FoodManger.cs

using System.Data;

using System.Data.SQLite;

using System.Linq.Expressions;

namespace FoodManagerReceipr

{

public partial class frmFdMgr : Form

{

public string dbfile = "URI=file:FoodManagerReceipe.db";

public SQLiteConnection ARSqlconn;

public SQLiteCommand ARCmd;

public SQLiteDataReader ARDataReader;

public SQLiteDataAdapter ARDap;

public DataTable ARTbl;

public static string MNU, CSN, CTG, TPY;

public frmFdMgr()

{

InitializeComponent();

ARSqlconn = new SQLiteConnection(dbfile);

//this.WindowState = FormWindowState.Maximized;

}

// To create common function to fill data grid box which will update dynamically latest data from Vehicle table

public void FillCuisineGrid()

{

// Connect to Member table and fetch all members details

ARSqlconn.Open();

ARDap = new SQLiteDataAdapter("select \* from Cuisines", ARSqlconn);

ARSqlconn.Close();

SQLiteCommandBuilder ARCdB = new SQLiteCommandBuilder(ARDap);

ARTbl = new DataTable();

ARDap.Fill(ARTbl);

//dGVCuisine.DataSource = ARTbl;

}

private void Form1\_Load(object sender, EventArgs e)

{

FillCuisineGrid();

}

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

{

pnLCuisines.Visible = true;

pnlCatg.Visible = true;

MNU = "1";

}

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

{

pnLCuisines.Visible = true;

pnlCatg.Visible = true;

MNU = "2";

}

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

{

pnLCuisines.Visible = true;

pnlCatg.Visible = true;

MNU = "3";

}

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

{

pnLCuisines.Visible = true;

pnlCatg.Visible = true;

MNU = "4";

}

private void btnEng\_MouseMove(object sender, MouseEventArgs e)

{

CSN = "1";

pnlmov.Visible = true;

pnlmov.Left = btnEng.Left;

pnlmov.Width = btnEng.Width;

pnlCatg.Visible=true;

//pnlCatg.Location = new Point(218, 230);

pnlCatg.Left = btnEng.Left;

pnlCatg.Width = btnEng.Width;

}

private void btnInd\_MouseMove(object sender, MouseEventArgs e)

{

CSN = "3";

pnlmov.Visible = true;

pnlmov.Left = btnInd.Left;

pnlmov.Width = btnInd.Width;

pnlCatg.Visible = true;

pnlCatg.Left = btnInd.Left;

pnlCatg.Width = btnInd.Width;

}

private void btnIta\_MouseMove(object sender, MouseEventArgs e)

{

CSN = "2";

pnlmov.Visible = true;

pnlmov.Left = btnIta.Left;

pnlmov.Width = btnIta.Width;

pnlCatg.Visible = true;

pnlCatg.Left = btnIta.Left;

pnlCatg.Width = btnIta.Width;

}

private void btnChn\_MouseMove(object sender, MouseEventArgs e)

{

CSN = "4";

pnlmov.Visible = true;

pnlmov.Left = btnChn.Left;

pnlmov.Width = btnChn.Width;

pnlCatg.Visible = true;

//pnlCatg.Location = new Point(806, 230);

pnlCatg.Left = btnChn.Left;

pnlCatg.Width = btnChn.Width;

}

private void btnJap\_MouseMove(object sender, MouseEventArgs e)

{

CSN = "5";

pnlmov.Visible = true;

pnlmov.Left = btnJap.Left;

pnlmov.Width = btnJap.Width;

pnlCatg.Visible = true;

//pnlCatg.Location = new Point(1002, 230);

pnlCatg.Left = btnJap.Left;

pnlCatg.Width = btnJap.Width;

}

private void btnDrnk\_MouseMove(object sender, MouseEventArgs e)

{

CSN = "6";

pnlmov.Visible = true;

pnlmov.Left = btnDrnk.Left;

pnlmov.Width = btnDrnk.Width;

