

Visual C#.Net 程序设计课程设计报告

(2020—2021学年 第二学期)

系 别 信息与控制工程

专 业 计算机科学与技术

班 级 计算机2001

学 号 203230133

姓 名 颜奇昌

指导教师 贺飞

完成时间

评定成绩

目录

[一、功能需求分析。 3](#_Toc25936)

[1、1功能分析 3](#_Toc24713)

[(1) 用户登录管理: 3](#_Toc3840)

[(2) 员工信息管理: 3](#_Toc359)

[(3)工资信息管理: 3](#_Toc22081)

[(3) 考勤信息管理: 3](#_Toc9012)

[(4) 奖惩信息管理: 3](#_Toc10956)

[1、2需求分析 3](#_Toc5598)

[二、概念设计模型： 3](#_Toc10119)

[2、1 E-R图设计 3](#_Toc12448)

[2、2实体属性 4](#_Toc30467)

[三、 逻辑设计模型： 5](#_Toc27396)

[3.1数据字典 5](#_Toc13558)

[3.2系统功能模块 6](#_Toc13033)

[四、数据库设计 6](#_Toc31910)

[五、界面设置 8](#_Toc4066)

[5、1主页面设计 8](#_Toc4939)

[（1）登录界面 8](#_Toc26214)

[（2）注册界面 10](#_Toc22270)

[5、2员工管理 12](#_Toc428)

[（1）新增员工信息 12](#_Toc7602)

[（2）修改员工信息 14](#_Toc289)

[（3）查询员工信息 15](#_Toc19774)

[5、3工资管理 16](#_Toc28381)

[（1）修改工资 16](#_Toc31884)

[（2）添加工资信息 17](#_Toc11914)

[（3）查询工资信息 19](#_Toc12132)

[5、4考勤管理 20](#_Toc12425)

[（1）添加考勤记录 20](#_Toc2414)

[（2）修改考勤记录 23](#_Toc29548)

[（3）查询考勤记录 25](#_Toc31328)

[5、5奖惩管理 25](#_Toc28143)

[（1）修改奖惩信息 25](#_Toc13421)

[（2）查询奖惩信息 27](#_Toc3615)

[六、 程序特点 28](#_Toc1912)

[七、心得体会 28](#_Toc7759)

# 一、功能需求分析。

## 1、1功能分析

1. 用户登录管理:

创建新的用户；

删除新的用户 ；

1. 员工信息管理:

可以查询企业员工的基本信息；

员工基本信息的修改；

员工基本信息的删除；

(3)工资信息管理:

工资信息的修改；

工资信息的删除；

工资信息的查询；

1. 考勤信息管理:

查询考勤信息;

考勤的修改；

考勤的删除；

考勤的查询；

1. 奖惩信息管理:

查询奖惩信息。

奖惩信息的修改；

奖惩信息的删除；

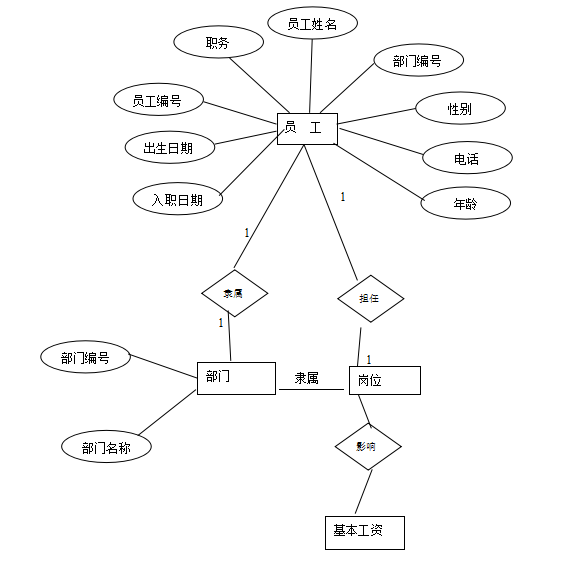
奖惩信息的查询

1、2需求分析

1. 用户登录信息表：用户账号，用户密码；
2. 员工信息表：员工编号，姓名，年龄，性别，电话，部门，岗位；
3. )部门信息表：部门编号，部门名称，员工编号；
4. 职称信息表：职称等级编号，职位，所属部门；
5. 工资信息表：员工编号，职位级别，职位，基本工资。

