

实验二名称：综合编程实验

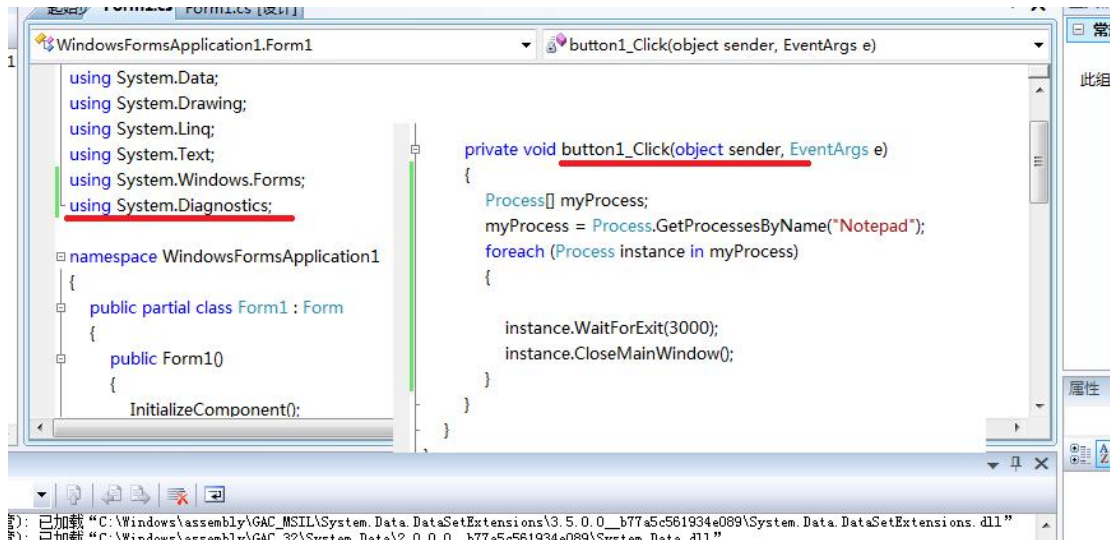
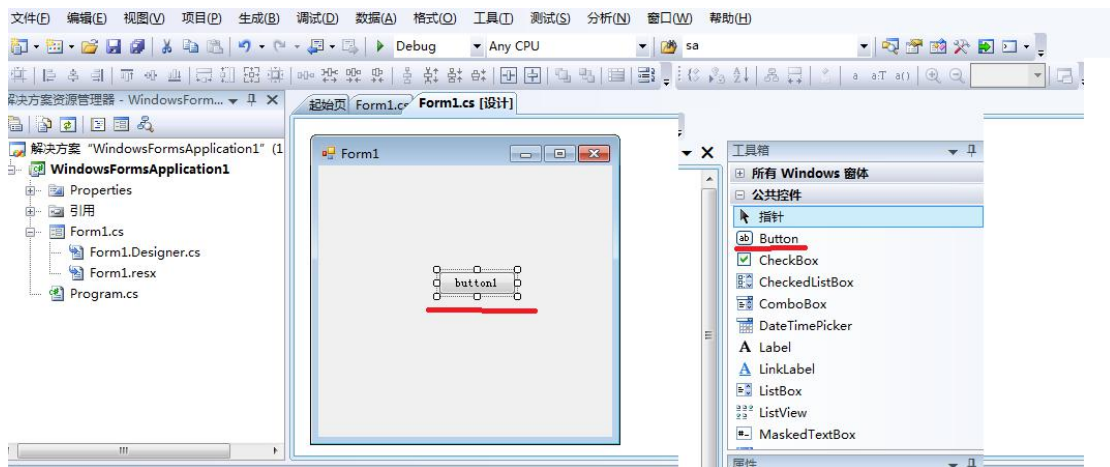
任务二-1（可视化与对象编程，下例为 C#或 C++，每题需有扩展部分）

