

南京信息工程大学

滨江学院

用户界面设计课程设计

题 目 :	基于 vb (c#) 的学生成绩管理系统设计与实现
学 号 :	20182344050
姓 名 :	毛济洲
专 业 :	软件工程
班 级	2018-2
指 导 教 师 :	王玉祥

2020 年 11 月 26 日

目录

基于 c#的学生成绩管理系统.....	3
第一章 绪论.....	3
1.1 研究背景.....	3
1.2 设计思想.....	3
第二章 系统分析	3
2.1 可行性分析	3
2.1.1 经济可行性	3
2.2 需求分析.....	4
第三章 系统设计	4
3.1 系统整体设计：	4
3.2 系统功能设计.....	4
3.3 数据库设计	5
3.3.1 数据库设计原则	5
3.3.2 数据库表设计	5
第四章 界面设计	7
第五章 代码实现	9
5.1 代码.....	9
5.1.1 Form.cs.....	9
5.1.2 Form2.cs.....	14
5.1.3 Form3.cs.....	30
5.1.4 Program.cs	34
第六章 总结.....	34
6.1 总结.....	34

基于 c#的学生成绩管理系统

第一章 绪论

1.1 研究背景

学生成绩信息管理系统是一个学校不可缺少的部分,传统的人工管理档案的方式存在着很多的缺点,如:效率低、保密性差等,所以开发一套综合教务系统管理软件很有必要,它应该具有传统的手工管理所无法比拟的优点,如:检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这些优点能够极大的提高学校的管理效率,有助于推进学校的信息化建设。

1.2 设计思想

学生成绩信息管理系统的设计与实现虽然功能比较简单,相对的操作也较为容易,界面设计较为简洁。首先要做的对实现系统所需的开发语言的掌握,然后对系统进行初步的分析,系统分析是开发程序的必要环节,通过调查研究,对系统的功能进行深入的分析,这样才能设计出一个较好的系统。

第二章 系统分析

2.1 可行性分析

对所有系统完成设计,都可能受到时间和空间上的制约。所以,在设计每一个项目开始,必须对系统实行可行性分析,这样可以降低项目的危害,降低人力、物力和财力的损耗。为了保证学生成绩信息管理系统软件的成功开发和正常运行,分别从技术、经济、操作、发展四方面对本系统进行可行性分析和研究。

2.1.1 经济可行性

学生成绩信息管理系统属于信息管理系统的一种,所以有很多通用的模块和流程,比如登录注册、找回密码、添加信息、查询信息、修改密码等功能。所以开发相对比较简单,数据库的设计也相对比较简单。

2.2 需求分析

需求分析的任务是通过详细调查现实管理系统软件情况，在了解了系统的业务逻辑后，明确要实现的功能模块，以此基础确定系统功能。

在开发中我们需要保证系统的安全性和有效性，我们要充分考虑以下几点：

安全性：身处信息时代，信息的安全问题在开发过程中一定要放在前列。

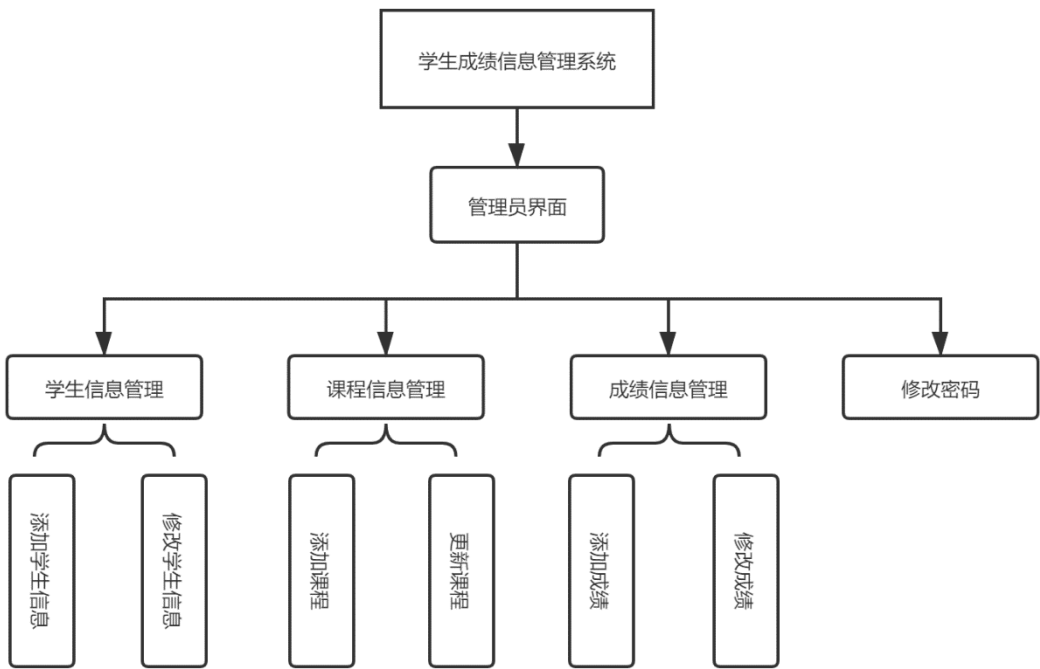
先进性：结合当下的趋势，符合用户需求，可以快速实现查询成绩的软件。

可扩展性：信息管理软件给用户带来了极大的便利，提高了查询效率，备受使用者的关注，因此在开发系统管理软件的过程中，我们要考虑到系统的可扩展性，便于将来系统的升级、更新。