# 二、概念设计模型：

## 2、1 E-R图设计



## 2、2实体属性

员工：员工编号，姓名，年龄，性别，电话，部门编号，出生日期，入职日期

部门：部门编号，部门名称，员工编号

岗位：职称等级编号，职位，所属部门

基本工资：员工编号，职位级别，职位，基本工资

各联系的属性为：

员工隶属：隶属部门

员工担任：担任岗位

岗位影响：工资数

# 三、 逻辑设计模型：

员工（员工编号，姓名，年龄，性别，电话，部门编号，出生日期，入职日期）

部门（部门编号，部门名称，员工编号）

岗位（岗位编号，岗位名称，所属部门）

基本工资（员工编号，岗位级别，岗位，基本工资）

员工隶属（员工编号，部门编号，隶属部门）

员工担任（员工编号，岗位编号，担任岗位）

岗位影响（员工编号，工资数）

## 3.1数据字典

**岗位表结构**

|  |  |  |
| --- | --- | --- |
| **属性名** | **数据类型** | **说明** |
| 岗位名称 | char (10) | Not null |
| 岗位编号 | char (10) | 主键 |
| 基本工资 | char(20) | not null |
| 员工编号 | char(10) | not null |

**部门表结构**

|  |  |  |
| --- | --- | --- |
| **属性名** | **数据类型** | **说明** |
| 部门编号 | char (10) | 主键 |
| 部门名称 | char (10) | not null |

**员工表结构**

|  |  |  |
| --- | --- | --- |
| **属性名** | **数据类型** | **说明** |
| 员工编号 | char (20) | 主键 |
| 员工姓名 | char (10) | not null |
| 部门编号 | char (10) |  |
| 职务 | char (10) | not null |
| 性别 | char (10) | not null |
| 年龄 | smallint | Not null |
| 出生日期 | char (20） | Not null |
| 电话 | char (20) |  |
| 入职日期 | char (20 | Not null |

**工资表结构**

|  |  |  |
| --- | --- | --- |
| **属性名** | **数据类型** | **说明** |
| 员工编号 | char (10) | 主键 |
| 基本工资 | int | not null |
| 职位名称 | Char(20) | not null |

**用户登录信息**

|  |  |  |
| --- | --- | --- |
| **属性名** | **数据类型** | **说明** |
| Account（SID） | char (20) | 主键 |
| password | Char(20) | not null |

## 3.2系统功能模块

人力资源管理系统

部门管理

招聘管理

人事管理

工

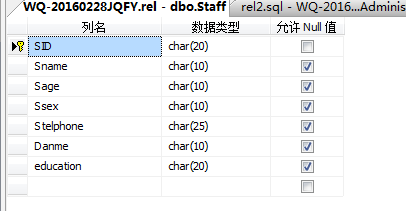
资

管

理

# 四、数据库设计

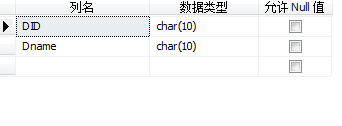
（1）员工Staff表



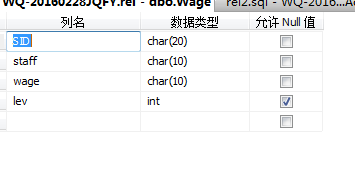
（2）职位表Position



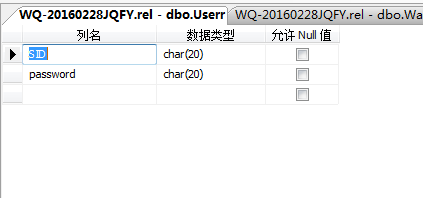
（3）部门信息表Department



（4）工资信息表Wage



（5）用户登录表Userr



# 五、界面设置

## 5、1主页面设计

### （1）登录界面



主要部分代码：

public Login()

{

InitializeComponent();

}

private bool CheckInput()//新定义一个方法，用于判断登陆是否合法

{

if (this.txtUserName.Text.Trim().Equals(string.Empty))

