湖南科技大学计算机科学与工程学院  
《 数据库系统》  
课程设计  
题 目： \_\_\_\_人事管理系统\_\_\_\_专 业： \_\_\_\_物联网工程\_\_\_\_\_\_年 级： \_\_\_\_\_\_2015级\_\_\_\_\_\_\_ 班 级： \_\_\_\_\_一班\_\_\_\_\_\_\_\_\_\_\_ 学 号： \_\_\_\_\_1505040117\_\_\_\_\_ 姓 名： \_\_\_\_\_\_\_李东\_\_\_\_\_\_\_\_\_ 指导老师： \_\_\_\_\_莫尚丰\_\_\_\_\_\_ 完成时间： \_\_\_\_\_2017.6.5\_\_\_\_\_

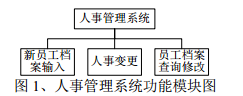
前言

这次我做的是人事管理系统，生产环境是Win10+VS2012+SQL SERVER 2012，与课程设计指导有点出入，除了某些函数不同，做出来的方法和指导书差不多。

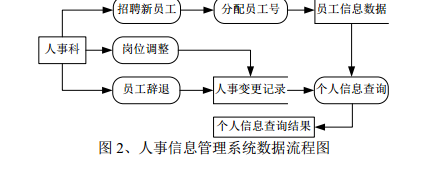
企业人事管理系统主要用于员工个人资料的录入、职务变动的记录和管理。使用人事管理系统，便于公司领导掌握人员的动向，及时调整人才的分配。企业人事管理系统主要用于员工个人资料的录入、职务变动的记录和管理。使用人事管理系统，便于公司领导掌握人员的动向，及时调整人才的分配。

现在在各大中小企业，用计算机管理企业的信息已经越来越普遍了。用计算机不但可以提高工作效率，而且还节省了许多人力物力，增强了企业资料的安全性。提高了企业的管理能力，为此，用计算机来管理企业的信息，是非常必要的。

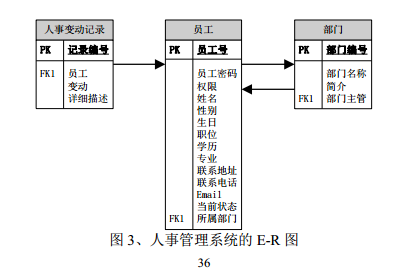
系统需求分析  
1.1 系统功能分析  
人事管理系统主要有以下几项功能要求：  
⚫新员工资料的输入。  
⚫自动分配员工号，并且设置初始的用户密码。  
⚫人事变动的详细记录，包括岗位和部门的调整。  
⚫员工信息的查询和修改，包括员工个人信息和密码等。  
1.2 系统功能模块设计（划分）  
根据系统功能要求，可以将系统分解成几个模块来  
分别设计应用程序界面，如图 1 所示。  
1.3 与其它系统的关系  
本系统是公司管理信息系统的基础部分。为其他子  
系统，如考勤管理系统、工资管理系统和员工培训系统，提供员工的基本信息。同时为其它  
系统提供了员工登录的密码认证和权限分配功能。



1.4 数据流程图  
人事信息管理系统的数据流程如图 2 所示，所有数据由人事科管理人员输入。



**2、数据库设计**



2.1 数据库需求分析  
根据数据流程图，可以列出以下数据项和数据结构：  
⚫员工信息：员工号、密码、权限、姓名、性别、生日、专业、通讯地址、电话、 email、  
当前状态、其他。  
⚫人事变动：记录号、员工、变动、详细描述。  
所需的外部数据支持：  
⚫部门设置：部门编号、名称......  
2.2 数据库概念结构设计  
图 3 是本系统所需数据的 E-R 模型图。