}

private void btnEng\_MouseLeave(object sender, EventArgs e)

{

pnlmov.Visible = false;

pnlmovII.Visible = false;

//pnlCatg.Visible = false;

}

private void btnInd\_MouseLeave(object sender, EventArgs e)

{

pnlmov.Visible = false;

pnlmovII.Visible = false;

//pnlCatg.Visible = false;

}

private void btnIta\_MouseLeave(object sender, EventArgs e)

{

pnlmov.Visible = false;

pnlmovII.Visible = false;

// pnlCatg.Visible = false;

}

private void btnChn\_MouseLeave(object sender, EventArgs e)

{

pnlmov.Visible = false;

pnlmovII.Visible = false;

//pnlCatg.Visible = false;

}

private void btnJap\_MouseLeave(object sender, EventArgs e)

{

pnlmov.Visible = false;

pnlmovII.Visible = false;

//pnlCatg.Visible = false;

}

private void btnDrnk\_MouseLeave(object sender, EventArgs e)

{

pnlmov.Visible = false;

//pnlCatg.Visible = false;

}

private void btnckn\_MouseMove(object sender, MouseEventArgs e)

{

pnlmovII.Visible = true;

pnlmovII.Left = btnckn.Left;

pnlmovII.Width = btnckn.Width;

pBxChk.Visible = true;

}

private void btnbef\_MouseMove(object sender, MouseEventArgs e)

{

pnlmovII.Visible = true;

pnlmovII.Left = btnbef.Left;

pnlmovII.Width = btnbef.Width;

pBxbef.Visible = true;

}

private void btnprk\_MouseMove(object sender, MouseEventArgs e)

{

pnlmovII.Visible = true;

pnlmovII.Left = btnprk.Left;

pnlmovII.Width = btnprk.Width;

pBxprk.Visible = true;

}

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

{

CTG = "2";

// Call Select Form

SelC();

}

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

{

CTG = "3";

// Call Select Form

SelC();

}

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

{

CTG = "4";

// Call Select Form

SelC();

}

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

{

TPY = "1";

// Call Select Form

SelC();

}

private void btnlmb\_MouseMove(object sender, MouseEventArgs e)

{

pnlmovII.Visible = true;

pnlmovII.Left = btnlmb.Left;

pnlmovII.Width = btnlmb.Width;

pBxlmb.Visible = true;

}

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

{

TPY = "2";

// Call Select Form

SelC();

}

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

{

TPY = "3";

// Call Select Form

SelC();

}

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

{

TPY = "4";

// Call Select Form

SelC();

}

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

{

TPY = "5";

// Call Select Form

SelC();

}

private void btnDrnk\_Click\_1(object sender, EventArgs e)

{

CSN = "6";

// Call Select Form

SelC();

}

private void btnckn\_MouseLeave(object sender, EventArgs e)

{

pBxChk.Visible = false;

}

private void btnbef\_MouseLeave(object sender, EventArgs e)

{

pBxbef.Visible = false;

}

private void btnprk\_MouseLeave(object sender, EventArgs e)

{

pBxprk.Visible = false;

}

private void btnlmb\_MouseLeave(object sender, EventArgs e)

{

pBxlmb.Visible = false;

}

private void btnsf\_MouseLeave(object sender, EventArgs e)

{

pBxsf.Visible = false;

}

private void btnsf\_MouseMove(object sender, MouseEventArgs e)

{

pnlmovII.Visible = true;

pnlmovII.Left = btnsf.Left;

pnlmovII.Width = btnsf.Width;

pBxsf.Visible = true;

}

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

{

pnltyp.Visible = true;

CTG = "1";

}

// Select which for, to load

public void SelC()

{

if (MNU == "1")

{

// Display Receipe

//MessageBox.Show(" Display Recipe Selected");

pnlmov.Visible = false;

pnlmovII.Visible = false;

pnlCatg.Visible = false;

pnltyp.Visible = false;

pnLCuisines.Visible = false;

frmDspRcp dsp = new frmDspRcp();

dsp.Show();

