**广州航海学院**

课程设计（实训）报告

****

课 程 C# 程 序 设 计

题 目 餐厅菜品数据管理系统

专 业 计 算 机 应 用

指导教师 严 健 武

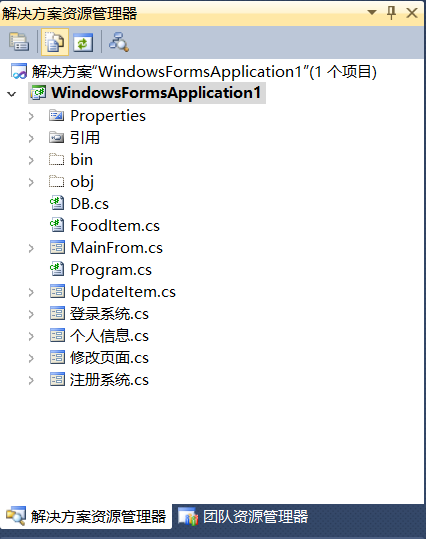
班级 计应172 学号 201715010218 姓名 李志豪

20 18 年 6 月 27 日《C#程序设计》课程设计任务书

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 系 别 | | 计算机系 | 专 业 | 计应171、172 |
| 题 目 | | 餐厅菜品数据管理系统 | | |
| 指导教师 | | 严健武 | | |
| 设计时间 | | 2018 年 6 月 25 日 至 2018 年 6 月 29 日 共1周 | | |
| 课  程  设  计  要  求 | 1. **根据已提供的数据库，实现如下功能：**   1.实现系统登录功能  1）制作登录界面，根据输入的用户名和密码，从提供的数据库的用户表中，查询用户是否存在来决定是否允许进入系统主操作界面。(如果账号输入错误，提示用户重新输入)。  2）登录界面在屏幕居中显示，窗体不可以调整大小，标题栏显示“登录系统”。  2.实现菜品数据的添加、修改、删除和查询功能。  1）制作主界面，添加“操作”和“关于”两个菜单项，其中“操作”菜单项又包含以下子菜单项：a)显示数据/关闭显示；b)分隔线；c)退出系统 ；添加工具栏，工具栏包含“显示数据”按钮，实现功能与对应的菜单项功能一致。  2)“显示数据”菜单项功能：单击该菜单，如果当前显示的文本是“显示数据”，则切换文字为“关闭显示”，同时打开显示数据窗体。  3)制作显示数据窗体：在窗体中包含数据表格控件DataGridView，打开该窗体时在DataGridView显示菜品表的所有数据；在窗体下方添加Panel面板控件，包含“添加”、“修改”、“删除”、“查询”四个按钮。  其中：   添加按钮：实现功能为，打开添加数据窗体，提供用户录入菜品数据界面，并实现添加数据到DataGridView和对应的数据表功能。   修改按钮：实现功能为，打开修改数据窗体，实现修改在DataGridView选择的行数据，并修改对应数据表数据功能。   删除按钮：实现功能为，直接删除在DataGridView中选中的行。   查询按钮：根据文本框中输入的菜品名，以模糊匹配查询方式将查询结果显示到DataGridView中。如果文本框没有输入，则显示全部数据。  4）主界面“退出系统”菜单项功能：实现单击该菜单项时，退出系统。  5）主界面“关于”菜单项功能：实现单击该菜单项时，弹出自定义窗体，在自定义窗体中显示自己的姓名、班级和学号信息。  **二、撰写课程设计报告**  要求大家在完成课程设计的同时，以书面形式交一份课程设计报告（用A4纸打印成册）。   * 课程设计报告的格式按照提供的格式，封面一定要严格按照标准格式设计； * 在课程设计报告内容中需要有总体设计的结构图及功能模块的名称（E—R图、功能模块图、关系数据模型）；需要有课程设计的表的结构，名称、类型和宽度的描述；也需要有创建的查询等的描述；实现的功能和应用都需要在此有描述。如果实例太多，可选择典型说明等。基本可以参考数据库设计步骤去写。 * 在课程设计报告中需要图文并茂，有文字说明，有程序，有图形佐证。 * 最后还需要写心得体会，总结设计中遇到的问题和解决方法。   **三、其它要求**  准时参加课程设计安排的课程，不迟到、早退，生病或有事应在课前请假，考勤情况将计入课程设计总评成绩。 | | | |