[题目一]下面是关闭特定程序的代码，其中红色部分为核心代码。

(1) 请在 VS 里面建立一个项目，然后在 Form 里放置一个按钮控件

(2) 双击按钮控件，在合适的部分加入下面红色的代码

(3) 拓展：进一步修改程序，给出一个列表显示正在运行的程序，让用户选择关闭哪个程序。



```
using System;
using System.Diagnostics;
class close_special_exe
{
    static void Main()
    {
        Process[] myProcess;
        myProcess=Process.GetProcessesByName ("Notepad");
```

```
foreach(Process instance in myProcess)
{

    instance.WaitForExit (3000);
    instance.CloseMainWindow ();
}
}
```

[题目二]判断是否为汉字，改写程序，在编辑框里输入一个字符，判断是否是汉字。

单字节的字符一般在0~127之间，而双字节的汉字一般在128~255之间。

拓展：（也可以用正则表达式来实现判断）+修改错误

正则表达式：

http://www.360doc.com/content/12/0225/13/19147_189500941.shtml

<http://www.cnblogs.com/minlecun/archive/2011/12/06/2277399.html>

当然，下面的代码是有问题的：

比如输入：i 我是汉字，或者我是1个汉字，答案都是错的



Form1

输入:
我是汉子

判断结果:
输入的全是汉字

检测

```
private void button1_Click(object sender, EventArgs e)
{
    string String1;
    String1 = textBox1.Text;
    bool ischinese = IsChina(String1);
    if (ischinese) textBox2.Text = "输入的全是汉字";
    else textBox2.Text = "输入的包含其他字符";
}

public bool IsChina(string CString)
{
    bool BoolValue=false;
    for (int i=0; i<CString.Length; i++)
    {
        if(Convert.ToInt32(Convert.ToChar(CString.Substring(i,1)))<Convert.ToInt32(Convert.ToChar(128)))
```

```
public bool IsChina(string CString)
{
    bool BoolValue=false;
    for (int i=0; i<CString.Length; i++)
    {
```

```
        if(Convert.ToInt32(Convert.ToChar(CString.Substring(i,1)))<Convert.ToInt32(Convert.ToChar(128)))
        {
            BoolValue = false;
        }
        else
        {
            BoolValue = true;
        }
    }
    return BoolValue;
}
```

[题目三]密码验证：模拟登陆界面，判别用户名和密码，给出合适的提示，如果超过三次，锁定输入。

拓展 1：用*代替密码；或者最新输入显示，前面的变成*；安全性措施。

扩展 2：如下图



[题目四]浏览器：利用 VC++.net 里的浏览器控件实现浏览器，除以下步骤外，请参经常用的浏览器扩充功能。不少浏览器在发展初期都是利用 IE 的内核来实现的。

拓展：各种浏览器功能的实现。

步骤：

1、在 Form 上放置 webbrowser、button、textbox 控件

2、利用 panel 控件（容器）来帮助安排界面

容器是什么：容易就是一类控件，用来放其他控件，以使得界面整洁有序

比如：



3、利用控件的各种属性来设定控件的位置符合需要

4、Button 点击后，操作：

```
webBrowser1.Navigate(textBox1.Text);
```

```
//下面为 C++代码，可参考：
```

```
//webBrowser1->Navigate( this->textBox1->Text );
```