有效性。有效性主要包括功能有效性，设计信息管理软件的功能一定要最大限度地满足用户的需求。

第三章 系统设计

3.1 系统整体设计：



3.2 系统功能设计

学生信息管理

添加学生信息；

管理员可以添加学生信息，这些信息包括学生的学号，密码，年级，专业，出生日期等等。

修改学生信息；

管理员可以修改学生的信息；

课程管理

开设课程；

管理员可以录入课程信息，学生就可以浏览到这些课程信息，从而选课。

查询课程；

可以通过学期查询课程，或通过学期和课程结合的方式查询课程。

成绩管理

录入成绩；

管理员可以录入学生的成绩。

统计成绩；

管理员可以通过学期和课程信息来统计某门课的学生成绩，平均分、最高分、最低分以及可以通过成绩区间来查询某个学期某门课的学生的成绩信息。并且可以实现排序功能。

修改成绩；

管理员可以对成绩出错的学生的成绩进行修改。

修改个人密码。

3.3 数据库设计

3.3.1 数据库设计原则

本系统的核心以及基础就是数据库。数据库设计的是否完备，决定了所开发系统的好与坏。无论是什么系统，其基础都必然是数据，换一种说法，设计出一个完善的数据库就好比建好一座大厦的地基，只有设计的合理了才能够在未来进行系统后续开发、维护、功能变更以及添加时不会出现较多错误。如若数据库设计的不完善，还需要再从头设计整个系统，做大量的重复工作。

3.3.2 数据库表设计

本系统采用 Oracle 数据库对数据进行存储，如图显示各表的详细信息
管理员信息表：

管理员信息表：manager				
字段名称	数据类型	可空	默认值	说明
manid	int	不可		主键，管理员 id，无实际意思，每次自增 1
manname	Nvarchar(30)	不可		管理员用户名
manpasswd	Nvarchar(30)	不可		管理员密码

学生信息表：

用户表：student				
字段名称	数据类型	可空	默认值	说明
stuid	int	不可		主键，学生 id，学号
stuname	Nvarchar(30)	不可		学生姓名
stugrade	Nvarchar(30)			学生年级
stumajor	Nvarchar(30)			学生专业
stusex	varchar(2)			学生性别

课程信息表：

课程表：class				
字段名称	数据类型	可空	默认值	说明
claid	int	不可		主键，课程 id，无实际意思，每次自增 1
claname	Nvarchar(30)	不可		课程名
term	Nvarchar(30)	不可		课程学分

成绩表：

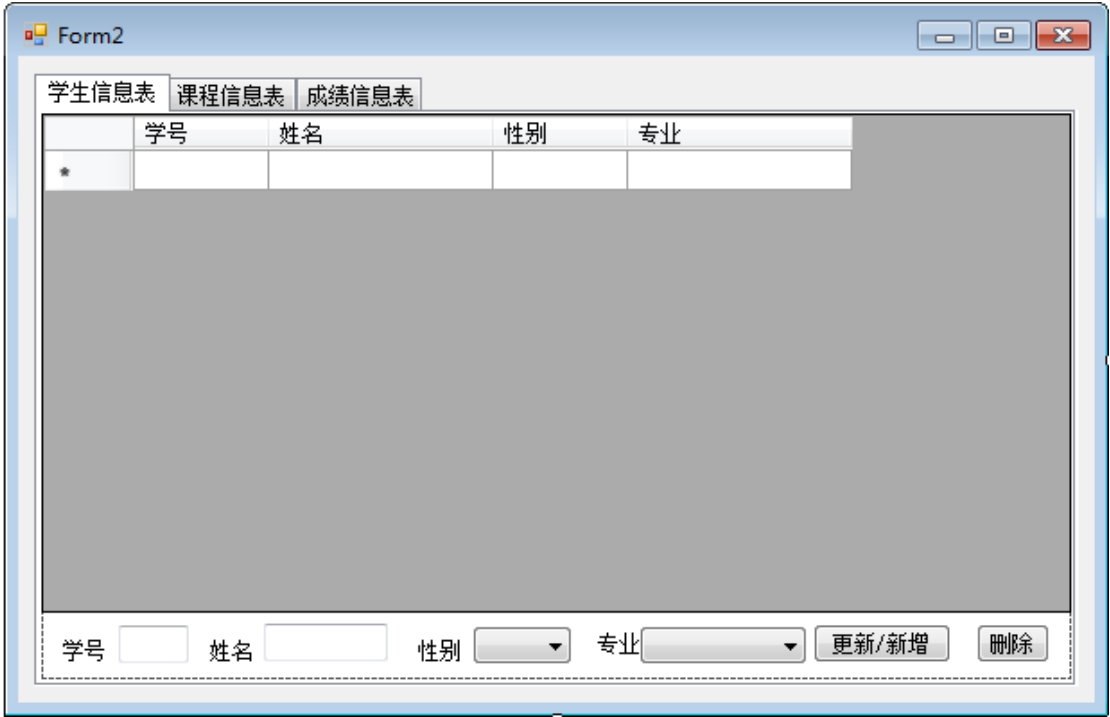
选课表：sc				
字段名称	数据类型	可空	默认值	说明
scid	int	不可		主键，选课 id，无实际意思，每次自增 1
stuid	int	不可		外键，学生 id
stuname	Nvarchar(30)	不可		学生姓名
claid	int	不可		外键，课程 id
claname	Nvarchar(30)	不可		课程名
grades	int	不可		学生成绩