1. 设计内容  
    数据库系统课程设计是为了配合数据库原理及应用开发而设置的，是计算机科学与技术、网络工程、信息安全、物联网工程、软件工程等专业集中实践的教学环节，是将关系数据库理论知识转化为解决实际问题能力的重要环节。数据库系统课程设计目的在于加深对关系数据库理论知识的理解，通过使用具体的 DBMS，掌握一种实际的数据库管理系统并掌握其操作技术，熟练掌握使用数据库前端开发工具（如 VB、 C++、 Java、 Delphi、 PowerBuilder等），进一步提高我们运用数据库技术解决实际问题的能力。设计这个人事管理系统是为了让企业管理企业资料更加方便。所设计的内容包括登录界面，然后根据用户不同的管理权限判断进入不同的菜单。菜单包括管理员界面和普通用户界面。普通用户能够修改自己的个人信息，修改自己密码，查询，管理员除此之外能够进行人事变更，删除记录。
2. 设计方法

用ado连接数据库，用datagrid控件显示数据，利用 Connection 对象的 Execute 方法执行 SQL 命令，直接用 Recordset 对象进行查询取得记录集，不同窗口的切换用新建继承了CDialog 的类，然后在主窗口包括这个类的头文件，用类名.Domodal();切换到类对应的窗口。

1. 创建数据库

这个系统有8个数据表，如下：

表1 PERSON员工个人信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否可空 | 说明 |
| ID | 员工号（主关键字） |  |  |
| PASSWD | 密码 |  |  |
| AUTHORITY | 用户权限 |  |  |
| NAME | 姓名 |  |  |
| SEX | 性别 |  |  |
| BIRTHDAY | 生日 |  |  |
| DEPARTMENT | 所在部门 |  |  |
| JOB | 职务 |  |  |
| EDU\_LEVEL | 受教育程度 |  |  |
| SPCIALTY | 专业技能 |  |  |
| ADDRESS | 家庭住址 |  |  |
| TEL | 联系电话 |  |  |
| EMAIL | 电子邮箱 |  |  |
| STATE | 当前状态（ T-员工、 F-非员工） |  |  |
| REMARK | 备注 |  |  |

表2 PERSONNEL人事变更记录表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否可空 | 说明 |
| ID | 记录编号 |  |  |
| PERSON | 员工号 |  |  |
| CHANGE | 变更代码 |  |  |
| DESCRIPTION | 详细记录 |  |  |

表3 EDU\_LEVEL受教育程度代码表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否可空 | 说明 |
| CODE | 代码 |  |  |
| DESCRIPTION | 描述 |  |  |

表4 JOB职务代码表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否可空 | 说明 |
| CODE | 代码 |  |  |
| DESCRIPTION | 描述 |  |  |

表5 PERSONNEL\_CHANGE人事变动代码表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否可空 | 说明 |
| CODE | 代码 |  |  |
| DESCRIPTION | 描述 |  |  |

表6 DEPARTMENT部门信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否可空 | 说明 |
| ID | 部门编号 |  |  |
| NAME | 部门名称 |  |  |
| MANAGER | 部门经理 |  |  |
| INTRO | 简介 |  |  |

表7 受教育程度代码

|  |  |
| --- | --- |
| 代码 | 说明 |
| 0 | 小学 |
| 1 | 初中 |
| 2 | 高中 |
| 3 | 职高 |
| 4 | 大本 |
| 5 | 大专 |
| 6 | 硕士 |
| 7 | 博士 |
| 8 | 博士后 |

