最小白的C#模拟QQ实现

## 使用sql执行创建数据库

C:\Users\Administrator\Desktop\2019-04-12_091229.png

SQL代码例子

F:\Documents\Scrshot\2019-04-12_091406.png

核心代码如下

C:\Scrshot\2019-04-12_093127.png

public object ExSca(string sqlText)//返回结果查询的第一行第一列

{

if (conn.State == System.Data.ConnectionState.Open)

conn.Close();

conn.Open();

SqlCommand comd = new SqlCommand(sqlText, conn);

object o = comd.ExecuteScalar();

conn.Close();

return o;

}

public int ExNonQuary(string sqlText)//执行非查询（增删改）操作，返回

{

if (conn.State == System.Data.ConnectionState.Open)

conn.Close();

conn.Open();

SqlCommand comd = new SqlCommand(sqlText, conn);

int count = comd.ExecuteNonQuery();

conn.Close();

return count;

}

public SqlDataReader ExcDataReader(string sqlText)//返回查询的第一行

{

if (conn.State == System.Data.ConnectionState.Open)

conn.Close();

conn.Open();

SqlCommand comd = new SqlCommand(sqlText, conn);

SqlDataReader dr = comd.ExecuteReader();

//conn.Close();

return dr;

}

C:\Scrshot\2019-04-12_093755.png

## 登陆窗口



//登陆功能核心代码

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

{

if (textBox1.Text == "" || textBox2.Text == "")

{

//提示为空

MessageBox.Show("您输入为空", "提示",MessageBoxButtons.OK,MessageBoxIcon.Information);

}

SQLHelper s = new SQLHelper();

string strSql = string.Format("select \* from qq\_User where userId={0} and userPwd={1}",textBox1.Text,textBox2.Text);

object cont = s.ExSca(strSql);

if (cont != null)

{

//登陆成功

main main1 = new main(textBox1.Text);

main1.Show();

}

else

{

MessageBox.Show("您输入的用户名密码错误", "提示", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

//conn.Close();

}

## 注册窗口



//核心代码

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

{

SQLHelper s = new SQLHelper();

string strSql = string.Format("insert into qq\_User(userPwd,nickName,age,sex,realName,email,personalMsg,regTime,face) values('{0}','{1}','{2}','{3}','{4}','{5}','{6}','{7}','{8}') select @@IDENTITY", textBox3.Text, textBox1.Text, textBox2.Text, radioButton2.Text, textBox5.Text, textBox6.Text, textBox7.Text, DateTime.Now.ToLongTimeString(),Convert.ToInt16(fn));

MessageBox.Show("恭喜您注册成功！QQ号码为" + Convert.ToInt16(s.ExSca(strSql)));

}

存在的bug ：

C:\Scrshot\2019-04-12_094607.png

Now.ToLongTimeString()传递的是一个短的日期，但是数据默认长日期，所以会自动补1900默认日期



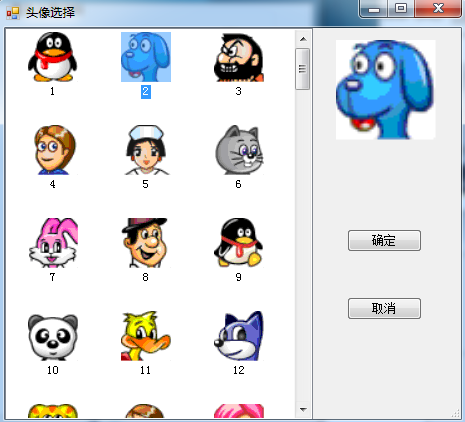
解决方案：

C:\Scrshot\2019-04-12_102230.png

代码更改为DateTime.Now.ToString()，时间正常

C:\Scrshot\2019-04-12_102220.png

头像选择



在register中，将pictureBox1传递给Face窗体  
private void pictureBox1\_Click(object sender, EventArgs e)

{

//打开头像选择窗体

//String fn="1";

Face f = new Face(pictureBox1);

f.ShowDialog();

if (f.DialogResult == DialogResult.OK)

{

fn=f.str;

}

main m = new main();

pictureBox1.Image = m.imageList1.Images[Convert.ToInt32(fn) - 1];