{

MessageBox.Show("请您输入账号！", "操作提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.txtUserName.Focus();

return false;

}

else if (this.txtPassWord.Text.Trim().Equals(string.Empty))

{

MessageBox.Show("请您输入密码！", "操作提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.txtPassWord.Focus();

return false;

}

else if (this.cmbType.Text.Trim().Equals(string.Empty))

{

MessageBox.Show("请选择登陆类型！", "操作提示", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.cmbType.Focus();

return false;

}

else

{

return true;

}

}

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

{

string UserName = txtUserName.Text.Trim();

string UserPassWord = txtPassWord.Text.Trim();

string UserType = cmbType.Text.Trim();

if (CheckInput())//用户名，密码，登录类型都不能为空

{

//添加一个登陆密码判断

SqlConnection conn = new SqlConnection("Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

string str = "select count(\*) from UserTable where UserName='" + UserName + "' and PassWord='" + UserPassWord + "'and Type='" + UserType+ "'";

SqlCommand cmd = new SqlCommand(str, conn);

int n = (int)cmd.ExecuteScalar();

if (n >= 1)

{

if (this.cmbType.Text.Equals("管理员"))

{

AdminForm frmAdmin = new AdminForm();//新建一个管理员窗体,打开管理员界面

frmAdmin.Show();

}

else//非管理员登陆进入这里

{

AdminForm frmOtherUser = new AdminForm();//新建一个非管理员界面

frmOtherUser.tsbUserManagement.Visible=false;

frmOtherUser.Show();

}

this.Hide();

}

else

{

MessageBox.Show("您输入的用户名，密码或类型有误，请重新输入！");

}

conn.Close();

}

}

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

{

DialogResult result = MessageBox.Show("你确定要退出么？", "操作提示", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

Application.Exit();

}

}

### （2）注册界面



部分代码：

private void button1\_Click(object sender, EventArgs e) //提交到数据库的Userr表中

{

string strCon = "Data Source=WQ-20160228JQFY;Initial Catalog=rel; Trusted\_Connection=true;";

string ac = account.Text;

string pass = word.Text;

SqlConnection sqlCon = new SqlConnection(strCon);

sqlCon.Open();

string strSql = "insert into Userr VALUES('" + ac + "','" + pass + "');";

SqlCommand cmd = new SqlCommand(strSql, sqlCon);

string str = "INSERT INTO Staff(SID) values('"+ac+"');";

SqlCommand cmdd = new SqlCommand(str, sqlCon);

if (cmd.ExecuteNonQuery() == 1)

{

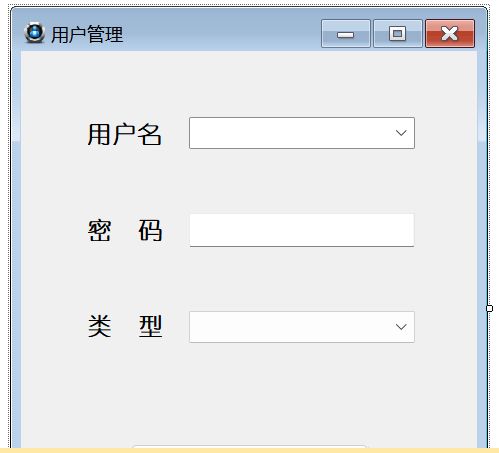
MessageBox.Show("注册成功！");

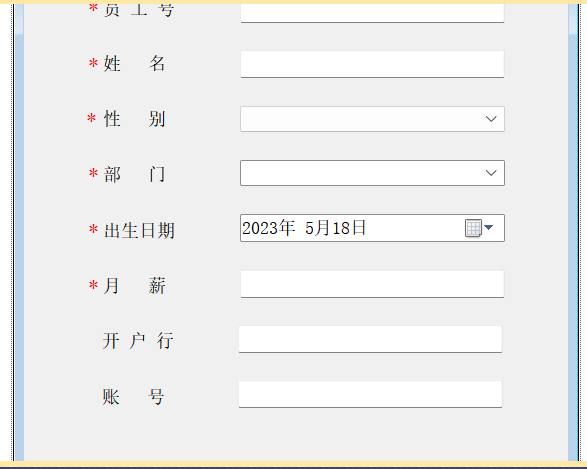
this.Close();

}

}

## 5、2员工管理





### （1）新增员工信息



部分代码;

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

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小?颜?\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

SqlCommand comm = new SqlCommand("Select \* from UserTable where UserName=@UserName", conn);

comm.Parameters.AddWithValue("@UserName", cmbUserName.Text);

SqlDataReader reader = comm.ExecuteReader();

if (reader.Read())

{

txtPassWord.Text = reader["PassWord"].ToString();

cmbType.Text = reader["Type"].ToString();

}

//-----------------数据库有修改时，自动更新.