[题目五]：为游戏做准备

1、建议在 C#下使用 GDI+

2、MFC 或 C++.Net 下用起来比较麻烦，网上 MFC 的例子较多

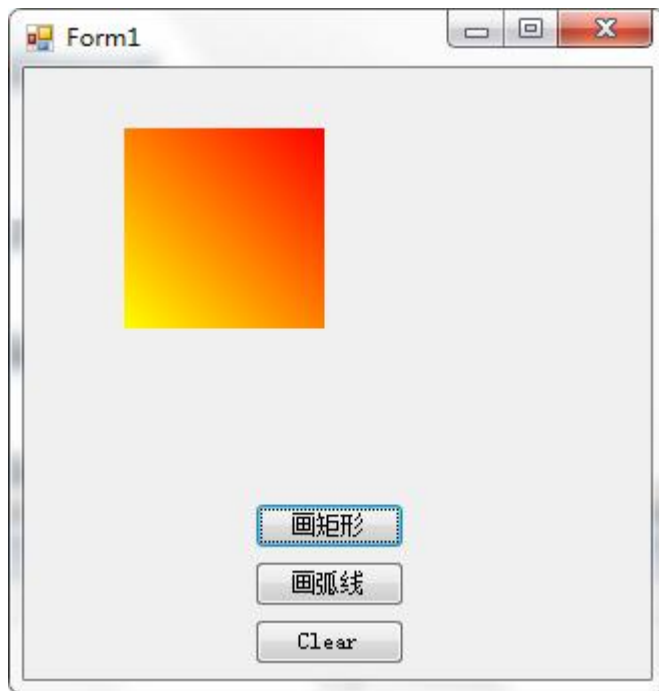
3、主要的类 Graphics 参见 MSDN：

[http://msdn.microsoft.com/zh-cn/library/system.drawing.graphics\(v=vs.110\).aspx?cs-save-lang=1&cs-lang=csharp#code-snippet-1](http://msdn.microsoft.com/zh-cn/library/system.drawing.graphics(v=vs.110).aspx?cs-save-lang=1&cs-lang=csharp#code-snippet-1)

4、参考下面的例子，进一步拓展其他功能，实现更复杂的图形编程。

步骤一：

如图，在 From 上放 3 个按钮



步骤二：

在按钮事件中添加代码

```
private void button1_Click(object sender, EventArgs e)
{
    Graphics g = this.CreateGraphics();
    Rectangle rect = new Rectangle(50, 30, 100, 100);
    LinearGradientBrush lBrush = new LinearGradientBrush(rect,
Color.Red, Color.Yellow, LinearGradientMode.BackwardDiagonal);
    g.FillRectangle(lBrush, rect);
}

private void button2_Click(object sender, EventArgs e)
{
    Graphics g = this.CreateGraphics();
    Pen pn = new Pen(Color.Blue);
    Rectangle rect = new Rectangle(50, 50, 200, 100);
    g.DrawArc(pn, rect, 12, 84);
}

private void button3_Click(object sender, EventArgs e)
{
    Graphics g = this.CreateGraphics();
    Color color1 = Form1.ActiveForm.BackColor;
    g.Clear(color1);
}
```

}

[题目六-1][小型项目]

[可结合六-2，采用可视化编程实现，此处可以不单独做；如结合并采用可视化完成，可视为扩展]

旅馆信息管理程序 基本要求：

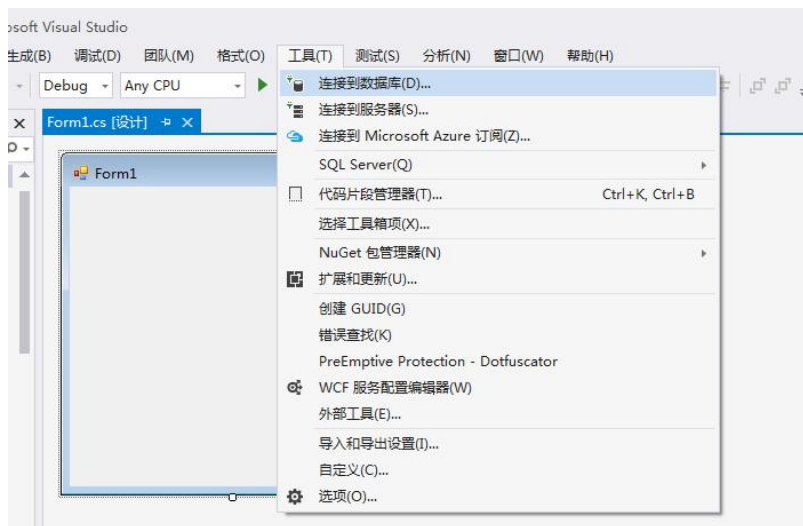
1. 要求实现旅客的入住、换房和退房、查看旅客和房间信息，以及从文件读取相关信息，将相关信息储存到文件的功能，从而实现旅馆信息管理的基本功能。要求记录入住的时间，以及是否已经入住的信息。
2. 可以采用两个结构体分别存储旅客和房间的信息。
3. 实现旅客入住、换房、退房三种行为。
4. 能将文件中的信息读到结构体中，能将结构体的信息保存到文件中。
5. 系统制作完成后应实现类似下图所示界面

```
1 查看旅店信息
2 查看某一房间信息
3 查看旅客信息
4 查找某一旅客信息
5 旅客入住
6 旅客换房
7 旅客退房
8 退出
请选择：
```