**目 录**

说明：课程设计报告完成后，在此处插入目录



正文

# 需求分析



假定一个餐厅的每张餐台都配备了一台无线联网的触屏式终端，需要开发一个具有网络点餐功能的管理系统，将顾客在终端点餐的数据自动传送到服务器。在结账时，收银台可以通过该系统自动计算出每张餐台顾客的消费金额并打印消费清单，整个系统构成的示意图如图所示。

根据系统功能需求，从顾客的角度来分析可知，顾客要点餐，必须有菜单(菜品列表)，考虑的餐厅菜品的多样性，人为的将菜品按照菜系分类，从而得到菜系和菜品两个实体。

站在餐厅商家的角度分析，老板可能要在某个时间段进行菜品分析，统计得到最受欢迎菜品，或进行营业额统计，通过结账单，可以检索到餐台点餐单及明细等信息。

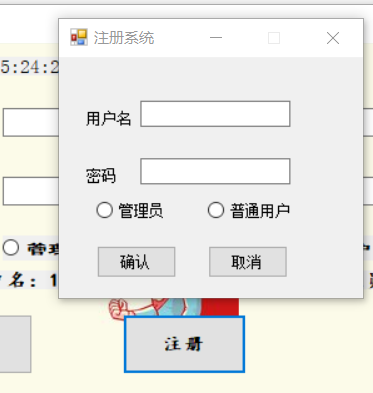
站在前台的角度分析，前台需要统计客人下的订单，就要有购物车，此处通过双击某一行就可以把菜名和价格加入购物车，如菜单信息量太大，就可以通过查询处用模糊查询输入字段查找菜品名，省去了用眼睛找的麻烦。若客人需要修改，就按清空购物车的按钮。重新输入。