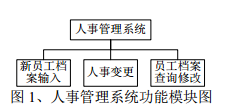
表8 人事变更代码

|  |  |
| --- | --- |
| 代码 | 说明 |
| 0 | 新员工加入 |
| 1 | 职务变动 |
| 2 | 辞退 |

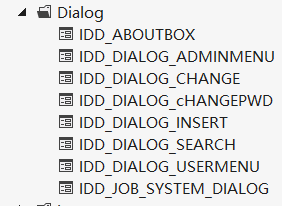
创建这几个表的sql语句如下：

1. 实现方法

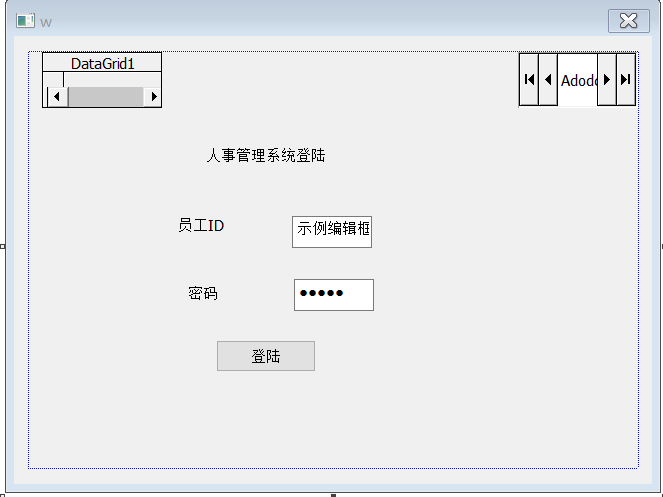
根据系统功能要求，可以将系统分解成几个模块来分别设计应用程序界面



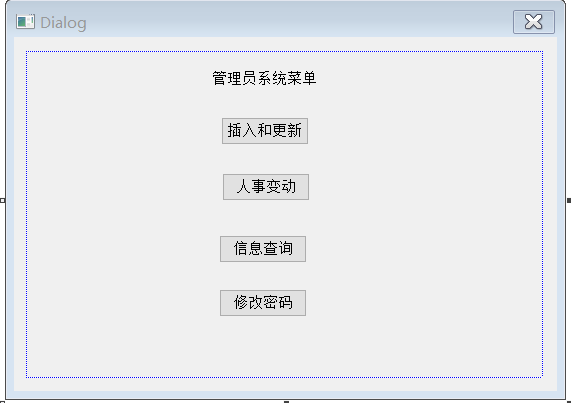
所以我们做了几个Dialog窗口如下



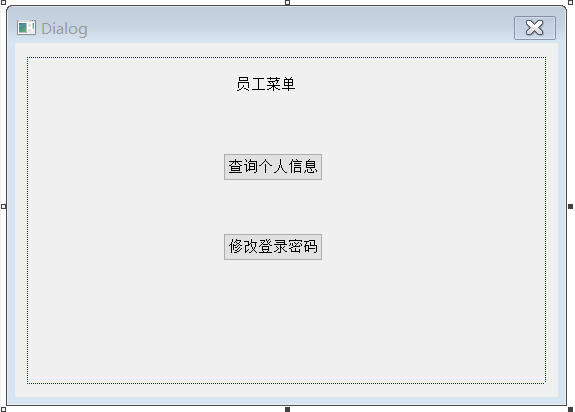
这几个窗口详情如下：



登陆窗口



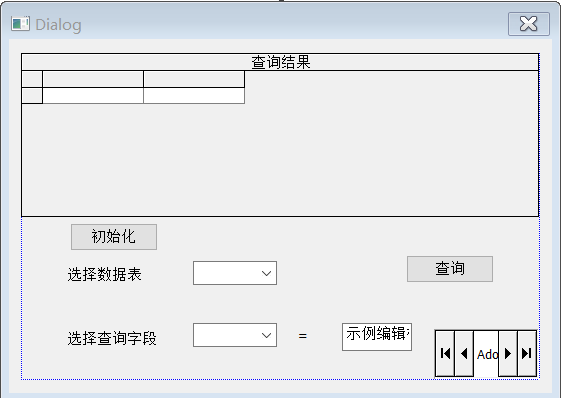
管理员菜单



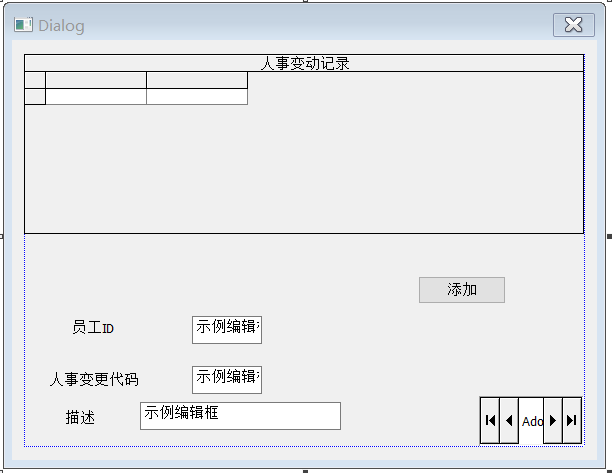
用户菜单



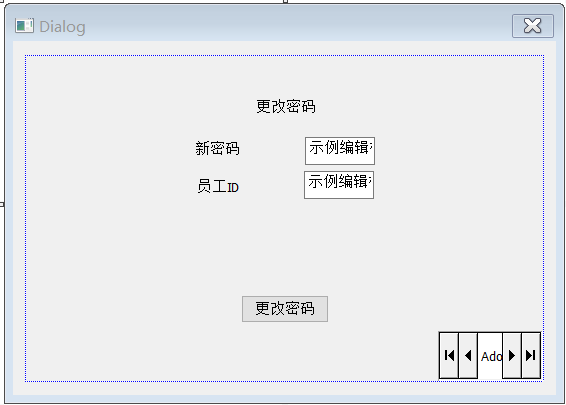
插入删除修改菜单



查询菜单



人事变动菜单



更改密码菜单

这些菜单的启动顺序是

登陆-》（根据权限）菜单-》具体功能。

实现方法就是对按钮增加处理程序，编辑代码，做出不同反应。

1. 算法说明

1、登陆界面部分代码；

void CJob\_SystemDlg::OnClickedButton6()//登陆按钮

{

// TODO: 在此添加控件通知处理程序代码

\_ConnectionPtr m\_pConnection;//连接数据库对象

\_RecordsetPtr m\_pRecordset;//记录集

::CoInitialize(NULL);//初始化

//在 ADO 操作中建议语句中要常用 try„catch()来捕获错误信息

//因为它有时会经常出现一些想不到的错误

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";//连接数据库命令字符串

m\_pConnection->Open(strConn,"","",adModeUnknown);//打开数据库

}

catch (\_com\_error e)

{//若连接不成功显示错误信息

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

UpdateData(TRUE);

\_variant\_t id,auth;//下面是检测有没有输入密码和id相符合的人