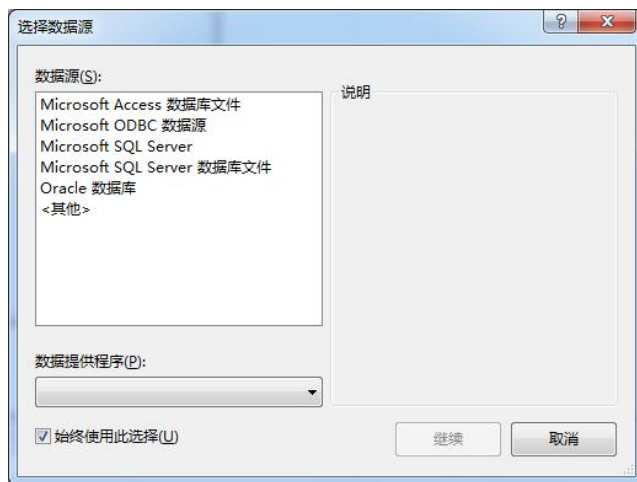
[题目六-2][小型项目]：连接数据库

(一)、手动连接数据库（与下一个的程序连接数据库是独立的 2 个部分）

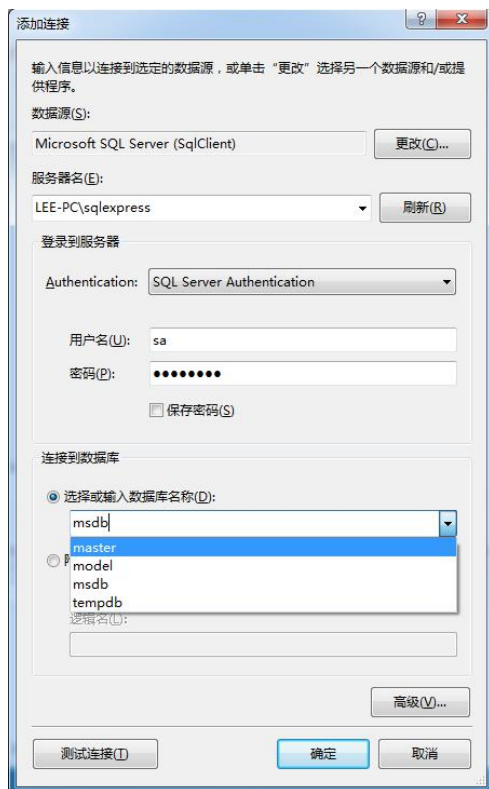
(1) 连接数据库



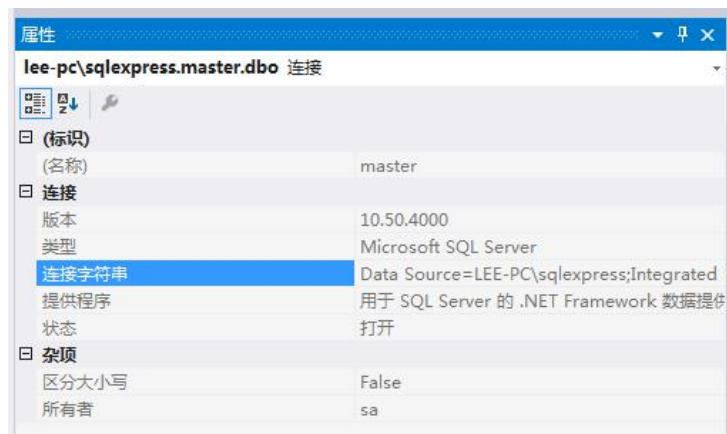
(2) 设置数据源类型



(3) 连接及测试



(4) 属性中的连接字符串



(二) 编程中使用数据库

(5) 在 Form 上加入 datagridview, 加入两个按钮



(6) 代码中加上数据库相关命名空间

```
using System.Data.SqlClient;
```

(7) 第一个按钮“连接数据库”中的代码



```
//定义数据库连接字符
```

```
string str_con = @"Data Source=LEE-PC\sqlexpress;Integrated  
Security=SSPI;Initial Catalog=testDB;";
```

```
//Integrated Security 身份验证方式
```

```
//当为 false 时，将在连接中指定用户 ID 和密码。
```

```
//当为 true 时，将使用当前的 Windows 帐户凭据进行身份验证。
```

```
//可识别的值为 true、false、yes、no 以及与 true 等效的 sspi
```

```
//如果非集成认证，必须在联接字符串里写上
```

```
uid=sa;password=00
```

```
//定义并实例化数据库连接类型变量
```

```
SqlConnection conn = new SqlConnection(str_con);
```

```
//打开数据库连接
```

```
conn.Open();
```

```
//定义 SQL 语句字符串 select
```

```
string sql = "select * from testStudent";
```

```
string sql1 = "select * from course";
```



```

        // string sql = string.Format("insert into danwei (danwei)
values ('{0}']", textBox1.Text);

```

```

SqlCommand comm = new SqlCommand(sql, conn);
SqlDataAdapter da = new SqlDataAdapter(comm);
SqlCommand comm1 = new SqlCommand(sql1, conn);
SqlDataAdapter da1 = new SqlDataAdapter(comm1);

```

```

DataSet ds = new DataSet();