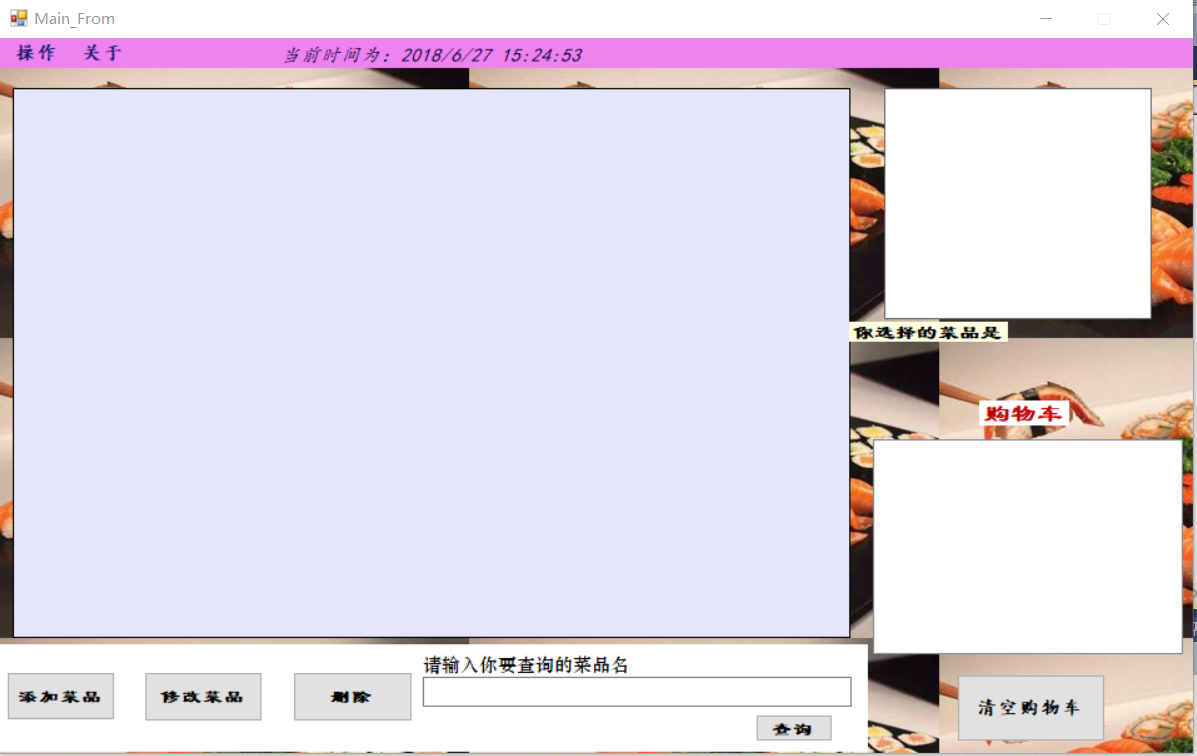
# 总体设计

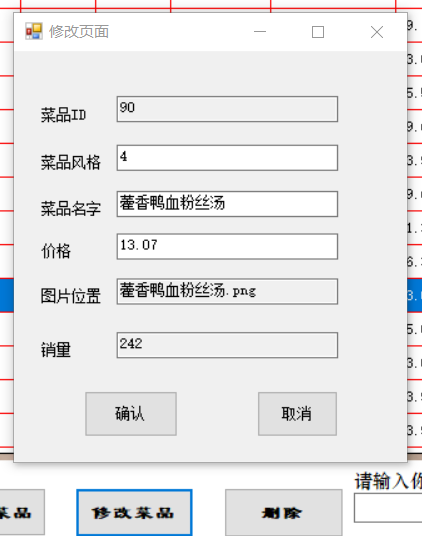
设计一个餐厅点餐系统的需求：

1. 登录注册系统。两个界面都要连接数据库。
2. 主窗口即点餐页面，其中包括“添加菜品、修改菜品、删除、查询”作为主要控件，也有“dataGridView”控件作为菜单显示的界面。
3. 添加控件与修改控件需要新建新的窗口。其他都是通过代码实现。

如图：



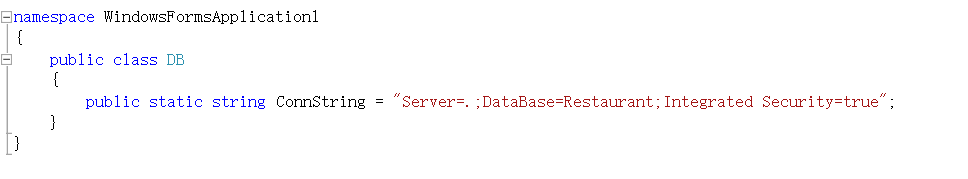


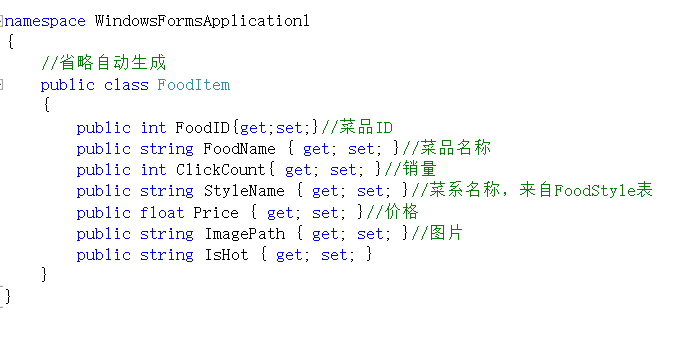


# 详细设计

系统中所运用的类:DB和FoodItem。

如图所示：





1. 登录系统的代码为

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace WindowsFormsApplication1

{

public partial class 登录系统 : Form

{

public 登录系统()

{

InitializeComponent();

}

private void button2\_Click(object sender, EventArgs e)

{

注册系统 frm2 = new 注册系统();

frm2.Show();

}

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

{

this.Close();

}

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

{

try

{

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

{

MessageBox.Show(" 请输入用户名和密码");

return;

}

else if (radioButton1.Checked == false && radioButton2.Checked == false)