}

else if (MNU == "2")

{

// Add recipe

//MessageBox.Show(" Add Recipe Selected");

pnlmov.Visible = false;

pnlmovII.Visible = false;

pnlCatg.Visible = false;

pnltyp.Visible = false;

pnLCuisines.Visible = false;

FrmAddRecipe Adrsp = new FrmAddRecipe();

Adrsp.Show();

}

else if (MNU == "3")

{

// Modify Receipe

MessageBox.Show(" Modify Recipe Selected");

}

else if (MNU == "4")

{

// Delete Receipe

MessageBox.Show(" Delete Recipe Selected");

}

}

}

}

DisplayRecipe.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Data.SQLite;

using System.Drawing;

using System.Drawing.Drawing2D;

using System.Linq;

using System.Reflection;

using System.Reflection.Emit;

using System.Runtime.Intrinsics.Arm;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Xml.Linq;

using static System.ComponentModel.Design.ObjectSelectorEditor;

using static System.Windows.Forms.VisualStyles.VisualStyleElement.ListView;

namespace FoodManagerReceipr

{

public partial class frmDspRcp : Form

{

public string dbfile = "URI=file:FoodManagerReceipe.db";

public SQLiteConnection ARSqlconn;

public SQLiteCommand ARCmd;

public SQLiteDataReader ARDataReader;

public SQLiteDataAdapter ARDap;

public DataTable ARTbl;

public static string RccPID;

public frmDspRcp()

{

InitializeComponent();

ARSqlconn = new SQLiteConnection(dbfile);

this.WindowState = FormWindowState.Maximized;

}

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

{

msgb();

FillRecpNamGrd();

}

public void FillRecpNamGrd()

{

// Fill Listbox

lBxRcpnam.Items.Clear();

ARSqlconn.Open();

SQLiteCommand command = new SQLiteCommand(ARSqlconn);

// Show Receipe as per selection

//frmFdMgr.CSN, frmFdMgr.CTG, frmFdMgr.TPY

string Cun = frmFdMgr.CSN, Cat = frmFdMgr.CTG, Ty = frmFdMgr.TPY;

if (Cun == "1" && Cat == "1" && Ty == "1") // English - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "1" && Ty == "2") // English - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "1" && Ty == "3") // English - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

//MessageBox.Show(" Display - Non Veg - Pork Selected");

}

else if (Cun == "1" && Cat == "1" && Ty == "4") // English - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "1" && Ty == "5") // English - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "2") // English - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "3") // English - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "4") // English - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "2" && Cat == "1" && Ty == "1") // Italian - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "2") // Italian - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "3") // Italian - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "4") // Italian - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "5") // Italian - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "2") // Italian - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "3") // Italian - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "4") // Italian - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "3" && Cat == "1" && Ty == "1") // Indian - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "2") // Indian - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "3") // Indian - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "4") // Indian - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "5") // Indian - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID AND Status = @stus ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "2") // Indian - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "3") // Indian - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "4") // Indian - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "4" && Cat == "1" && Ty == "1") // Chinese - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "2") // Chinese - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "3") // Chinese - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "4") // Chinese - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "5") // Chinese - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "2") // Chinese - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "3") // Chinese - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "5") // Chinese - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "5" && Cat == "1" && Ty == "1") // Japanese - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "2") // Japanese - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "3") // Japanese - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "4") // Japanese - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "5") // Japanese - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "2") // Japanese - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "3") // Japanese - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "5") // Japanese - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "6") // Drink

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C06");

command.Parameters.AddWithValue("@tID", "T05");

command.Parameters.AddWithValue("@sID", "ST10");

command.Parameters.AddWithValue("@stus", "Active");

}

command.ExecuteNonQuery();

SQLiteDataReader reader = command.ExecuteReader();

while (reader.Read())

{

lBxRcpnam.Items.Add(reader["RecipeName"]);

}

ARSqlconn.Close();