da.Fill(ds, "student");
da1.Fill(ds, "course"); //可插入多表

```

```

dataGridView1.DataSource = ds;
dataGridView1.DataMember = "student";
//尝试下打开下面一行
//dataGridView1.DataMember = "course";

```

```

//设置 dataGridView1 控件第一列的列头文字
dataGridView1.Columns[0].HeaderText = "序号";
//设置 dataGridView1 控件第一列的列宽
dataGridView1.Columns[0].Width = 80;

```

```

//设置 dataGridView1 控件第二列的列头文字
dataGridView1.Columns[1].HeaderText = "姓名";
//设置 dataGridView1 控件第二列绑定的数据集中第一个数据表的
的第二列的数据

```

```

//前面说了，ds 数据集是一个多表的集合
dataGridView1.Columns[1].DataPropertyName =
ds.Tables[0].Columns[1].ToString();
//设置 dataGridView1 控件第二列的列宽
dataGridView1.Columns[1].Width = 100;

```

```

conn.Close(); //关闭数据库连接

```

(8) 第二个按钮“插入一条数据”中的代码

```

string str_con = @"Data Source=LEE-PC\sqlexpress;Integrated
Security=SSPI;Initial Catalog=testDB;";

```

```

SqlConnection conn = new SqlConnection(str_con);
//打开数据库连接
conn.Open();

```

```

//插入一条新数据
SqlCommand cmdinsert = new SqlCommand("insert into
testStudent (name,sno,sex) values('何芳','16054002','女')", conn);
cmdinsert.ExecuteNonQuery();

```

```
string sql = "select * from testStudent";  
SqlCommand comm = new SqlCommand(sql, conn);  
SqlDataAdapter da = new SqlDataAdapter(comm);  
DataSet ds = new DataSet();  
da.Fill(ds, "student");  
dataGridView1.DataSource = ds;  
dataGridView1.DataMember = "student";  
conn.Close();
```