{

MessageBox.Show("请选择管理员和用户");

return;

}

SqlConnection conn = new SqlConnection();

conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

conn.Open();

int dj = radioButton1.Checked ? 1 : 0;

string sql = "select count(\*)from Users where username='" + textBox1.Text + "'and pwd='" + textBox2.Text + "'and userclass=" + dj;

SqlCommand comm = new SqlCommand(sql, conn);

int rows = (int)comm.ExecuteScalar();

if (rows == 1)

{

MessageBox.Show("成功登录");

Main\_From frm3 = new Main\_From();

frm3.Show();

this.Hide();

}

else

{

MessageBox.Show("用户不存在或者密码错误" + "\r\n" + "请重新输入");

}

}

catch (Exception)

{

MessageBox.Show("数据库未连接");

}

}

private void 登录系统\_Load(object sender, EventArgs e)

{

timer1.Enabled = true;

}

private void timer1\_Tick(object sender, EventArgs e)

{

label3.Text = "当前时间为:" + DateTime.Now.ToString();

}

}

}



其中，包括登录、注册、取消，等按钮，注册按钮的代码为打开注册系统，登录按钮的代码是先查询数据库Restaurant中的fooditem表格如果存在，输出登录成功，打开主菜单MainFrom。如果输入为空，则显示提示。输入信息在数据库不存在，则提示错误。

1. 注册系统

代码为：

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

{

try

{

string username = textBox1.Text;

string pwd = textBox2.Text;

string userclass = "";

if (radioButton1.Checked) userclass = "1";

if (radioButton2.Checked) userclass = "0";

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

{

MessageBox.Show("请输入用户名和密码");

return;

}

SqlConnection conn = new SqlConnection();

conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

conn.Open();

string sql = string.Format("Insert into Users(username,pwd,userclass)values('{0}','{1}','{2}')", username, pwd, userclass);

SqlCommand comm = new SqlCommand(sql, conn);

comm.ExecuteNonQuery();

conn.Close();

MessageBox.Show("成功注册");

this.Close();

}

catch (Exception)

{

MessageBox.Show("用户已存在");

}

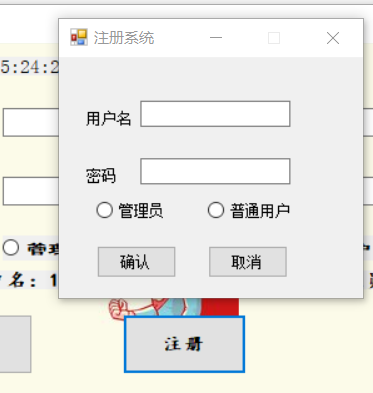
}

private void button2\_Click(object sender, EventArgs e)

{

this.Close();

}

如图，窗口中存在两个输入条，输入用户名和密码完成注册。代码说明：如果输入相同的用户，提输出提示。如果输入数据库不存在的用户，则显示注册成功。

1. 主菜单设计

代码：

public partial class Main\_From : Form

{

public Main\_From()

{

InitializeComponent();

}

void InitGridView()//新定义画表代码

{

dataGridView1.Rows.Clear();

dataGridView1.Columns.Clear();

//建表

dataGridView1.Columns.Add("Pos", "序号");

dataGridView1.Columns.Add("ClickCount", "销量");

dataGridView1.Columns.Add("StyleName", "菜系名称");

dataGridView1.Columns.Add("FoodID", "菜品ID");

dataGridView1.Columns.Add("FoodName", "菜品名称");

dataGridView1.Columns.Add("Price", "价格");

dataGridView1.Columns.Add("ImagePath", "图片位置");

dataGridView1.Columns.Add("IsHot", "是否热门");

//设置列宽

dataGridView1.Columns[0].Width = 60;

dataGridView1.Columns[1].Width = 60;

dataGridView1.Columns[2].Width = 60;

dataGridView1.Columns[3].Width = 80;

dataGridView1.Columns[4].Width = 100;

dataGridView1.Columns[5].Width = 80;

dataGridView1.Columns[6].Width = 160;

dataGridView1.Columns[7].Width = 50;