第四章 界面设计

登录界面：



学生信息管理：



课程管理：

Form2

学生信息表 课程信息表 成绩信息表

	课程号	课程名	学分
*			

课程号 课程名 学分

成绩管理:

Form2

学生信息表 课程信息表 成绩信息表

	学号	姓名	课程号	课程名	分数
*					

学号 课程号 成绩

修改密码:



第五章 代码实现

5.1 代码

5.1.1 Form.cs

```
using System;
using System.Windows.Forms;
using MySql.Data.MySqlClient;
using System.Data.SqlClient;
using System.Data.OracleClient;
using System.IO;
using System.Runtime.Serialization.Formatters.Binary;

namespace Test3_1
{
    public partial class 用户登录 : Form
    {
        public 用户登录()
        {
            InitializeComponent();
        }
    }
}
```

```

private void cancel_Click(object sender, EventArgs e)
{
    this.Close();
}

private void determine_Click(object sender, EventArgs e)
{
    string mystr = "Data Source=ORCL;user Id=c##tiger;Password=
tiger";

    OracleConnection mycon = null;
    using (mycon = new OracleConnection(mystr)) {
        //打开数据库连接
        mycon.Open();

        string usr= user.Text;
        string passwd = password.Text;    //将密码的 string 类型转
换为 base64 格式
        string sql = "select * from users where userNo = '{0}'"
;

        sql = String.Format(sql, usr);
        OracleCommand command = new OracleCommand(sql,mycon);
        OracleDataReader reader = command.ExecuteReader();
        //判断 SQL 语句是否执行成功
        if (reader.Read())
        {
            reader.Close();
            string loginSql = "select * from users where UserNo
='{0}' and Password='{1}'";
            loginSql = String.Format(loginSql, usr, passwd);
            command = new OracleCommand(loginSql, mycon);
            reader = command.ExecuteReader();
            if (reader.Read()==true)
            {
                reader.Close();
                MessageBox.Show(" 欢迎使用学生信息管理系统。");

                //验证成
功,保存帐号

                FileStream stream = null;
                BinaryFormatter formater = null;
                try
                {

```

```

        stream= new FileStream("UserInfo.txt", FileMode.OpenOrCreate, FileAccess.Write, FileShare.None);
        formater= new BinaryFormatter();

        formater.Serialize(stream, usr);           //如果
        登陆成功并且记住密码选项勾选的情况下将账号保存在 Userinfo.txt 下面
        if (remeberPassword.Checked)

        {
            string pawd = stringToBase64String(password.Text);

            formater.Serialize(stream, "true");
            formater.Serialize(stream, pawd);
        }
        else
        {
            formater.Serialize(stream, "false");
        }
        this.Visible = false;
        Form2 f = new Form2(user.Text.Trim()); //
        创建窗体 Form2 的对象

        f.ShowDialog(); // 显示 Form2

    }
    catch (Exception e1)
    {
        MessageBox.Show(e1.Message);
    }
    finally {
        stream.Close();
        this.Close();
    }
}
else
{
    MessageBox.Show("密码错误。");
}

}
else
{
    MessageBox.Show("帐号不存在。");
}
}