}

//On double click - Display the Receipe details

public void msgb()

{

if (frmFdMgr.CSN == "1") lblcus.Text = "ENGLISH";

else if (frmFdMgr.CSN == "2") lblcus.Text = "ITALIAN";

else if (frmFdMgr.CSN == "3") lblcus.Text = "INDIAN";

else if (frmFdMgr.CSN == "4") lblcus.Text = "CHINESE";

else if (frmFdMgr.CSN == "5") lblcus.Text = "JAPANESE";

else if (frmFdMgr.CSN == "6") lblcus.Text = "DRINKS";

if (frmFdMgr.CTG == "1") lblCat.Text = "NON VEG";

else if (frmFdMgr.CTG == "2") lblCat.Text = "VEG";

else if (frmFdMgr.CTG == "3") lblCat.Text = "VEGAN";

else if (frmFdMgr.CTG == "4") lblCat.Text = "DESSERT";

if (frmFdMgr.TPY == "1") lbltyp.Text = "CHICKEN";

else if (frmFdMgr.TPY == "2") lbltyp.Text = "BEEF";

else if (frmFdMgr.TPY == "3") lbltyp.Text = "PORK";

else if (frmFdMgr.TPY == "4") lbltyp.Text = "LAMB";

else if (frmFdMgr.TPY == "5") lbltyp.Text = "SEAFOOD";

}

//int RwId;

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

{

// When double click the row in the Grid table it should fetch the values in the text box

// Blank All Label Cell

label1.Text = "";

label2.Text = "";

label3.Text = "";

label4.Text = "";

label5.Text = "";

label6.Text = "";

label7.Text = "";

label8.Text = "";

label9.Text = "";

label10.Text = "";

label1.Visible = false;

label2.Visible = false;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

label7.Visible = false;

label8.Visible = false;

label9.Visible = false;

label10.Visible = false;

//RwId = e.RowIndex;

//DataGridViewRow RecpRowId = dGWRcpNam.Rows[RwId];

//File the textbox / label box and radio button values with the row selected

//lblRnm.Text = RecpRowId.Cells[0].Value.ToString().Trim();

if (lBxRcpnam.SelectedItem != null)

{

lblRnm.Text = lBxRcpnam.SelectedItem.ToString();

}

string rcname = lblRnm.Text.Trim();

// ARDap = new SQLiteDataAdapter("select Rcp\_ID from Receipe WHERE RecipeName='" + rcname + "'", ARSqlconn);

ARSqlconn.Open();

ARCmd = new SQLiteCommand(ARSqlconn);

ARCmd.CommandText = "SELECT Rcp\_ID from Receipe WHERE RecipeName = @RCiD";

ARCmd.Parameters.AddWithValue("@RCiD", rcname);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

string Rcp = (string)ARCmd.ExecuteScalar();

RccPID = Rcp;

ARDap = new SQLiteDataAdapter("select \* from Receipe WHERE Rcp\_ID='" + Rcp + "'", ARSqlconn);

ARSqlconn.Close();

SQLiteCommandBuilder ARCdB1 = new SQLiteCommandBuilder(ARDap);

ARTbl = new DataTable();

ARDap.Fill(ARTbl);

dGWRcnm.DataSource = ARTbl;

string Cui = dGWRcnm.Rows[0].Cells["Cus\_ID"].Value.ToString();

string Cta = dGWRcnm.Rows[0].Cells["Typ\_ID"].Value.ToString();

string Tpy = dGWRcnm.Rows[0].Cells["SubCat\_ID"].Value.ToString();

lblCal.Text = dGWRcnm.Rows[0].Cells["NutritionInfo"].Value.ToString();

lblYld.Text = dGWRcnm.Rows[0].Cells["Yield"].Value.ToString();

lblCt.Text = dGWRcnm.Rows[0].Cells["CookTime"].Value.ToString();

lblRt.Text = dGWRcnm.Rows[0].Cells["Rating"].Value.ToString();

lblRI.Text = dGWRcnm.Rows[0].Cells["receipeInfo"].Value.ToString();

pBxRcp.ImageLocation = @"D:\SHU\FoodManagerReceipr\data resorces\" + dGWRcnm.Rows[0].Cells["Pic"].Value.ToString();