SqlDataAdapter da = new SqlDataAdapter("select distinct UserName from UserTable", conn);

DataTable dt = new DataTable();

da.Fill(dt);

cmbUserName.DisplayMember = "UserName";

cmbUserName.ValueMember = "UserName";

cmbUserName.DataSource = dt;

SqlDataAdapter da1 = new SqlDataAdapter("select distinct Type from UserTable", conn);

DataTable dt1 = new DataTable();

da1.Fill(dt1);

cmbType.DisplayMember = "Type";

cmbType.ValueMember = "Type";

cmbType.DataSource = dt1;

conn.Close();

conn.Close();

}

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

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小?颜?\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

string id = cmbUserName.Text;

string sql = "Select count(\*) from UserTable where UserName='" + id + "'";

SqlCommand cmd = new SqlCommand(sql, conn);

int count = Convert.ToInt32(cmd.ExecuteScalar());

if (count > 0)

{

MessageBox.Show("您添加的数据已经存在，请重新输入!"，"操作提示

");

}

else {

### （2）修改员工信息



部分代码：

private void OwnFindEmployee(string id)

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

SqlCommand comm = new SqlCommand("Select \* from 员工表 where 员工号='" + id + "'", conn);

conn.Open();

SqlDataReader reader = comm.ExecuteReader();

if (reader.Read())

{

txtID.Text = id;

txtName.Text = reader["姓名"].ToString();

cmbSex.Text = reader["性别"].ToString();

cmbPart.Text = reader["部门"].ToString();

dateTimePicker1.Value = (DateTime)reader["出生日期"];

txtWages.Text = reader["月薪"].ToString();

txtBankName.Text = reader["开户行"].ToString();

txtAccount.Text = reader["账号"].ToString();

}

conn.Close();

}

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

{

OwnFindEmployee(txtID.Text);

}

}

}

### （3）查询员工信息



## 5、3工资管理



### （1）修改工资



部分代码：

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

{

if (txtEmployeeID.Text == "" || txt计?薪?年ê月?.Text == "" || txt月?薪?.Text == ""||txt实害?发ぁ?工¤资哩?Text=="")

{

MessageBox.Show(""请至少将带“\*”部分补充完整!"，"操作提示

");

}

else

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小?颜?\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

string temp = txtEmployeeID.Text;

string SQL = "Select count(\*) from员工表 where员工号='” + temp +"’

SqlCommand cmd1 = new SqlCommand(SQL, conn);

int count1 = Convert.ToInt32(cmd1.ExecuteScalar());

if (count1 == 0)

{

MessageBox.Show("您输入的员工号不存在，请重新输入!"，"操作提示")；

}

else

{

### （2）添加工资信息



部分代码;

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

{

if (txtEmployeeID.Text == "" || txt计薪年月.Text == "" || txt月薪.Text == ""||txt实发工资.Text=="")

{

MessageBox.Show("请至少将带“ \* ”部分补充完整！", "操作提示");

}

else

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

string temp = txtEmployeeID.Text;

string SQL = "Select count(\*) from 员工表 where 员工号='" + temp + "'";

SqlCommand cmd1 = new SqlCommand(SQL, conn);

int count1 = Convert.ToInt32(cmd1.ExecuteScalar());

if (count1 == 0)

{

MessageBox.Show("您输入的员工号不存在，请重新输入！", "操作提示");