```

```

    }

    private void password_TextChanged(object sender, EventArgs e)
    {

    }

    private void 用户登录_Load(object sender, EventArgs e)
    {
        if (File.Exists("UserInfo.txt"))
        {
            FileStream readstream = null;
            BinaryFormatter formater = null;
            try {
                readstream = new FileStream("UserInfo.txt", FileMode.Open, FileAccess.Read, FileShare.Read);
                formater = new BinaryFormatter();
                string readdata = (string)formater.Deserialize(readstream);

                user.Text = readdata;
                string readdata1 = (string)formater.Deserialize(readstream);

                if ("true".Equals(readdata1)){
                    remeberPassword.Checked = true;
                    string pawd = (string)formater.Deserialize(readstream);

                    pawd = base64StringToString(pawd);
                    password.Text = pawd;
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            finally
            {
                readstream.Close();
            }
        }
    }
}

```

```
private void checkBox1_CheckedChanged(object sender, EventArgs
e)
{
    if (remeberPassword.Checked)
    {

    }
}
public string stringToBase64String(string s)
{
    try
    {
        byte[] b = System.Text.Encoding.Default.GetBytes(s);
        //转成 Base64 形式的 System.String

        return Convert.ToBase64String(b);

    }
    catch
    {
        return null;
    }
}

//base64 编码的字符串转为正常字符串
public string base64StringToString(string a)
{
    try
    {
        byte[] c = Convert.FromBase64String(a);
        return System.Text.Encoding.Default.GetString(c);
    }
    catch
    {
        return null;
    }
}
}
```

5.1.2 Form2.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.OracleClient;
using System.Drawing;
using System.Text;
using System.Text.RegularExpressions;
using System.Windows.Forms;

namespace Test3_1
{
    public partial class Form2 : Form
    {
        string userr;
        string mystr = "Data Source=ORCL;user Id=c##tiger;Password=tiger";

        public Form2()
        {
            InitializeComponent();
            dataGridView1.AutoGenerateColumns = false;
        }
        public Form2(string userr)
        {
            this.userr = userr;
            InitializeComponent();
            dataGridView1.AutoGenerateColumns = false;
        }

        private void Form2_Load(object sender, EventArgs e)
        {

            tab1_Load(mystr);
//设置 tab1 绑定源

            tab2_Load(mystr);
//设置 tab2 绑定源

```

```

        tab3_Load(mystr);
        //设置 tab3 绑定源

    }
    private void tab1_Load(string mystr)                //加载学生信息表
    {
        DataSet studentDs = new DataSet();
        string sql = "select * from Students";
        using (OracleDataAdapter da = new OracleDataAdapter(sql, my
str))
        {
            da.Fill(studentDs);
        }
        try
        {
            dataGridView1.DataSource = null;            //将绑定关系
设置为空

            studentNumber.DataBindings.Clear();
            name.DataBindings.Clear();
            sexKind.DataBindings.Clear();
            subject.DataBindings.Clear();

            dataGridView1.DataSource = studentDs.Tables[0];
//datagridview1 绑定源
            studentNumber.DataBindings.Add("Text", studentDs.Tables
[0], "studentno"); //设置学号绑定源
            name.DataBindings.Add("Text", studentDs.Tables[0], "Stu
dentName");        //姓名添加绑定源
            sexKind.DataBindings.Add("Text", studentDs.Tables[0], "
sexKind");          //设置性别绑定源
            subject.DataBindings.Add("Text", studentDs.Tables[0], "
Major");            //设置专业绑定源
        }
        catch (Exception ex)
        {
            MessageBox.Show("students: " + ex.Message);
        }
    }
    private void tab2_Load(string mystr)                //记载课
程信息表
    {
        DataSet courseDs = new DataSet();
        string courseSql = "select * from courses";    //查找课程
表 sql

```

```

        using (OracleDataAdapter da = new OracleDataAdapter(courseS
ql, mystr))
        {
            da.Fill(courseDs);
        }
        try
        {
            dataGridView2.DataSource = null;           //将原有绑定
关系设置为空

            courseNo.DataBindings.Clear();
            courseName.DataBindings.Clear();
            credit.DataBindings.Clear();

            dataGridView2.DataSource = courseDs.Tables[0];
            //课程表 gridview 添加绑定源
            courseNo.DataBindings.Add("Text", courseDs.Tables[0], "
courseNo"); //课程号添加绑定源
            courseName.DataBindings.Add("Text", courseDs.Tables[0],
"courseName"); //课程名添加绑定源
            credit.DataBindings.Add("Text", courseDs.Tables[0], "cr
edit"); //学分添加绑定源
        }
        catch (Exception ex)
        {
            MessageBox.Show("course:" + ex.Message);
        }
    }
    private void tab3_Load(string mystr)           //加载成绩信
息表
    {
        DataSet ds = new DataSet();
        string scoreSql = "select scores.studentno,studentname,scores.courseno,coursename,score" + //查找成绩表 sql
            " from COURSES, STUDENTS, scores " +
            " where students.studentno = scores.student
no and courses.courseno = scores.courseno"; //查找成绩表 sql
        using (OracleDataAdapter da = new OracleDataAdapter(scoreSql, mystr))
        {
            da.Fill(ds);
        }
        try
        {

```