}

//Face中显示可选择的头像库

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

{

main m=new main();

listView1.LargeImageList=m.imageList1;

for (int i = 0; i < 100; i++)

{

ListViewItem lvi = new ListViewItem();

lvi.ImageIndex = i;

lvi.Text = (i+1).ToString();

listView1.Items.Add(lvi);

}

}

//改造构造函数，Face中

public Face(PictureBox p1)

{

this.pictureBox1 = p1;

InitializeComponent();

}

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

{

str = this.listView1.SelectedItems[0].Text;

this.DialogResult = DialogResult.OK;

this.Close();

}

主页面



public void getUserInfo()

{

//加载个人信息

//User User = new User();

User.UserId = userid;

SQLHelper s = new SQLHelper();

SqlDataReader dr=s.ExcDataReader("select \* from qq\_User where userId=" +userid);

dr.Read();//通过read方法一次读取一行数据

User.UserPwd = dr[1].ToString();

User.NickName = dr[2].ToString();

User.RealName=dr[3].ToString();

User.PersonalMsg=dr[4].ToString();

User.Age = Convert.ToInt32(dr[5]);

User.Sex = dr[6].ToString();

User.Email = dr[7].ToString();

User.Face = Convert.ToInt32(dr[8]);

label1.Text = User.NickName;

label2.Text = User.PersonalMsg;

pictureBox1.Image = imageList1.Images[User.Face-1];

}

public void getFriendList()

{

//创建分组

ListViewGroup lvg1 = new ListViewGroup();

lvg1.Header = "好友";

ListViewGroup lvg2 = new ListViewGroup();

lvg2.Header = "陌生人";

ListViewGroup lvg3 = new ListViewGroup();

lvg3.Header = "黑名单";

listView1.Groups.Add(lvg1);

listView1.Groups.Add(lvg1);

listView1.Groups.Add(lvg1);

SQLHelper s = new SQLHelper();

string sql = string.Format("select \* from qq\_Friends where hostId='{0}' ",userid);

SqlDataReader sdr = s.ExcDataReader(sql);

while (sdr.Read())

{

SQLHelper s1=new SQLHelper();

string sqlSelect="select \* from qq\_User where userId="+sdr[2].ToString();

User friend=new User();

SqlDataReader dr=s1.ExcDataReader(sqlSelect);

dr.Read();

ListViewItem lvi=new ListViewItem();

lvi.Name=dr[0].ToString();//将好友ID赋值给Name

lvi.ImageIndex=Convert.ToInt32(dr["face"]);//头像符号

lvi.Text=dr["nickName"].ToString();

listView1.Items.Add(lvi);

if (Convert.ToInt32(sdr[3]) == 0)//陌生人

lvi.Group = listView1.Groups[0];

else if (Convert.ToInt32(sdr[3]) == 1)//好友

lvi.Group = listView1.Groups[0];

else

lvi.Group = listView1.Groups[2];//黑名单

}

}

存在bug：每次添加好友会发生重叠

解决方案：添加listView1.Items.Clear();每次读取好友库前清空一下空间

修改个人信息



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

{

SQLHelper s = new SQLHelper();

string strSql = string.Format("update qq\_User set nickName='{0}',personalMsg='{1}',sex='{2}' where userId='{3}'", textBox1.Text, textBox2.Text, radioButton2.Text,Convert.ToInt32(MainUserID));

MessageBox.Show("修改代码成功返回1，失败0：" + Convert.ToInt16(s.ExNonQuary(strSql)));

this.DialogResult = DialogResult.OK;

this.Close();

}

# 关于显示新窗口的两种方式

|  |  |
| --- | --- |
| ShowDialog(); | Show (); |
| 在新窗口执行完毕后，再继续执行本代码段 | 继续执行本代码段后再执行新窗口 |
| 模态显示（showdialog） | 非模态显示（show） |