\_bstr\_t strAdd1="SELECT \* FROM PERSON WHERE ID='";

strAdd1=strAdd1+m\_s\_id+"' AND PASSWD='";

strAdd1=strAdd1+m\_s\_pwd+"'";

m\_pRecordset=m\_pConnection->Execute(strAdd1,&id,adCmdText);

if(!m\_pRecordset->jgoEOF)//如果记录集中有记录

{

strAdd1=strAdd1+" and AUTHORITY='";//判断是不是管理员

strAdd1=strAdd1+"1'";//1为管理员权限

m\_pRecordset=m\_pConnection->Execute(strAdd1,&auth,adCmdText);

if(!m\_pRecordset->jgoEOF)//是管理员

{

Cadminmenu admin;//要包括Cadminmenu.h

admin.DoModal();//打开管理员窗口

}

else

{//否则打开用户窗口

Cuser user;

user.DoModal();

}

}

else

{

MessageBox("密码或者用户名不正确");

}

}

2、管理员菜单部分代码，（ps：用户界面差不多，就不单独贴出来了）

void Cadminmenu::OnBnClickedButton2()

{//按下第二个按钮打开人事变动界面

// TODO: 在此添加控件通知处理程序代码

Cchange change;

change.DoModal();

}

void Cadminmenu::OnBnClickedButton1()

{//按下第一个按钮打开插入界面

// TODO: 在此添加控件通知处理程序代码

Cinsert insert;

insert.DoModal();

}

void Cadminmenu::OnBnClickedButton3()

{//按下第三个按钮打开搜索界面

// TODO: 在此添加控件通知处理程序代码

Csearch search;

search.DoModal();

}

void Cadminmenu::OnBnClickedButton4()

{//按下第四个按钮打开修改密码界面

// TODO: 在此添加控件通知处理程序代码

Ccpd a;

a.DoModal();

}

3、插入、修改、删除界面部分代码

void Cinsert::OnBnClickedButton1()

{

// TODO: 在此添加控件通知处理程序代码

UpdateData(TRUE);