```

        dataGridView3.DataSource = null; //将
原有绑定关系设置为空
        stunumber.DataBindings.Clear();
        courseNu.DataBindings.Clear();
        grade.DataBindings.Clear();

        dataGridView3.DataSource = ds.Tables[0];
        //成绩表 gridview 添加绑定源
        stunumber.DataBindings.Add("Text", ds.Tables[0], "stude
ntno"); //学号添加绑定源
        courseNu.DataBindings.Add("Text", ds.Tables[0], "course
no"); //课程号添加绑定源
        grade.DataBindings.Add("Text", ds.Tables[0], "score");
        //成绩添加绑定源
    }
    catch (Exception ex)
    {
        MessageBox.Show("score:" + ex.Message);
    }
}

private void studentUorC_Click(object sender, EventArgs e)
{
    String stu = studentNumber.Text.Trim();
    String nam = name.Text.Trim();
    String sex = sexKind.Text.Trim();
    String sub = subject.Text.Trim();
    OracleConnection conn = null;
    OracleDataReader reader = null;
    try
    {
        conn = new OracleConnection(mystr);
        conn.Open();
        {
            string selectSql = "select * from students where ST
UDENTNO='{0}'";
            selectSql = String.Format(selectSql, stu);
            reader = new OracleCommand(selectSql, conn).Execute
Reader();

            if (reader.Read()) //判断 sql 语句是否执行成功
            {
                //记录已存在
                reader.Close();
                string updateSql = "update students set STUDEN
TNAME='{0}',SEXKIND='{1}', MAJOR='{2}' where STUDENTNO={3}";

```

```

        updateSql = String.Format(updateSql, nam, sex,
sub, stu);

        int update = new OracleCommand(updateSql, conn)
.ExecuteNonQuery();

        if (update > 0)
        {
            MessageBox.Show("更新了学生表的一条记录");
            tab1_Load(mystr);
            //dataGridView1_Load(mystr, ds);
        }
        else if (update == -1)
        {
            MessageBox.Show("update 语句执行错误");
        }
        else
        {
            MessageBox.Show("没有修改任何记录");
        }
    }
    else
    {
        reader.Close();
        string insertSql = "insert into students(studen
tno,studentname,sexkind,major) values({0}','{1}','{2}','{3}')";
        insertSql = String.Format(insertSql, stu, nam,
sex, sub);

        int insert = new OracleCommand(insertSql, conn)
.ExecuteNonQuery();

        if (insert > 0)
        {
            MessageBox.Show("增加了一条学生记录");
            tab1_Load(mystr);
        }
        else if (insert == -1)
        {
            MessageBox.Show("insert 语句执行错误");
        }
        else
        {
            MessageBox.Show("未增加学生记录");
        }
    }
}

```

```

    }
}
catch (Exception e1)
{
    MessageBox.Show(e1.Message);
}
finally
{
    conn.Close();
}
}

private void delete_Click(object sender, EventArgs e)
//学生表删除记录
{
    String stu = studentNumber.Text.Trim();
    if (MessageBox.Show("是否删除学号为"+ stu+"的学生", "确认",
        MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        OracleConnection conn = null;
        OracleDataReader reader = null;
        try
        {
            conn = new OracleConnection(mystr);
            conn.Open();
            {
                string selectSql = "select * from students where
                e STUDENTNO='{0}'";

                selectSql = String.Format(selectSql, stu);
                reader = new OracleCommand(selectSql, conn).ExecuteReader();

                if (reader.Read()) //判断查询 sql 语句是否执行成功
                { //记录已
                存在

                    reader.Close();

                    string selectGradeSql = "select * from scores where studentno='{0}'";
                    selectGradeSql = String.Format(selectGradeSql, stu);

                    if (new OracleCommand(selectGradeSql, conn).ExecuteScalar() == null)
                    { //判断是否有该学号的成绩记录
                        //没有该学号的成绩记录

```

```

        string deleteSql = "delete students whe
re  STUDENTNO={0}";

        deleteSql = String.Format(deleteSql, st
u);

        int update = new OracleCommand(deleteSq
l, conn).ExecuteNonQuery();

        if (update > 0)
        {
            MessageBox.Show("删除了学生表的一条记
录");

            tab1_Load(mystr);
        }
        else if (update == -1)
        {
            MessageBox.Show("delete 语句执行错误
");

        }
        else
        {
            MessageBox.Show("没有删除任何记录");
        }
    }
    else
    {
        MessageBox.Show("存在该学生的成绩记录,不能
删除");
    }
}
else
{
    reader.Close();
    MessageBox.Show("不存在学号的记录");
}
}

}
catch (Exception ex)
{
    MessageBox.Show("deleteStudent" + ex.Message);
}
finally
{
    conn.Close();
}
}

```