**区别**：

    模态与非模态窗体的主要区别是窗体显示的时候是否可以操作其他窗体。模态窗体不允许操作其他窗体，非模态窗体可以操作其他窗体。

**模态显示后，**  
弹出窗口阻止调用窗口的所有消息响应。  
只有在弹出窗口结束后调用窗口才能继续。  
在模态窗口“关闭”后，可以读取模态窗口中信息，包括窗口的返回状态，窗口子控件的值。

在调用Form.ShowDialog方法后,直到关闭对话框后，才执行此方法后面的代码

窗体显示为模式窗体时，单击“关闭”按钮会隐藏窗体，并将DialogResult属性设置为DialogResult.Cancel    
与无模式窗体不同，当用户单击对话框的关闭窗体按钮或设置DialogResult属性的值时,不调用窗体的Close方法    
实际上是把窗体的Visible属性赋值为false,隐藏窗体了    
这样隐藏的窗体是可以重新显示，而不用创建该对话框的新实例    
因为未关闭窗体,所以在应用程序不再需要该窗体时,请调用该窗体的Dispose方法

所以模态窗口在关闭时，不会调用close方法，也不调用dispose方法，窗口仍然存在，占有资源，所以可以继续获得窗口相关信息，在窗口不再使用时，需要手动释

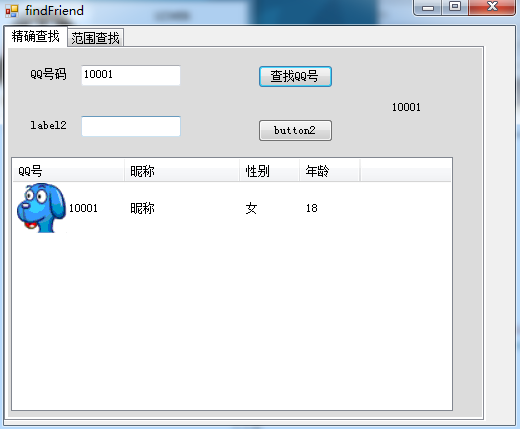
**非模态显示后，**  
可以在弹出窗口和调用窗口之间随意切换。  
调用窗口调用show方法后，下面的代码可以立即执行。  
在非模态窗口关闭后，窗口的所有资源被释放，窗口不存在，无法获取窗口的任何信息。

怎么判断一个窗体是模式窗体呢？    
利用Form.Modal属性,如果该窗体是模式显示，则为true,否则为false    
根据通过Show和ShowDialog而显示出来的窗体的Modal属性分别对应false和true    
特别注意：    
由于在窗体创建之前是无法得知显示方式的,所以在窗体构造函数中,Modal属性总是对应false,所以我们只能在Load事件中或者之后利用Modal属性值

怎么确定窗体间的所有者关系?    
Form类的Owner属性:窗体的所有者    
当一个窗体归另一窗体所有时，它便随着所有者窗体最小化和关闭。    
例如，如果Form2归窗体Form1所有，则关闭或最小化Form1时，Form2也会关闭或最小化。

——来源自网络博客

添加好友



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

{

if (textBox1.Text != "")

{

//查找用户信息

string sql = "select \* from qq\_User where userId=" + textBox1.Text.Trim();//Trim去除前后空格

SQLHelper s = new SQLHelper();

SqlDataReader sdr = s.ExcDataReader(sql);

sdr.Read();

//显示在下方listview中

ListViewItem lvi = new ListViewItem();//QQ号码

label3.Text = sdr["userId"].ToString();

lvi.Text = sdr["userId"].ToString();

lvi.ImageIndex = Convert.ToInt32(sdr["face"]);//头像

lvi.SubItems.Add(sdr["nickName"].ToString());//昵称

lvi.SubItems.Add(sdr["sex"].ToString());//性别

lvi.SubItems.Add(sdr["age"].ToString());//年龄

this.listView1.Items.Add(lvi);

s.closeConn();

}

}

private void listView1\_DoubleClick(object sender, EventArgs e)

{

//添加好友

string friendID=listView1.SelectedItems[0].Text;

label1.Text = mainUserId;

label2.Text = friendID;

string sql = string.Format("insert into qq\_Friends(hostId,friendId,state) values('{0}','{1}',1)", mainUserId,friendID);

SQLHelper s = new SQLHelper();

if (s.ExNonQuary(sql) > 0)

{

MessageBox.Show("恭喜，好友添加成功！", "提示");

}

}

private void button3\_Click(object sender, EventArgs e)

{

//查找符合条件的所有用户，显示在下方的dataGridView1中

string sql = "select \* from qq\_User where ";

string age = "";

string sex = "(sex='男' or sex='女')";

if (radioButton1.Checked || radioButton2.Checked)