\_ConnectionPtr m\_pConnection,m\_pConnection1;

\_RecordsetPtr m\_recordset;

::CoInitialize(NULL);

int tmp=1;//计算当前id个数

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";

m\_pConnection->Open(strConn,"","",adModeUnknown);

m\_pConnection1.CreateInstance("ADODB.Connection");

m\_pConnection1->Open(strConn,"","",adModeUnknown);

m\_recordset.CreateInstance(\_uuidof(Recordset));//创建记录集实例

m\_recordset->Open("SELECT \* FROM PERSONNEL",

m\_pConnection1.GetInterfacePtr(),

adOpenStatic, //用静态游标

adLockOptimistic,

adCmdText

);

VARIANT var;

while(!m\_recordset->jgoEOF)//循环直到记录集中没有数据

{

var = m\_recordset->GetCollect("id");

if(var.vt != VT\_NULL)

tmp+=1;

m\_recordset->MoveNext();

}

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

CString id;

id.Format("%d",tmp);//强制类型转换

\_variant\_t ra;//插入数据

\_bstr\_t strAdd = "INSERT INTO PERSON VALUES('";

strAdd=strAdd+id+"','";

strAdd = strAdd +id+ "','";

strAdd = strAdd + m\_s\_authority + "','";

strAdd = strAdd + m\_s\_name + "','";

strAdd = strAdd + m\_s\_sex + "','";

strAdd = strAdd + m\_s\_birthday + "','";

strAdd = strAdd + m\_s\_department + "','";

strAdd = strAdd + m\_s\_job + "','";

strAdd = strAdd + m\_s\_edu\_level + "','";

strAdd = strAdd + m\_s\_spcialty + "','";

strAdd = strAdd + m\_s\_address + "','";

strAdd = strAdd + m\_s\_tel + "','";

strAdd = strAdd + m\_s\_email + "','";

strAdd = strAdd + m\_s\_state + "','";

strAdd = strAdd + m\_s\_remark + "')";

MessageBox(strAdd);

m\_pConnection->Execute(strAdd,&ra,adCmdText);

m\_pConnection->Close();

MessageBox("插入数据成功");

// 以下显示当前数据

m\_adodc1.Refresh();

UpdateData(FALSE);

}

BEGIN\_EVENTSINK\_MAP(Cinsert, CDialog)

ON\_EVENT(Cinsert, IDC\_DATAGRID1, DISPID\_DBLCLICK, Cinsert::OnDblclickDatagrid1, VTS\_NONE)

END\_EVENTSINK\_MAP()

void Cinsert::OnDblclickDatagrid1()

{//双击某行读取数据

// TODO: 在此处添加消息处理程序代码

UpdateData(TRUE);

int ColNum = 2;

CString colVal;

CColumns cols; //要包含头文件"Columns.h"

CColumn col; //要包含头文件"Column.h"

VARIANT v\_ColNum,v\_Value;

cols = m\_datagrid1.get\_Columns();//调用 GetColumns 返回 CDataGrid 对应的 CColumns 对象

v\_ColNum.vt = VT\_I2;

v\_ColNum.iVal = 0;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_id = v\_Value.bstrVal;

v\_ColNum.iVal = 1;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

v\_ColNum.iVal = 2;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_authority = v\_Value.bstrVal;

v\_ColNum.iVal = 3;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_name = v\_Value.bstrVal;

v\_ColNum.iVal = 4;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_sex = v\_Value.bstrVal;

v\_ColNum.iVal = 5;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_birthday = v\_Value.bstrVal;

v\_ColNum.iVal = 6;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_department = v\_Value.bstrVal;

v\_ColNum.iVal = 7;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_job = v\_Value.bstrVal;

v\_ColNum.iVal = 8;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_edu\_level = v\_Value.bstrVal;

v\_ColNum.iVal = 9;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_spcialty = v\_Value.bstrVal;

v\_ColNum.iVal = 10;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_address = v\_Value.bstrVal;

v\_ColNum.iVal = 11;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_tel = v\_Value.bstrVal;

v\_ColNum.iVal = 12;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_email = v\_Value.bstrVal;

v\_ColNum.iVal = 13;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_state = v\_Value.bstrVal;

v\_ColNum.iVal = 14;

col = cols.GetItem(v\_ColNum);

v\_Value = col.GetValue();

m\_s\_remark = v\_Value.bstrVal;