}

else

{

SqlCommand comm = new SqlCommand("insert into 工资表(员工号,计薪年月,发放日期,月薪,加班费,节假日补贴,高温补贴,住房补贴,交通补贴,其他加项,所得税,公积金,社会保险,其他扣资,实发工资,制表人,制表时间,审核人,审核时间,备注) values(@员工号,@计薪年月,@发放日期,@月薪,@加班费,@节假日补贴,@高温补贴,@住房补贴,@交通补贴,@其他加项,@所得税,@公积金,@社会保险,@其他扣资,@实发工资,@制表人,@制表时间,@审核人,@审核时间,@备注)", conn);

comm.Parameters.AddWithValue("@员工号", txtEmployeeID.Text);

comm.Parameters.AddWithValue("@发放日期", dateTime发放日期.Value);

comm.Parameters.AddWithValue("@计薪年月", txt计薪年月.Text);

comm.Parameters.AddWithValue("@月薪", txt月薪.Text);

comm.Parameters.AddWithValue("@加班费", txt加班费.Text);

comm.Parameters.AddWithValue("@节假日补贴", txt节假日补贴.Text);

comm.Parameters.AddWithValue("@高温补贴", txt高温补贴.Text);

comm.Parameters.AddWithValue("@住房补贴", txt住房补贴.Text);

comm.Parameters.AddWithValue("@交通补贴", txt交通补贴.Text);

comm.Parameters.AddWithValue("@其他加项", txt其他加项.Text);

comm.Parameters.AddWithValue("@所得税", txt所得税.Text);

comm.Parameters.AddWithValue("@公积金", txt公积金.Text);

comm.Parameters.AddWithValue("@社会保险", txt社会保险.Text);

comm.Parameters.AddWithValue("@其他扣资", txt其他扣资.Text);

comm.Parameters.AddWithValue("@实发工资", txt实发工资.Text);

comm.Parameters.AddWithValue("@制表人", txt制表人.Text);

comm.Parameters.AddWithValue("@审核人", txt审核人.Text);

comm.Parameters.AddWithValue("@制表时间", dateTime制表时间.Value);

comm.Parameters.AddWithValue("@审核时间", dateTime审核时间.Value);

comm.Parameters.AddWithValue("@备注", richTextBox备注.Text);

comm.ExecuteNonQuery();

frmWagesManagement main = this.Owner.ActiveMdiChild as frmWagesManagement;

if (main != null) main.FillData();

MessageBox.Show("添加成功！");

}

conn.Close();

}

### （3）查询工资信息



部分代码：

private void OwnFindWages(string id)

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

SqlCommand comm = new SqlCommand("Select \* from 工资表 where 流水号='" + id + "'", conn);

conn.Open();

SqlDataReader reader = comm.ExecuteReader();

if (reader.Read())

{

## 5、4考勤管理



### （1）添加考勤记录



部分代码：

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

{

if (txt记录号.Text == "" || txt计薪年月.Text == "" || txt员工号.Text == "" || txt应勤天数.Text == "" || txt实际出勤.Text == "")

{

MessageBox.Show("请至少将带“ \* ”部分补充完整！", "操作提示");

}

else

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 判断是否重复 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

string id = txt记录号.Text;

string sql = "Select count(\*) from 考勤表 where 记录号='" + id + "'";

SqlCommand cmd = new SqlCommand(sql, conn);

int count = Convert.ToInt32(cmd.ExecuteScalar());

if (count > 0)

{

MessageBox.Show("您添加的数据已经存在，请重新输入！", "操作提示");

}

else//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 判断是否重复 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

{

string temp = txt员工号.Text;

string SQL = "Select count(\*) from 考勤表 where 员工号='" + temp + "'";

SqlCommand cmd1 = new SqlCommand(SQL, conn);

int count1 = Convert.ToInt32(cmd1.ExecuteScalar());

if (count1 > 0)

{

MessageBox.Show("您输入的员工号已存在，请重新输入！", "操作提示");

}

else

{

string temp1 = txt员工号.Text;

string SQL1 = "Select count(\*) from 员工表 where 员工号='" + temp1 + "'";

SqlCommand cmd2 = new SqlCommand(SQL1, conn);

int count2 = Convert.ToInt32(cmd2.ExecuteScalar());

if (count2 == 0)

{

MessageBox.Show("您输入的员工号不存在，请重新输入！", "操作提示");

