8-12学分的话则会跳出窗口显示无法达到要求



当选课条件满足要求之后点击按钮将会显示已将已选课程添加至课表当中，这时候就会发现已选的课程菜单栏中的课程都会消失不见，是因为这些数据全都被传输到课表里去了。



这一块的函数功能代码实现如下：

private void button4\_Click(object sender, EventArgs e)

{

if((cnt >= 8 && cnt2 >= 3) &&(cnt <= 12 && cnt2 <= 5))

{

string sql = "insert into 我的课表(课程编号,教师编号,教师名,课程名,学分,时间) select 课程编号,教师编号,教师名,课程名,学分,时间 from 当前课程";

SJK sjk = new SJK();

sjk.Execute(sql);

MessageBox.Show("已将已选的课程添加到你的课表当中，请按下课表按钮查看");

sql = "truncate table 当前课程";

sjk.Execute(sql);

function2();

sjk.SJKclose();

}

else

{

MessageBox.Show("您无法达到选课要求，您需要最低选择三门课八学分，最高选择五门课十二学分" );

MessageBox.Show("您的学分是："+cnt,"您目前的选课数：" + cnt2);

}

}



打开课表就会发现所有已选课程都进入到了我的课表当中。

这一块的函数功能代码实现如下：

private void kebiao\_Load(object sender, EventArgs e)

{

function();

}

public void function()

{

dataGridView1.Rows.Clear();

SJK sjk = new SJK();

string sql = "select \* from 我的课表";

IDataReader cc = sjk.Read(sql);

while (cc.Read())

{

dataGridView1.Rows.Add(cc[0].ToString(), cc[1].ToString(), cc[2].ToString(), cc[3].ToString(), cc[4].ToString(), cc[5].ToString());

}

cc.Close();

sjk.SJKclose();

}

1. 使用哪种数据提供程序？

使用.NET Framework 数据提供程序。

1. 使用的数据连接对象是哪一个？连接对象是如何建立的？最后生成的连接对象中的连接字符串是什么？代表什么含义？

点击VS2019上方的工具选项，点击连接到数据库选项，可以发现数据源是Microsoft SQL Server (SqlClient)，在服务器名栏里输入自己的计算机数据库名字，并在选择或输入数据库名称栏里选择自己建立的选课管理数据库，点击确定按钮就能自动连接到数据库了。连接字符串是Data Source=LAPTOP-Q6JM3HLF\SQLEXPRESS;Initial Catalog=数据库学生管理系统;User ID=sa;Pwd=123456。代表含义为该数据源是本台计算机。

1. 使用的数据适配器对象是什么？其中的查询或更新语句是什么？如果有参数则参数是如何处理的？

使用数据适配器为OleDbDataAdapter，查询语句为：SELECT列名称FROM表名称，更新语句为：UPDATE表名SET列名称=新值WHERE列名称=某值。如果出现参数则先用局部变量构造出整个语句，然后再为局部变量赋值。

1. 使用的数据集对象是什么？数据集中有哪些数据表？数据表是由哪些适配器对象生成的？（或采用其它方法）。

使用数据集对象为Dataset，数据集的数据表有选课表，已选课表和我的课表，数据表的生成方法是在数据库中直接创建表。