//属性设置

dataGridView1.ReadOnly = true;

dataGridView1.AllowUserToAddRows = true;

dataGridView1.AllowUserToDeleteRows = false;

dataGridView1.AllowUserToResizeRows = false;

dataGridView1.RowHeadersVisible = false;

dataGridView1.MultiSelect = false;

dataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect;

}

void FillGrid(FoodItem Food)

{

int count = dataGridView1.Rows.Count;

dataGridView1.Rows.Add();

dataGridView1.Rows[count - 1].Cells["pos"].Value = count ;

dataGridView1.Rows[count - 1].Cells["FoodID"].Value = Food.FoodID;

dataGridView1.Rows[count - 1].Cells["FoodName"].Value = Food.FoodName;

dataGridView1.Rows[count - 1].Cells["ClickCount"].Value = Food.ClickCount;

dataGridView1.Rows[count - 1].Cells["StyleName"].Value = Food.StyleName;

dataGridView1.Rows[count - 1].Cells["Price"].Value = Food.Price;

dataGridView1.Rows[count - 1].Cells["ImagePath"].Value = Food.ImagePath;

dataGridView1.Rows[count - 1].Cells["IsHot"].Value = Food.IsHot;

}

void GetFood()

{

string connString = "Server=(local);DataBase=Restaurant;Integrated Security=true";

SqlConnection conn = new SqlConnection(connString);

conn.Open();

string sql = "select \* from FoodItem";

SqlCommand comm = new SqlCommand(sql, conn);

SqlDataReader dr = comm.ExecuteReader();

FoodItem Food = new FoodItem();

while (dr.Read())

{

//读取并保存数据

Food.FoodID = int.Parse(dr["FoodID"].ToString());

Food.FoodName = dr["FoodName"].ToString();

Food.ClickCount = int.Parse(dr["ClickCount"].ToString());

Food.StyleName = dr["StyleName"].ToString();

Food.Price = float.Parse(dr["Price"].ToString());

Food.ImagePath = dr["imagePath"].ToString();

Food.IsHot = dr["IsHot"].ToString();

FillGrid(Food);

}

dr.Close(); conn.Close();

}

private void Main\_From\_Load(object sender, EventArgs e)

{

timer1.Enabled = true;

}

private void timer1\_Tick(object sender, EventArgs e)

{

label2.Text = "当前时间为：" + DateTime.Now.ToString();

}

private void 关于ToolStripMenuItem\_Click(object sender, EventArgs e)

{

个人信息 frm = new 个人信息();

frm.Show();

}

private void 退出ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private void 显示菜单ToolStripMenuItem\_Click(object sender, EventArgs e)

{

InitGridView();

GetFood();

if (dataGridView1.Visible == false)

{

dataGridView1.Visible = true;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

if (dataGridView1.CurrentRow == null) return;

string selectInfo = "";

selectInfo += dataGridView1.CurrentRow.Cells["pos"].Value + "";

selectInfo += dataGridView1.CurrentRow.Cells["FoodID"].Value + "";

selectInfo += dataGridView1.CurrentRow.Cells["FoodName"].Value + "";

selectInfo += dataGridView1.CurrentRow.Cells["ClickCount"].Value + "";

selectInfo += dataGridView1.CurrentRow.Cells["StyleName"].Value + "";

selectInfo += dataGridView1.CurrentRow.Cells["Price"].Value + "";

selectInfo += dataGridView1.CurrentRow.Cells["ImagePath"].Value;

selectInfo += dataGridView1.CurrentRow.Cells["IsHot"].Value + "";

this.Text = selectInfo;

string selectInfo1 = "";

selectInfo1 += dataGridView1.CurrentRow.Cells["FoodName"].Value + ",";

selectInfo1 += dataGridView1.CurrentRow.Cells["Price"].Value + ",";

listBox1.Items.Add(selectInfo1);

string selectInfo2 = "";

selectInfo2 += dataGridView1.CurrentRow.Cells["FoodName"].Value + ",";

label4.Text = selectInfo2;

try

{

SqlConnection conn = new SqlConnection();