}

else

{

SqlCommand comm = new SqlCommand("insert into 考勤表(记录号,计薪年月,员工号,应勤天数,实际出勤,加班天数,事假天数,病假天数,年假天数,婚葬产假,调休天数,矿工天数,迟到早退,备注) values(@记录号,@计薪年月,@员工号,@应勤天数,@实际出勤,@加班天数,@事假天数,@病假天数,@年假天数,@婚葬产假,@调休天数,@矿工天数,@迟到早退,@备注)", conn);

comm.Parameters.AddWithValue("@记录号", txt员工号.Text);

comm.Parameters.AddWithValue("@计薪年月", txt计薪年月.Text);

comm.Parameters.AddWithValue("@员工号", txt员工号.Text);

comm.Parameters.AddWithValue("@应勤天数", txt应勤天数.Text);

comm.Parameters.AddWithValue("@实际出勤", txt实际出勤.Text);

comm.Parameters.AddWithValue("@加班天数", txt加班天数.Text);

comm.Parameters.AddWithValue("@事假天数", txt事假天数.Text);

comm.Parameters.AddWithValue("@病假天数", txt病假天数.Text);

comm.Parameters.AddWithValue("@年假天数", txt年假天数.Text);

comm.Parameters.AddWithValue("@婚葬产假", txt婚葬产假.Text);

comm.Parameters.AddWithValue("@调休天数", txt调休天数.Text);

comm.Parameters.AddWithValue("@矿工天数", txt矿工天数.Text);

comm.Parameters.AddWithValue("@迟到早退", txt迟到早退.Text);

comm.Parameters.AddWithValue("@备注", richTextBox备注.Text);

comm.ExecuteNonQuery();

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

frmAttendanceManagement main = this.Owner.ActiveMdiChild as frmAttendanceManagement;

if (main != null) main.FillData();

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

MessageBox.Show("添加成功！");

}

}

}

conn.Close();

### （2）修改考勤记录



部分代码：

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

{

if (txt记录号.Text == "" || txt员工号.Text == "" || txt计薪年月.Text == "" || txt实际出勤.Text == "" || txt应勤天数.Text == "")

{

MessageBox.Show("请至少将带“ \* ”部分补充完整！", "操作提示");

}

else

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

conn.Open();

string temp = txt员工号.Text;

string SQL = "Select count(\*) from 考勤表 where 员工号='" + temp + "'";

SqlCommand cmd1 = new SqlCommand(SQL, conn);

int count1 = Convert.ToInt32(cmd1.ExecuteScalar());

if (count1 == 0)

{

MessageBox.Show("您输入的员工号已存在，请重新输入！", "操作提示");

}

else

{

//SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

//conn.Open();

SqlCommand comm = new SqlCommand("update 考勤表 set 计薪年月=@计薪年月,员工号=@员工号,应勤天数=@应勤天数,实际出勤=@实际出勤,加班天数=@加班天数,事假天数=@事假天数,病假天数=@病假天数,年假天数=@年假天数,婚葬产假=@婚葬产假,调休天数=@调休天数,矿工天数=@矿工天数,迟到早退=@迟到早退,备注=@备注 where 记录号=@记录号", conn);

comm.Parameters.AddWithValue("@员工号", txt员工号.Text);

comm.Parameters.AddWithValue("@计薪年月", txt计薪年月.Text);

comm.Parameters.AddWithValue("@员工号", txt员工号.Text);

comm.Parameters.AddWithValue("@应勤天数", txt应勤天数.Text);

comm.Parameters.AddWithValue("@实际出勤", txt实际出勤.Text);

comm.Parameters.AddWithValue("@加班天数", txt加班天数.Text);

comm.Parameters.AddWithValue("@事假天数", txt事假天数.Text);

comm.Parameters.AddWithValue("@病假天数", txt病假天数.Text);

comm.Parameters.AddWithValue("@年假天数", txt年假天数.Text);

comm.Parameters.AddWithValue("@婚葬产假", txt婚葬产假.Text);

comm.Parameters.AddWithValue("@调休天数", txt调休天数.Text);

comm.Parameters.AddWithValue("@矿工天数", txt矿工天数.Text);

comm.Parameters.AddWithValue("@迟到早退", txt迟到早退.Text);

comm.Parameters.AddWithValue("@备注", richTextBox备注.Text);