```

    }
}
private void courseUorC_Click(object sender, EventArgs e)
//课程表更新或创建记录
{
    String couNo = courseNo.Text.Trim();
    String courName = courseName.Text.Trim();
    String cred = credit.Text.Trim();
    OracleConnection conn = null;
    OracleDataReader reader = null;
    try
    {
        conn = new OracleConnection(mystr);
        conn.Open();
        {
            string selectSql = "select * from courses where cou
rseno='{0}'";
            selectSql = String.Format(selectSql, couNo);
            reader = new OracleCommand(selectSql, conn).Execute
Reader();
            if (reader.Read()) //判断 sql 语句是否执行成功
            {
                //记录已存在
                reader.Close();
                string updateSql = "update courses set coursename='{0}',credit='{1}' where courseno={2}";
                updateSql = String.Format(updateSql, courName,cred ,couNo );
                int update = new OracleCommand(updateSql, conn).ExecuteNonQuery();
                if (update > 0)
                {
                    MessageBox.Show("更新了课程表的一条记录");
                    tab2_Load(mystr);
                }
                else if (update == -1)
                {
                    MessageBox.Show("update 语句执行错误");
                }
                else
                {
                    MessageBox.Show("没有修改任何记录");
                }
            }
        }
    }
}

```

```

        else
        {
            reader.Close();
            string insertSql = "insert into courses values(
'{0}','{1}','{2}')"";
            insertSql = String.Format(insertSql, couNo, cou
rName, cred);
            int insert = new OracleCommand(insertSql, conn)
.ExecuteNonQuery();
            if (insert > 0)
            {
                MessageBox.Show("增加了一条课程记录");
                tab2_Load(mystr);
            }
            else if (insert == -1)
            {
                MessageBox.Show("insert 语句执行错误");
            }
            else
            {
                MessageBox.Show("未增加课程记录");
            }
        }
    }
}
catch (Exception e1)
{
    MessageBox.Show(e1.Message);
}
finally
{
    conn.Close();
}
}

private void courseDelete_Click(object sender, EventArgs e)
//课程表删除记录
{
    String couNo = courseNo.Text.Trim();
    if (MessageBox.Show("是否删除课程号为"+ couNo + "的课程", "确
认", MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        OracleConnection conn = null;
    }
}

```

```

        OracleDataReader reader = null;
        try
        {
            conn = new OracleConnection(mystr);
            conn.Open();
            {
                string selectSql = "select * from courses where
courseno='{0}'";

                selectSql = String.Format(selectSql, couNo);
                reader = new OracleCommand(selectSql, conn).ExecuteReader();

                if (reader.Read()) //判断查询 sql 语句是否执行成功
                { //记录已
存在
                    reader.Close();

                    string selectGradeSql = "select * from scores where courseno='{0}'";
                    selectGradeSql = String.Format(selectGradeSql, couNo);

                    if (new OracleCommand(selectGradeSql, conn).ExecuteScalar() == null)
                    { //判断是否有该课程号的成绩记录
                        //没有该课程号的成绩记录
                        string deleteSql = "delete courses where courseno={0}";

                        deleteSql = String.Format(deleteSql, couNo);

                        int delete = new OracleCommand(deleteSql, conn).ExecuteNonQuery();

                        if (delete > 0)
                        {
                            MessageBox.Show("删除了课程表的一条记录");

                            tab2_Load(mystr);
                        }
                    }
                    else if (delete == -1)
                    {
                        MessageBox.Show("delete 语句执行错误");
                    }
                }
            }
        }
    }
}

```

```

        MessageBox.Show("没有删除任何记录");
    }
}
else
{
    MessageBox.Show("存在该课程的成绩记录,不能
删除");
}
}
else
{
    reader.Close();
    MessageBox.Show("不存在课程号的记录");
}
}
}
}
catch (Exception ex)
{
    MessageBox.Show("deleteCourse" + ex.Message);
}
finally
{
    conn.Close();
}
}
}

private void gradeUorC_Click(object sender, EventArgs e)
//成绩表更新或创建记录
{
    String stunum = stunumber.Text.Trim();
    String cournu = courseNu.Text.Trim();
    String grad = grade.Text.Trim();
    OracleConnection conn = null;
    OracleDataReader reader = null;
    try
    {
        conn = new OracleConnection(mystr);
        conn.Open();
        {
            string selectSql = "select * from scores where cour
seno='{0}' and studentno='{1}'";

```