SqlConnectionStringBuilder scsb = new SqlConnectionStringBuilder();

scsb.DataSource = "(local)";

scsb.InitialCatalog = "Restaurant";

scsb.IntegratedSecurity = true;

conn.ConnectionString = scsb.ConnectionString;

conn.Open();

string s = dataGridView1.CurrentRow.Cells["imagepath"].Value.ToString();

string name = dataGridView1.CurrentRow.Cells["foodname"].Value.ToString();

pictureBox1.Image = Image.FromFile("pic\\" + s);

conn.Close();

}

catch (Exception)

{

Exception ex = new Exception();

label6.Text = "抱歉，没有这一张照片";

}

}

private void 关闭显示ToolStripMenuItem\_Click(object sender, EventArgs e)

{

dataGridView1.Visible = false;

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

//打开UpdateItem窗口

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

{

UpdateItem frm = new UpdateItem();

frm.Show();

}

private void button2\_Click(object sender, EventArgs e)

{

修改页面 frm1 = new 修改页面();

if (dataGridView1.Visible == false)

{

MessageBox.Show("当前数据为空，无法修改");

frm1.Hide();

}

else

{

frm1.dgv = dataGridView1;

frm1.Show();

}

}

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

{

try

{

if (dataGridView1.CurrentRow == null) return;

int FoodID = int.Parse(dataGridView1.CurrentRow.Cells["FoodID"].Value.ToString());

SqlConnection conn = new SqlConnection(DB.ConnString);

conn.Open();

string sql = "Delete From FoodItem Where FoodID=" + FoodID;

SqlCommand comm = new SqlCommand(sql, conn);

comm.ExecuteNonQuery();

conn.Close();

dataGridView1.Rows.Remove(dataGridView1.CurrentRow);

}

catch (Exception)

{

MessageBox.Show("请选择有效行");

}

}

private void button4\_Click(object sender, EventArgs e)

{

//SqlConnection conn = new SqlConnection();

//conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

//conn.Open();

if (textBox1.Text == "")

{

InitGridView();

GetFood();

}

else

{

SqlConnection conn = new SqlConnection();

conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

conn.Open();

string s = "";

s = textBox1.Text.ToString();

string sql = string.Format("select \* from FoodItem where foodName Like '{0}%'", s);

string sql2 = string.Format("Insert \* from FoodItem where foodName Like '{0}%'", s);

SqlCommand comm = new SqlCommand(sql,conn);

SqlCommand comm2 = new SqlCommand(sql2, conn);

comm.ExecuteNonQuery();

SqlDataReader dr = comm.ExecuteReader();

FoodItem Food = new FoodItem();

InitGridView();

int count = dataGridView1.Rows.Count;

dataGridView1.Rows.Add();

dataGridView1.Rows[count-1].Cells["pos"].Value = count;

dataGridView1.Rows[count-1].Cells["FoodID"].Value = Food.FoodID;

dataGridView1.Rows[count-1].Cells["FoodName"].Value = Food.FoodName;

dataGridView1.Rows[count-1].Cells["ClickCount"].Value = Food.ClickCount;

dataGridView1.Rows[count-1].Cells["StyleName"].Value = Food.StyleName;

dataGridView1.Rows[count-1].Cells["Price"].Value = Food.Price;

dataGridView1.Rows[count-1].Cells["ImagePath"].Value = Food.ImagePath;

dataGridView1.Rows[count-1].Cells["IsHot"].Value = Food.IsHot;

while (dr.Read())

{

//读取并保存数据

Food.FoodID = int.Parse(dr["FoodID"].ToString());

Food.FoodName = dr["FoodName"].ToString();

Food.ClickCount = int.Parse(dr["ClickCount"].ToString());

Food.StyleName = dr["StyleName"].ToString();

Food.Price = float.Parse(dr["Price"].ToString());

Food.ImagePath = dr["imagePath"].ToString();

Food.IsHot = dr["IsHot"].ToString();

FillGrid(Food);

}

conn.Close(); dr.Close();