//----------------------更新数据库 5.17

frmAttendanceManagement main = this.Owner.ActiveMdiChild as frmAttendanceManagement;

if (main != null) main.FillData();

//----------------------更新数据库 5.17

MessageBox.Show("修改成功！");

}

conn.Close();

}

}

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

{

Over();

}

}

}

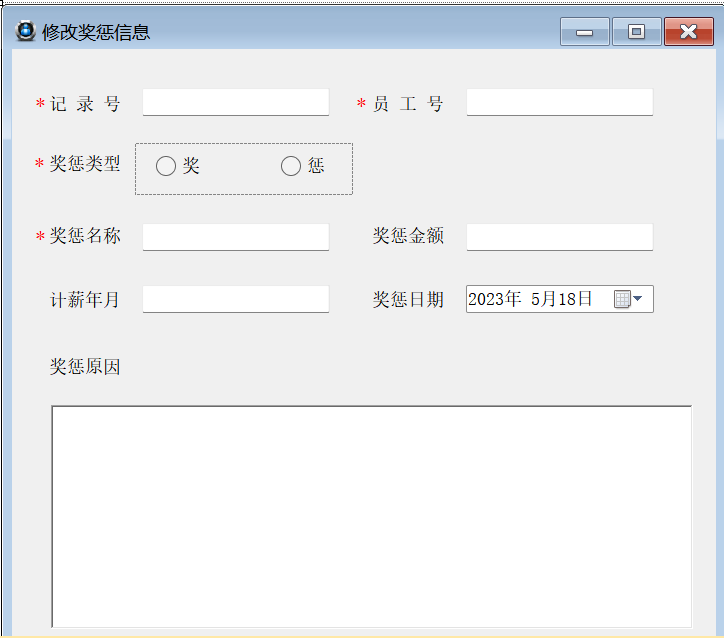
### （3）查询考勤记录



## 5、5奖惩管理



### （1）修改奖惩信息



部分代码;

private void OwnFindReward(string id)

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小颜\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

SqlCommand comm = new SqlCommand("Select \* from 奖惩表 where 员工号='" + id + "'", conn);

conn.Open();

SqlDataReader reader = comm.ExecuteReader();

if (reader.Read())

{

txt员工号.Text = id;

txt计薪年月.Text = reader["计薪年月"].ToString();

txt奖惩金额.Text = reader["奖惩金额"].ToString();

txt奖惩名称.Text = reader["奖惩名称"].ToString();

txt员工号.Text = reader["员工号"].ToString();

richTextBox奖惩原因.Text = reader["奖惩原因"].ToString();

dateTime奖惩日期.Value = (DateTime)reader["奖惩日期"];

if (reader["奖惩类型"].ToString() == "奖")

{

radioButton1.Checked = true;

}

else

{

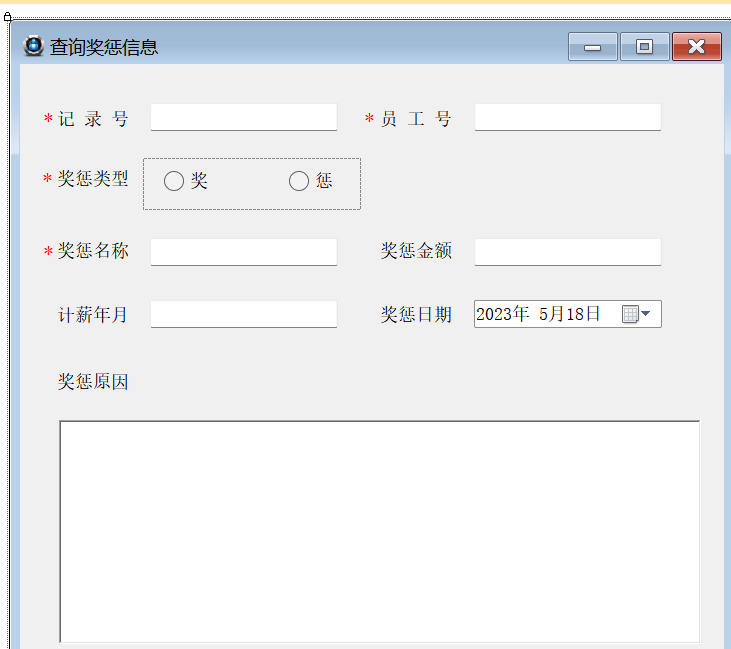
radioButton2.Checked = true;