}

void Cinsert::OnBnClickedButton2()

{//更新数据

// TODO: 在此添加控件通知处理程序代码

UpdateData(TRUE);

\_ConnectionPtr m\_pConnection;

::CoInitialize(NULL);

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";

m\_pConnection->Open(strConn,"","",adModeUnknown);

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

//以下插入数据

\_variant\_t ra;

\_bstr\_t strAdd = "UPDATE PERSON SET PERSON.AUTHORITY='";

strAdd = strAdd + m\_s\_authority+ "',PERSON.NAME='";

strAdd = strAdd + m\_s\_name+ "',PERSON.SEX='";

strAdd = strAdd + m\_s\_sex+ "',PERSON.BIRTHDAY='";

strAdd = strAdd + m\_s\_birthday+ "',PERSON.DEPARTMENT='";

strAdd = strAdd + m\_s\_department+ "',PERSON.JOB='";

strAdd = strAdd + m\_s\_job+ "',PERSON.EDU\_LEVEL='";

strAdd = strAdd + m\_s\_edu\_level+ "',PERSON.SPCIALTY='";

strAdd = strAdd + m\_s\_spcialty+ "',PERSON.ADDRESS='";

strAdd = strAdd + m\_s\_address+ "',PERSON.TEL='";

strAdd = strAdd + m\_s\_tel+ "',PERSON.EMAIL='";

strAdd = strAdd + m\_s\_email+ "',PERSON.STATE='";

strAdd = strAdd + m\_s\_state+ "',PERSON.REMARK='";

strAdd = strAdd + m\_s\_remark+ "'";

strAdd = strAdd + " WHERE PERSON.ID='" + m\_s\_id+"'"; /\*这里要注意不能有sql保留字，否则会报Isispatch err3092\*/

MessageBox(strAdd);

try

{

m\_pConnection->Execute(strAdd,&ra,adCmdText); //执行sql语句

m\_pConnection->Close();

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.Description());

return;

}

// 以下显示当前数据

m\_adodc1.put\_RecordSource("SELECT \* FROM PERSON");

m\_adodc1.Refresh();

AfxMessageBox("更新成功");

UpdateData(FALSE);

}

void Cinsert::OnClickedButton3()

{

// TODO: 在此添加控件通知处理程序代码

CString tmp2 = "确认要删除此编号的记录吗？ ";

if(MessageBox(tmp2, \_T("数据删除提示"), MB\_YESNO|MB\_ICONSTOP)!=IDYES)

return;

UpdateData(TRUE);

\_ConnectionPtr m\_pConnection;

::CoInitialize(NULL);

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";

m\_pConnection->Open(strConn,"","",adModeUnknown);

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

//以下插入数据

\_variant\_t ra;

\_bstr\_t strAdd = "DELETE FROM PERSON WHERE ID = '"+m\_s\_id+"'";

MessageBox(strAdd);

try

{

m\_pConnection->Execute(strAdd,&ra,adCmdText);

m\_pConnection->Close();

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.Description());

return;

}

// 以下显示当前数据

m\_adodc1.put\_RecordSource("SELECT \* FROM PERSON");

m\_adodc1.Refresh();

AfxMessageBox("删除成功");

UpdateData(FALSE);

}

void Cinsert::OnBnClickedButton4()

{

// TODO: 在此添加控件通知处理程序代码

CString tmp2 = "确认要删除所有的记录吗？ ";

if(MessageBox(tmp2, \_T("数据删除提示"), MB\_YESNO|MB\_ICONSTOP)!=IDYES)

return;

UpdateData(TRUE);

\_ConnectionPtr m\_pConnection;

::CoInitialize(NULL);

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";

m\_pConnection->Open(strConn,"","",adModeUnknown);

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

//以下插入数据

\_variant\_t ra;

\_bstr\_t strAdd = "DELETE FROM PERSON";