}

}

private void button5\_Click(object sender, EventArgs e)

{

}

private void button6\_Click(object sender, EventArgs e)

{

listBox1.Items.Clear();

}

}

其中每段代码都有注释。

主界面中的添加菜品按钮打开窗体的代码为：

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

{

try

{

string StyleName = textBox1.Text;

string FoodName = textBox2.Text;

string Price = textBox3.Text;

if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "")

{

MessageBox.Show("请输入完整的菜品信息");

return;

}

SqlConnection conn = new SqlConnection();

conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

conn.Open();

string sql = string.Format("Insert into FoodItem(StyleName,FoodName,Price)values('{0}','{1}','{2}')", StyleName, FoodName, Price);

SqlCommand comm = new SqlCommand(sql, conn);

comm.ExecuteNonQuery();

conn.Close();

MessageBox.Show("成功录入");

}

catch (Exception)

{

MessageBox.Show("请输入数字类型");

}

}

代码说明：先判断三条输入条是否为空，否则提示。连接数据库，用Insert代码录入数据库，再执行。

修改页面代码为：

public partial class 修改页面 : Form

{

public 修改页面()

{

InitializeComponent();

}

public DataGridView dgv;

private void button2\_Click(object sender, EventArgs e)

{

this.Close();

}

private void 修改页面\_Load(object sender, EventArgs e)

{

try

{

textBox1.Text = dgv.CurrentRow.Cells["FoodID"].Value.ToString();

textBox2.Text = dgv.CurrentRow.Cells["StyleName"].Value.ToString();

textBox3.Text = dgv.CurrentRow.Cells["FoodName"].Value.ToString();

textBox4.Text = dgv.CurrentRow.Cells["Price"].Value.ToString();

textBox5.Text = dgv.CurrentRow.Cells["ImagePath"].Value.ToString();

textBox6.Text = dgv.CurrentRow.Cells["ClickCount"].Value.ToString();

}

catch (Exception)

{

MessageBox.Show("选择信息为空");

this.Close();

}

}

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

{

try

{

SqlConnection conn = new SqlConnection();

conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

conn.Open();

if (textBox2.Text == "" || textBox3.Text == "" || textBox4.Text == "")

{

MessageBox.Show("请输入完整信息");

return;

}

else

{

string sql = string.Format("update FoodItem set StyleName='{0}',FoodName='{1}',Price='{2}'where FoodID={3}", textBox2.Text, textBox3.Text, textBox4.Text,textBox1.Text);

SqlCommand comm = new SqlCommand(sql,conn);

comm.ExecuteNonQuery();

MessageBox.Show("修改成功");

conn.Close();

this.Close();

}

}

catch (Exception)

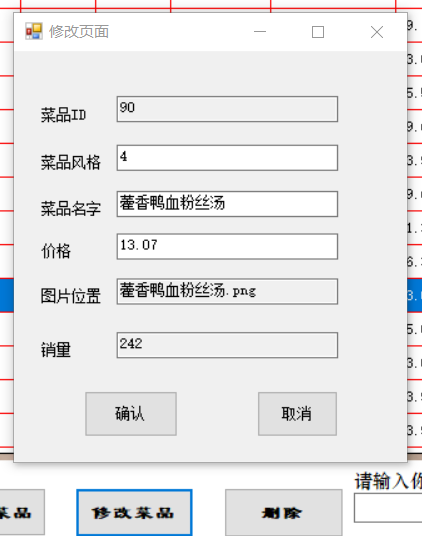
{

MessageBox.Show("价格处请输入数字");

return;

}

}

代码说明：先连接数据库，读取数据库中的数据显示在如图的输入条中，通过改变输入条中的信息来修改数据。

删除按钮的代码为：

try

{

if (dataGridView1.CurrentRow == null) return;

int FoodID = int.Parse(dataGridView1.CurrentRow.Cells["FoodID"].Value.ToString());

SqlConnection conn = new SqlConnection(DB.ConnString);

conn.Open();

string sql = "Delete From FoodItem Where FoodID=" + FoodID;