// Display Ingredients details for selected Receipe

SQLiteCommand cmd = new SQLiteCommand(

@"select

Ingredients.OUM,

UOM.Abbv,

Ingredients.Items

from Ingredients

inner join UOM on Ingredients.UOM\_ID =

UOM.UOM\_ID

where Ingredients.Rcp\_ID= @RiD", ARSqlconn);

cmd.Parameters.AddWithValue("@RiD", Rcp);

cmd.CommandType = System.Data.CommandType.Text;

SQLiteDataAdapter sda = new SQLiteDataAdapter(cmd);

SQLiteCommandBuilder ARCdB2 = new SQLiteCommandBuilder(sda);

ARTbl = new DataTable();

sda.Fill(ARTbl);

dGVIngdt.DataSource = ARTbl;

ARSqlconn.Open();

ARCmd = new SQLiteCommand(ARSqlconn);

ARCmd.CommandText = "SELECT CuisineName FROM Cuisines WHERE Cuisines.Cus\_ID = @CiD";

ARCmd.Parameters.AddWithValue("@CiD", Cui);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

lblCui.Text = (string)ARCmd.ExecuteScalar();

ARCmd.CommandText = "SELECT FoodType FROM Types WHERE Types.Typ\_ID = @TiD";

ARCmd.Parameters.AddWithValue("@TiD", Cta);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

lblCatg.Text = (string)ARCmd.ExecuteScalar();

ARCmd.CommandText = "SELECT SubCat FROM Sub\_Category WHERE Sub\_Category.SubCat\_ID = @SiD";

ARCmd.Parameters.AddWithValue("@SiD", Tpy);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

lblTp.Text = (string)ARCmd.ExecuteScalar();

ARCmd.CommandText = "SELECT count(\*) FROM PrepareSteps WHERE PrepareSteps.Rcp\_ID = @RiD";

ARCmd.Parameters.AddWithValue("@RiD", Rcp);

int cnt = Convert.ToInt32(ARCmd.ExecuteScalar());

ARDap = new SQLiteDataAdapter("SELECT Preparation\_Details FROM PrepareSteps WHERE PrepareSteps.Rcp\_ID ='" + Rcp +"'", ARSqlconn);

ARSqlconn.Close();

SQLiteCommandBuilder ARCdB = new SQLiteCommandBuilder(ARDap);

ARTbl = new DataTable();

ARDap.Fill(ARTbl);

dGWStp.DataSource = ARTbl;

int idx = 0;

foreach (DataGridViewRow row in dGWStp.Rows)

{

if (idx == 0 && idx < cnt)

{

label1.Visible = true;

label1.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 1 && idx < cnt)

{

label2.Visible = true;

label2.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 2 && idx < cnt)

{

label3.Visible = true;

label3.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 3 && idx < cnt)

{

label4.Visible = true;

label4.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 4 && idx < cnt)

{

label5.Visible = true;

label5.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 5 && idx < cnt)

{

label6.Visible = true;

label6.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 6 && idx < cnt)

{

label7.Visible = true;

label7.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 7 && idx < cnt)

{

label8.Visible = true;

label8.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 8 && idx < cnt)

{

label9.Visible = true;

label9.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

else if (idx == 9 && idx < cnt)

{

label10.Visible = true;

label10.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

}

idx++;

}

}

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

{

ARSqlconn.Open();

ARCmd = new SQLiteCommand(ARSqlconn);

ARCmd.CommandText = "UPDATE Receipe SET Status = @st WHERE Rcp\_ID = @RCiD";

ARCmd.Parameters.AddWithValue("@st", "Inactive");

ARCmd.Parameters.AddWithValue("@RCiD", RccPID);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