MessageBox(strAdd);

try

{

m\_pConnection->Execute(strAdd,&ra,adCmdText);

m\_pConnection->Close();

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.Description());

return;

}

// 以下显示当前数据

m\_adodc1.put\_RecordSource("SELECT \* FROM PERSON");

m\_adodc1.Refresh();

AfxMessageBox("删除成功");

UpdateData(FALSE);

}

4、人事变更界面部分代码：

void Cchange::OnBnClickedButton1()

{

// TODO: 在此添加控件通知处理程序代码

UpdateData(TRUE);

\_ConnectionPtr m\_pConnection,m\_pConnection1;

\_RecordsetPtr m\_recordset;

::CoInitialize(NULL);

int tmp=1;

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";

m\_pConnection->Open(strConn,"","",adModeUnknown);

m\_pConnection1.CreateInstance("ADODB.Connection");

m\_pConnection1->Open(strConn,"","",adModeUnknown);

m\_recordset.CreateInstance(\_uuidof(Recordset));

m\_recordset->Open("SELECT \* FROM PERSON",

m\_pConnection1.GetInterfacePtr(),

adOpenStatic, //用静态游标

adLockOptimistic,

adCmdText

);

VARIANT var;

while(!m\_recordset->jgoEOF)//记录事件记录id数量

{

var = m\_recordset->GetCollect("id");

if(var.vt != VT\_NULL)

tmp+=1;

m\_recordset->MoveNext();

}

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

CString id;

id.Format("%d",tmp);

variant\_t ra;

\_bstr\_t strAdd = "INSERT INTO PERSONNEL VALUES('";//数据插入

strAdd = strAdd + id + "','";

strAdd = strAdd + m\_s\_id + "','";

strAdd = strAdd + m\_s\_code + "','";

strAdd = strAdd + m\_cg\_description + "')";

m\_pConnection->Execute(strAdd,&ra,adCmdText);

m\_pConnection->Close();

m\_adodc1.put\_RecordSource("SELECT \* FROM personnel");

m\_adodc1.Refresh();

UpdateData(FALSE);

}

5、查询界面部分代码

由于不知道放在哪里初始化conbo的值，只能单独做一个初始化按钮，专门按一下来初始化。

void Csearch::OnBnClickedButton2()

{//初始化按钮

// TODO: 在此添加控件通知处理程序代码

m\_combo1\_TabelName.SetWindowText("JOB");//默认显示JOB

m\_combo2\_FieldName.SetWindowText("CODE");

m\_combo1\_TabelName.AddString("PERSON");

m\_combo1\_TabelName.AddString("PERSONNEL");

m\_combo1\_TabelName.AddString("PERSONNEL\_CHANGE");

m\_combo1\_TabelName.AddString("DEPARTMENT");

m\_combo1\_TabelName.AddString("EDU\_LEVEL");

m\_combo1\_TabelName.AddString("AUTHORITY");

m\_combo2\_FieldName.AddString("CODE");

m\_combo2\_FieldName.AddString("DESCRIPTION");

}

void Csearch::OnClickedButton1()

{//查询按钮

// TODO: 在此添加控件通知处理程序代码

UpdateData(TRUE);

CString tmpStr, strSQL = "SELECT \* FROM ";

m\_combo1\_TabelName.GetWindowText(tmpStr);

strSQL += tmpStr+" WHERE ";

m\_combo2\_FieldName.GetWindowText(tmpStr);

strSQL += tmpStr+" = ";

strSQL += "'"+m\_Edit1\_FieldValue+"'";

m\_adodc1.put\_RecordSource(strSQL);

m\_adodc1.Refresh();

UpdateData(FALSE);

}

void Csearch::OnKillfocusCombo1()