{

sex = radioButton1.Checked ? "sex='男'" : "sex='女'";

}

if (listBox1.SelectedItems.Count > 0)

{

int index=listBox1.SelectedIndex;

switch (index)

{

case 0: age="age<10";break;

case 1: age = "age>=10 and age<20"; break;

case 2: age = "age>=20 and age<30"; break;

case 3: age = "age>=30 and age<40"; break;

case 4: age = "age>=40 and age<50"; break;

case 5: age = "age>=50"; break;

}

age = " and " + age;

}

sql = sql + sex + age;

SQLHelper s = new SQLHelper();

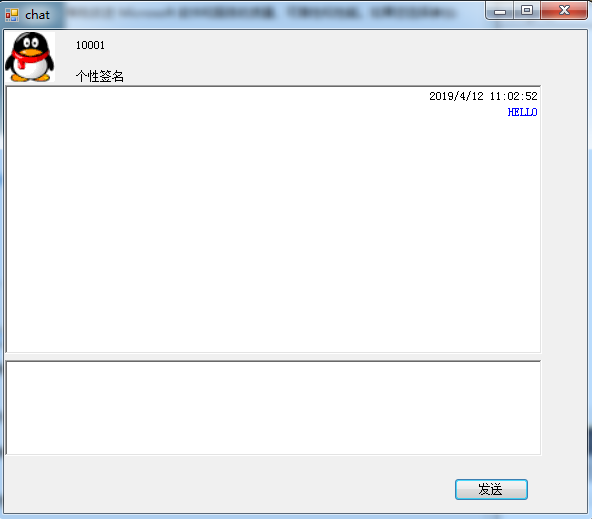
DataSet ds=s.ExcDataSet(sql);

dataGridView1.DataSource = ds.Tables[0];

dataGridView1.Columns[1].Visible = false;

}

好友消息聊天



private void SendMessage()

{

//把消息加入上文本框（注意格式）

richTextBox1.SelectionAlignment = HorizontalAlignment.Right;

DateTime d = DateTime.Now;

richTextBox1.AppendText(d.ToString());

richTextBox1.AppendText(Environment.NewLine);

richTextBox1.SelectionColor = Color.Blue;

richTextBox1.AppendText(richTextBox2.Text);

richTextBox1.AppendText(Environment.NewLine);

//把消息传入数据库

string sql3 = string.Format("insert into qq\_Messages(fromId,toId,mInfo,state,sendTime) values('{0}','{1}','{2}',0,'{3}')",MainUserID,ChatFriendID,richTextBox2.Text,d);

SQLHelper s3 = new SQLHelper();

s3.ExNonQuary(sql3);

richTextBox2.Text = "";

}

private void LoadMessage()

{

string sql = string.Format("select \* from qq\_Messages where fromId='{0}' and toId='{1}' and state=0",ChatFriendID,MainUserID);

SQLHelper s = new SQLHelper();

SqlDataReader sdr = s.ExcDataReader(sql);

while (sdr.Read())//读取消息

{

//读取时间

richTextBox1.SelectionColor = Color.Black;

richTextBox1.AppendText(sdr["sendTime"].ToString());

richTextBox1.AppendText(Environment.NewLine);

//消息

richTextBox1.SelectionColor = Color.Green;

richTextBox1.SelectionFont=new Font(richTextBox1.Font.FontFamily,20);

richTextBox1.AppendText(sdr["mInfo"].ToString());

richTextBox1.AppendText(Environment.NewLine);

//修改消息状态

string sql1=string.Format("update qq\_Messages set state=1 where fromId='{0}' and toId='{1}'",ChatFriendID,MainUserID);

SQLHelper s1 = new SQLHelper();

s1.ExNonQuary(sql1);

}

}

private void LoadFriendInfo()

{

string sql = "select \* from qq\_User where userId="+ChatFriendID;

SQLHelper s = new SQLHelper();

SqlDataReader sdr=s.ExcDataReader(sql);

sdr.Read();

label1.Text = sdr[2].ToString();

label2.Text = sdr["personalMsg"].ToString();

main m=new main();

pictureBox1.Image=m.imageList1.Images[Convert.ToInt32(sdr["face"])];

}