MessageBox.Show("The Receipe "+ RccPID + " is been Inactivate... Cannot be Displayed/Edited");

}

}

}

AddRecipe.cs

using System;

using System.Collections;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SQLite;

using System.Drawing;

using System.Linq;

using System.Numerics;

using System.Reflection;

using System.Security.AccessControl;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using static System.Net.Mime.MediaTypeNames;

namespace FoodManagerReceipr

{

public partial class FrmAddRecipe : Form

{

public string dbfile = "URI=file:FoodManagerReceipe.db";

public SQLiteConnection ARSqlconn;

public SQLiteCommand ARCmd;

public SQLiteDataReader ARDataReader;

public SQLiteDataAdapter ARDap;

public DataTable ARTbl;

public static int rat, stp;

public static string [] Psteps = new string[10];

public static string [,] IngrDet = new string[18,4];

public static int nct = 1, Qct =0, Uct =0, ITct =0;

public FrmAddRecipe()

{

InitializeComponent();

ARSqlconn = new SQLiteConnection(dbfile);

this.WindowState = FormWindowState.Maximized;

}

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

{

msgb();

}

public void msgb()

{

if (frmFdMgr.CSN == "1") lblCui.Text = "ENGLISH";

else if (frmFdMgr.CSN == "2") lblCui.Text = "ITALIAN";

else if (frmFdMgr.CSN == "3") lblCui.Text = "INDIAN";

else if (frmFdMgr.CSN == "4") lblCui.Text = "CHINESE";

else if (frmFdMgr.CSN == "5") lblCui.Text = "JAPANESE";

else if (frmFdMgr.CSN == "6") lblCui.Text = "DRINKS";

if (frmFdMgr.CTG == "1") lblCatg.Text = "NON VEG";

else if (frmFdMgr.CTG == "2") lblCatg.Text = "VEG";

else if (frmFdMgr.CTG == "3") lblCatg.Text = "VEGAN";

else if (frmFdMgr.CTG == "4") lblCatg.Text = "DESSERT";

if (frmFdMgr.TPY == "1") lblTp.Text = "CHICKEN";

else if (frmFdMgr.TPY == "2") lblTp.Text = "BEEF";

else if (frmFdMgr.TPY == "3") lblTp.Text = "PORK";

else if (frmFdMgr.TPY == "4") lblTp.Text = "LAMB";

else if (frmFdMgr.TPY == "5") lblTp.Text = "SEAFOOD";

pBxRcp.Image = null;

}

private void DrpBxStps\_SelectionChangeCommitted(object sender, EventArgs e)

{

// To get the selected value between 1..10

string s = DrpBxStps.SelectedItem.ToString().Trim();

int cnt = int.Parse(s.Trim());

stp = int.Parse(s.Trim());

TextBox[] a = new TextBox[cnt];

try

{

int pointX = 3;

int pointY = 3;

pnlSteps.Controls.Clear();

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

{

if (i < 5)

{

a[i] = new TextBox();

a[i].Text = (i + 1).ToString();

a[i].Multiline = true;

a[i].AutoSize = false;

a[i].Location = new Point(pointX, pointY);

a[i].Size = new System.Drawing.Size(590, 90);

a[i].Name = "Pst" + nct.ToString().Trim();

pnlSteps.Controls.Add(a[i]);

pnlSteps.Show();

pointY += 95;

nct = nct + 1;

}

else

{

pointX = 605;

if (i ==5) pointY = 3;

a[i] = new TextBox();

a[i].Text = (i + 1).ToString();

a[i].Multiline = true;

a[i].AutoSize = false;

a[i].Location = new Point(pointX, pointY);

a[i].Size = new System.Drawing.Size(590, 90);

a[i].Name = "Pst" + nct.ToString().Trim();

pnlSteps.Controls.Add(a[i]);

pnlSteps.Show();

pointY += 95;

nct = nct + 1;

}

}

}

catch (Exception)

{

MessageBox.Show(e.ToString());

}

DrpBxStps.Enabled = false;