{//第一个下拉栏按钮

// TODO: 在此添加控件通知处理程序代码

CString tableName;

m\_combo1\_TabelName.GetWindowText(tableName);

m\_combo2\_FieldName.ResetContent();

if(tableName=="PERSON")//根据表名不同显示不同列名

{

m\_combo2\_FieldName.AddString("ID");

m\_combo2\_FieldName.AddString("AUTHORITY");

m\_combo2\_FieldName.AddString("NAME");

m\_combo2\_FieldName.AddString("SEX");

m\_combo2\_FieldName.AddString("BIRTHDAY");

m\_combo2\_FieldName.AddString("DEPARTMENT");

m\_combo2\_FieldName.AddString("JOB");

m\_combo2\_FieldName.AddString("EDU\_LEVEL");

m\_combo2\_FieldName.AddString("SPCIALTY");

m\_combo2\_FieldName.AddString("ADDRESS");

m\_combo2\_FieldName.AddString("TEL");

m\_combo2\_FieldName.AddString("EMAIL");

m\_combo2\_FieldName.AddString("STATE");

m\_combo2\_FieldName.AddString("REMARK");

}

else if(tableName=="PERSONNEL")

{

m\_combo2\_FieldName.AddString("ID");

m\_combo2\_FieldName.AddString("PERSON");

m\_combo2\_FieldName.AddString("CHANGE");

m\_combo2\_FieldName.AddString("DESCRIPTION");}

else if(tableName=="DEPARTMENT")

{

m\_combo2\_FieldName.AddString("ID");

m\_combo2\_FieldName.AddString("NAME");

m\_combo2\_FieldName.AddString("MANAHER");

m\_combo2\_FieldName.AddString("INTRO");}

else if(tableName=="JOB")

{

m\_combo2\_FieldName.AddString("CODE");

m\_combo2\_FieldName.AddString("DESCRIPITION");}

else if(tableName=="EDU\_LEVEL")

{

m\_combo2\_FieldName.AddString("CODE");

m\_combo2\_FieldName.AddString("DESCRIPTION");

}

else if(tableName=="AUTHORITY")

{

m\_combo2\_FieldName.AddString("CODE");

m\_combo2\_FieldName.AddString("DESCRIPTION");

}

else

{

m\_combo2\_FieldName.AddString("CODE");

m\_combo2\_FieldName.AddString("DESCRIPTION");

}

}

6、修改密码界面部分密码

void Ccpd::OnBnClickedButton1()

{

// TODO: 在此添加控件通知处理程序代码

UpdateData(TRUE);

\_ConnectionPtr m\_pConnection;

::CoInitialize(NULL);

//在 ADO 操作中建议语句中要常用 try„catch()来捕获错误信息

//因为它有时会经常出现一些想不到的错误

try

{

m\_pConnection.CreateInstance("ADODB.Connection");

\_bstr\_t

strConn="Provider=SQLOLEDB;Server=DESKTOP-R21ACAN;DATABASE=Job\_Manage\_System;UID=sa;PWD=lidong;";

m\_pConnection->Open(strConn,"","",adModeUnknown);

}

catch (\_com\_error e)

{

::CoUninitialize();

AfxMessageBox(e.ErrorMessage());

return;

}

\_variant\_t ra;

if(m\_s\_id.IsEmpty()||m\_Npd.IsEmpty())

{

MessageBox("旧密码或者员工id错误或者为空");

}

else

{

\_bstr\_t strAdd = "UPDATE PERSON SET PASSWD='";

strAdd=strAdd+m\_Npd+"'";

strAdd=strAdd+" where ID='";

strAdd=strAdd+m\_s\_id+"'";

//MessageBox(strAdd);

m\_pConnection->Execute(strAdd,&ra,adCmdText);

MessageBox("成功更改密码!");

}

UpdateData(FALSE);

}

1. 心得与体会

我是在我自己电脑上做完的，用的是vc2012，vc2012和vc6.0差别还是有的，花了比较多的时间适应vc2012，原因是实在不想用这个古董软件了，数据库使用的是sql server 2012，很好用，没出什么大问题。刚开始编程时ado和datagrid都找不到，但是加载了类之后就可以使用了。途中还遇到了类未注册的情况，用谷歌英文搜索找到了解决方案，注册MSVCRTD.DLL和MFC42D.DLL就可以了。执行sql语句的时候善于使用try{}catch{e.Description()}能看到语句有什么错误。有错误善于搜索，请教他人。