```

        selectSql = String.Format(selectSql, cournu,stunum)
;
        reader = new OracleCommand(selectSql, conn).Execute
Reader();
        if (reader.Read()) //判断 sql 语句是否执行成功
        {
            //记录已存在
            reader.Close();
            string updateSql = "update scores set credit='
{2}' where courseno='{0}' and studentno='{1}'";
            updateSql = String.Format(updateSql, cournu, st
unum,grad);
            int update = new OracleCommand(updateSql, conn)
.ExecuteNonQuery();
            if (update > 0)
            {
                MessageBox.Show("更新了成绩表的一条记录");
                tab3_Load(mystr);
            }
            else if (update == -1)
            {
                MessageBox.Show("update 语句执行错误");
            }
            else
            {
                MessageBox.Show("没有修改任何记录");
            }
        }
        else
        {
            reader.Close();
            string insertSql = "insert into scores values('
{0}','{1}','{2}')";
            insertSql = String.Format(insertSql, stunum, co
urnu, grad);
            int insert = new OracleCommand(insertSql, conn)
.ExecuteNonQuery();
            if (insert > 0)
            {
                MessageBox.Show("增加了一条成绩记录");
                tab3_Load(mystr);
            }
            else if (insert == -1)
            {
                MessageBox.Show("insert 语句执行错误");
            }
        }
    }
}

```

```

        }
        else
        {
            MessageBox.Show("未增加成绩记录");
        }
    }

    }
}
catch (Exception e1)
{
    MessageBox.Show(e1.Message);
}
finally
{
    conn.Close();
}
}

private void button1_Click(object sender, EventArgs e)
//成绩表删除记录
{
    String stunum = stunumber.Text.Trim();
    String cournu = courseNu.Text.Trim();
    if (MessageBox.Show("是否删除课程号为" + cournu + ",学号为"
    "+stunum+ "的成绩", "确认
    ", MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        OracleConnection conn = null;
        OracleDataReader reader = null;
        try
        {
            conn = new OracleConnection(mystr);
            conn.Open();
            {
                string selectSql = "select * from scores where
                courseno='{0}' and studentno='{1}'";
                selectSql = String.Format(selectSql, cournu, stunum);
                reader = new OracleCommand(selectSql, conn).ExecuteReader();

                if (reader.Read()) //判断查询 sql 语句是否执行成功
                {
                    //记录已
                    存在
                }
            }
        }
    }
}

```

```

        reader.Close();
        string deleteSql = "delete scores where cou
rseno='{0}' and studentno='{1}'";
        deleteSql = String.Format(deleteSql, cournu
, stunum);

        int delete = new OracleCommand(deleteSql, c
onn).ExecuteNonQuery();

        if (delete > 0)
        {
            MessageBox.Show("删除了成绩表的一条记录
");

            tab3_Load(mystr);

        }
        else if (delete == -1)
        {
            MessageBox.Show("delete 语句执行错误");
        }
        else
        {
            MessageBox.Show("没有删除任何记录");
        }

    }
    else
    {
        reader.Close();
        MessageBox.Show("不存在学号="+stunum+",课程号
="+cournu+"的记录");
    }
}

}
catch (Exception ex)
{
    MessageBox.Show("deleteGrade" + ex.Message);
}
finally
{
    conn.Close();
}
}
}

```

```

        private void studentNumber_TextChanged(object sender, EventArgs e)
        //判断学号是否是一个正整数
        {
            if (!isInteger(studentNumber.Text.Trim()))
            {
                MessageBox.Show("输入的学号不是一个正整数");
            }
        }

        private void name_TextChanged(object sender, EventArgs e)
        //判断姓名长度
        {
            if (name.Text.Length < 1 || name.Text.Length > 9)
            {
                MessageBox.Show("姓名字符串长度在 1-9");
            }
        }

        private void sexKind_SelectedIndexChanged(object sender, EventArgs e)
        //判断性别和专业是否选了
        {
            if (sexKind.Text == null || subject.Text == null)
            {
                MessageBox.Show("没有选择性别或专业.");
            }
        }

        private void subject_SelectedIndexChanged(object sender, EventArgs e)
        //判断性别和专业是否选了
        {
            sexKind_SelectedIndexChanged(sender, e);
        }

        private void courseNo_TextChanged(object sender, EventArgs e)
        //判断课程号是否为一个正整数
        {
            if (!isInteger(courseNo.Text.Trim()))
            {
                MessageBox.Show("课程号不是一个正整数");
            }
        }

```

```

    }

    private void credit_TextChanged(object sender, EventArgs e)
    {

        Regex regex = new Regex("^(-?\d+)(\.\d+)?$");    //判断
是否为实数的正则表达式
        if (!regex.IsMatch(credit.Text.Trim()))
        {
            MessageBox.Show("学分不是一个实数");
        }
    }

    private void stunumber_TextChanged(object sender, EventArgs e)
//判断学号是否是一个正整数
    {
        if (!isInteger(stunumber.Text.Trim()))
        {
            MessageBox.Show("输入的学号不是一个正整数");
        }
    }

    private void courseNu_TextChanged(object sender, EventArgs e)
//判断课程号是否为一个整数
    {
        if (!isInteger(courseNu.Text.Trim()))
        {
            MessageBox.Show("课程号不是一个正整数");
        }
    }

    private void grade_TextChanged(object sender, EventArgs e)
//判断成绩是否为一个整数
    {
        if (!isInteger(grade.Text.Trim()))
        {
            MessageBox.Show("成绩不是一个正整数");
        }
        else
        {
            int inge = int.Parse(grade.Text.Trim());
            if (inge < 0 && inge > 100)
            {
                MessageBox.Show("成绩在 0-100 范围之内");
            }
        }
    }

```