SqlCommand comm = new SqlCommand(sql, conn);

comm.ExecuteNonQuery();

conn.Close();

dataGridView1.Rows.Remove(dataGridView1.CurrentRow);

}

catch (Exception)

{

MessageBox.Show("请选择有效行");

}

代码说明：连接数据库，用Delete代码执行删除。

查询按钮的代码为：

private void button4\_Click(object sender, EventArgs e)

{

if (textBox1.Text == "")

{

InitGridView();

GetFood();

}

else

{

SqlConnection conn = new SqlConnection();

conn.ConnectionString = "Data Source=(local);Initial Catalog=Restaurant;Integrated Security=true";

conn.Open();

string s = "";

s = textBox1.Text.ToString();

string sql = string.Format("select \* from FoodItem where foodName Like '{0}%'", s);

string sql2 = string.Format("Insert \* from FoodItem where foodName Like '{0}%'", s);

SqlCommand comm = new SqlCommand(sql,conn);

SqlCommand comm2 = new SqlCommand(sql2, conn);

comm.ExecuteNonQuery();

SqlDataReader dr = comm.ExecuteReader();

FoodItem Food = new FoodItem();

InitGridView();

int count = dataGridView1.Rows.Count;

dataGridView1.Rows.Add();

dataGridView1.Rows[count-1].Cells["pos"].Value = count;

dataGridView1.Rows[count-1].Cells["FoodID"].Value = Food.FoodID;

dataGridView1.Rows[count-1].Cells["FoodName"].Value = Food.FoodName;

dataGridView1.Rows[count-1].Cells["ClickCount"].Value = Food.ClickCount;

dataGridView1.Rows[count-1].Cells["StyleName"].Value = Food.StyleName;

dataGridView1.Rows[count-1].Cells["Price"].Value = Food.Price;

dataGridView1.Rows[count-1].Cells["ImagePath"].Value = Food.ImagePath;

dataGridView1.Rows[count-1].Cells["IsHot"].Value = Food.IsHot;

while (dr.Read())

{

//读取并保存数据

Food.FoodID = int.Parse(dr["FoodID"].ToString());

Food.FoodName = dr["FoodName"].ToString();

Food.ClickCount = int.Parse(dr["ClickCount"].ToString());

Food.StyleName = dr["StyleName"].ToString();

Food.Price = float.Parse(dr["Price"].ToString());

Food.ImagePath = dr["imagePath"].ToString();

Food.IsHot = dr["IsHot"].ToString();

FillGrid(Food);

}

conn.Close(); dr.Close();

}

}



代码说明：

如果输入条为空，则 运用之前新定义的InitGridView(); GetFood();直接打开表格。运用模糊查询，输入关键字，读取处表格中有关键字的数据。读取之前还要运用一开始新定义的建立表格代码与读取并录入信息的代码，方可执行查询。

# 四、课程设计总结(心得体会)

本次C#课程设计我花了差不多一周时间才完成，从老师开学的第一节课开始，就跟我们说本学期课程设计是做一个点餐系统，在一个学期的学习C#知识以来，我就感觉到用C#做一个系统是虽然很困难。但是做完实现运行所有功能的那一瞬间，真心能让我感觉到自己那么多天废寝忘食的努力是真的有收获的。

本次设计的点餐系统完全是由我个人不断地通过各种查询方式，把自己不会的，学过的，又或者是学过又忘记的知识点翻了一遍又一遍。编程过程中遇到过不少bug，但我能通过问同学，老师，自己查询书籍不断地解决一个有一个bug。其过程现在回想起来都觉得有趣，因为能够切切实实地感觉到自己学到了知识，编程能力得到了提升。

这次课程设计包含很多知识点，是我大学以来学习编程过程中的最好的一段经历。我以后会更加努力学习编程，把编程当作自己的兴趣，自己做出的系统我一定会站在客户的角度上，把系统完善得毫无漏洞。

# 五、参考资料

[1] Andrew Troelsen著 朱晔 译 《C#与.NET 4高级程序设计》 2011年4月 人民邮电出版社。

[2] 百度一下

[3] 文件: 数据库操作.pdf