}

}

}

### （2）查询奖惩信息



private void OwnFindReward(string id)

{

SqlConnection conn = new SqlConnection("MultipleActiveResultSets=True;Server=小?颜?\\SQLEXPRESS;Database=HR;Trusted\_Connection=yes;uid=sa;pwd=677520");

SqlCommand comm = new SqlCommand("Select \* from 奖±惩í表括?where 记?录?号?='" + id + "'", conn);

conn.Open();

SqlDataReader reader = comm.ExecuteReader();

if (reader.Read())

{

txt记?录?号?.Text = id;

txt计?薪?年ê月?.Text = reader["计?薪?年ê月?"].ToString();

txt奖±惩í金e额?.Text = reader["奖±惩í金e额?"].ToString();

txt奖±惩í名?称?.Text = reader["奖±惩í名?称?"].ToString();

txt员±工¤号?.Text = reader["员±工¤号?"].ToString();

richTextBox奖±惩í原-因皑?Text = reader["奖±惩í原-因皑?].ToString();

dateTime奖±惩í日?期ú.Value = (DateTime)reader["奖±惩í日?期ú"];

if (reader["奖±惩í类え?型í"].ToString()=="奖±")

{

radioButton1.Checked = true;

}

else

{

radioButton2.Checked = true;

}

}

conn.Close();

}

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

{

OwnFindReward(txt记?录?号?.Text);

}

}

}

# 程序特点

1.动画登陆界面

2.不同用户类型，进入不同窗口

3.MDI父窗体可以填充背景色，不是默认的灰色

4.做到数据添加或修改时至少填写一部分数据，否则有提示，要求用户输入

5.当修改，新增数据时同时刷新数据

6.添加数据前，先查询数据库，当数据已经存在时提示

7.在员工工资，考勤，奖惩管理中，进行查找时，必须输入员工表存在的数据，不允许非法输入

8.在填写数据时，TextBox中添加完数据后直接回车，即可进入下一文本框输入，快捷高效

9.数据可以导入到Excel中

10.添加了右键快捷菜单，包含了数据的增删改查导出功能和新增的条件查找，及条件查找后显示数据库所有数据

# 七、心得体会

作为一名学习C#的学生，我最近完成了一份人力资源管理系统的课程设计。在这个过程中，我经历了许多挑战和困难，但最终我成功地完成了这个项目，并从中学到了许多值得分享的经验和体会。

首先，我认为设计一个人力资源管理系统需要有良好的项目规划和时间管理。我在开始项目前花了很多时间了解管理系统的需求和功能，以及如何实现这些功能。我按照项目需求建立了一个详细的项目计划，在每个阶段设定了时间表，并遵循这个计划一步步地实现项目。

其次，我认为在设计人力资源管理系统时需要注重用户体验。这意味着要使系统易于使用和理解，以便用户能够快速找到他们需要的信息和功能。我通过添加清晰的标签和菜单来改善系统的导航，同时使用直观的图表和图形来展示数据，这使得用户在使用系统时感觉非常顺畅。

第三，在项目的实现过程中，编写清晰、可读性高的代码也非常重要。良好的代码可以提高程序的可维护性和可扩展性，减少后期的维护工作。因此，我特别重视代码的注释和规范，以及编写易于理解的函数和类，这使得我的代码更加清晰易读。

最后，我认为团队协作也是成功实现一个项目的关键。在整个项目过程中，我始终与我的同学和指导老师保持联系，及时解决问题并寻求合理的建议。这种合作文化不仅增强了项目的结果，也确保了我们共同前进。

总的来说，人力资源管理系统的课程设计让我受益匪浅。通过这个项目，我获得了许多实践经验，提高了自己的技能，并深刻理解了C#编程的重要性。这是一个非常值得学习和探索的领域，我期待今后的更多实践机会。