```

    }

}

private bool isInteger(string s)
{
    Regex regex = new Regex("[0-9]*[1-9][0-9]*$");    //判断是
否为正整数的正则表达式
    return regex.IsMatch(s);
}

private void 重置密码
ToolStripMenuItem_Click(object sender, EventArgs e)
{
    string user = userr;
    Form3 form3 = new Form3(user);
    form3.ShowDialog();
}
}
}

```

5.1.3 Form3.cs

```

5.1. using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.OracleClient;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace Test3_1
{
    public partial class Form3 : Form
    {
        string user;
        string mystr = "Data Source=ORCL;user Id=c##tiger;Password=tige
r";
        public Form3()
    }
}

```

```
{
    InitializeComponent();
}
public Form3(string s)
{
    this.user = s;
    InitializeComponent();
}

private void Form3_Load(object sender, EventArgs e)
{

}

private void label1_Click(object sender, EventArgs e)
{

}

private void textBox3_TextChanged(object sender, EventArgs e)
{

}

private void textBox1_TextChanged(object sender, EventArgs e)
{

}

private void close_Click(object sender, EventArgs e)
{
    this.Close();
}

private void textBox2_TextChanged(object sender, EventArgs e)
{

}

}
public bool isInputCorrect()
{
    if (String.IsNullOrEmpty(textBox1.Text))
    {
        MessageBox.Show("原密码不能为空");
    }
}
```

```

        return false;
    }
    if (String.IsNullOrEmpty(textBox2.Text))
    {
        MessageBox.Show("新密码不能为空");
        return false;
    }
    if (String.IsNullOrEmpty(textBox3.Text))
    {
        MessageBox.Show("重置密码不能为空");
        return false;
    }
    if (!textBox3.Text.Equals(textBox2.Text))
    {
        MessageBox.Show("两次密码不一样");
        return false;
    }
    return true;
}

private void submit_Click(object sender, EventArgs e)
{
    if (isInputCorrect()) { //如果输入正确
        OracleConnection conn = null;
        OracleDataReader reader = null;
        string originPwd = textBox1.Text.Trim();
        string updatePwd = textBox2.Text.Trim();
        try
        {
            conn = new OracleConnection(mystr);
            conn.Open();
            string selectSql = "select * from users where UserN
o='{0}' and Password='{1}'";
            selectSql = String.Format(selectSql, user, originPw
d);
            reader = new OracleCommand(selectSql, conn).Execute
Reader();

            MessageBox.Show(selectSql);
            if (reader.Read())
            {
                reader.Close();
                string updateSql = "update users set password='
{0}' where userno='{1}'";

```



```

        updateSql = String.Format(updateSql, updatePwd,
user);

        reader= new OracleCommand(updateSql, conn).ExecuteReader();

        if (reader.Read())
        {
            reader.Close();
            MessageBox.Show("修改密码成功");
        }
        else
        {
            MessageBox.Show("修改密码未成功");
        }
    }
    else
    {
        reader.Close();
        MessageBox.Show("原密码错误");
    }
}
catch(Exception ex)
{
    MessageBox.Show(ex.Message);
}
finally
{
    conn.Close();
}
}

private void label2_Click(object sender, EventArgs e)
{
}

private void label3_Click(object sender, EventArgs e)
{
}

}
}

```

5.1.4 Program.cs

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace Test3_1
{
    static class Program
    {
        /// <summary>
        /// 应用程序的主入口点。
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new 用户登录());
        }
    }
}
```

第六章 总结

6.1 总结

次系统设计中遇到的最困难的方面就是在数据库相关的知识,在最初进行设计遇到了许多的困难,但在我不断的坚持,与指导老师的帮助下,最终完成了本次的系统设计。

在此次系统的开发过程中,我通过 c#技术、Oracle 数据库和在校期间所学习到的一些相关技术以及指导老师的帮助下使我设计的整个系统得到了非常大的提升。学生成绩管理系统其实还是存在很多不完善的地方,例如页面图片的大小比例不是很完美,自己找的图片素材太过零散,导致某一模块的界面不是很统一,有些功能模块还需要加强。在接下来的日子里,我将会对自己所设计的系统进行全面的完善,使其达到真正能完美符合用户的需求。