}

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

{

// Add Dynamic Input boxes

pnlIngrt.Visible = true;

lblqty.Visible=true;

lblUOM.Visible=true;

lblitms.Visible=true;

int pX1 =0, pX2 =66 ,pX3=134;

int pY = 3, i=0;

TextBox[] Q = new TextBox[25];

ComboBox[] U = new ComboBox[25];

TextBox[] Itm = new TextBox[25];

for (i=0; i<18;i++)

{

// Add new text box for Quantity input

Q[i] = new TextBox();

Q[i].AutoSize = false;

Q[i].Location = new Point(pX1, pY);

Q[i].Size = new System.Drawing.Size(60, 25);

Q[i].Name = "QtyBx"+i.ToString();

pnlIngrt.Controls.Add(Q[i]);

// Add text box for Items input

Itm[i] = new TextBox();

Itm[i].AutoSize = false;

Itm[i].Location = new Point(pX3, pY);

Itm[i].Size = new System.Drawing.Size(175, 25);

Itm[i].Name = "ItmsBx" + i.ToString();

pnlIngrt.Controls.Add(Itm[i]);

// Add Combo box for selecting UOM

U[i] = new ComboBox();

U[i].Location = new Point(pX2, pY);

U[i].Size = new System.Drawing.Size(68, 25);

U[i].Name = "UOMBx" + i.ToString();

ARSqlconn.Open();

SQLiteCommand command = new SQLiteCommand(ARSqlconn);

command.CommandText = "SELECT Abbv from UOM";

command.ExecuteNonQuery();

SQLiteDataReader reader = command.ExecuteReader();

while (reader.Read())

{

U[i].Items.Add(reader["Abbv"]);

}

ARSqlconn.Close();

pnlIngrt.Controls.Add(U[i]);

pnlIngrt.Show();

pY += 30;

}

}

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

{

int Rid;

// To save the details in receipe and Ingredients tables

ARSqlconn.Open();

SQLiteCommand command = new SQLiteCommand(ARSqlconn);

command.CommandText = "SELECT R\_ID from Counter";

command.Prepare();

command.ExecuteNonQuery();

Rid = (Int32)command.ExecuteScalar();

Rid = Rid + 1;

string NRid = "RCP0" + Rid.ToString().Trim();

ARSqlconn.Close();

// To save the details in Receipe table

string sta = "Active";

string Tdt = DateTime.Now.ToString("yyyy/MM/dd");

//txtAddRcpNm

//Find Cuisine Name

string Cnam=" ", Ftyp=" " , subty=" ";

if (frmFdMgr.CSN == "1") Cnam = "C01";

else if (frmFdMgr.CSN == "2") Cnam = "C02";

else if (frmFdMgr.CSN == "3") Cnam = "C03";

else if (frmFdMgr.CSN == "4") Cnam = "C04";

else if (frmFdMgr.CSN == "5") Cnam = "C05";

else if (frmFdMgr.CSN == "6")

{

Cnam = "C06";

Ftyp = "T05";

subty = "ST10";

}

if (frmFdMgr.CTG == "1") Ftyp = "T03";

else if (frmFdMgr.CTG == "2")

{

Ftyp = "T01";

subty = "ST07";

}

else if (frmFdMgr.CTG == "3")

{

Ftyp = "T02";

subty = "ST08";

}

else if (frmFdMgr.CTG == "4")

{

Ftyp = "T04";

subty = "ST09";

}

if (frmFdMgr.TPY == "1") subty = "ST01";

else if (frmFdMgr.TPY == "2") subty = "ST04";

else if (frmFdMgr.TPY == "3") subty = "ST06";

else if (frmFdMgr.TPY == "4") subty = "ST03";

else if (frmFdMgr.TPY == "5") subty = "ST05";

string yld = txtAddYld.Text.Trim();

string ctm = txtAddCt.Text.Trim();

string ninf = txtAddCal.Text.Trim();

string Rinf = txtAddIfo.Text.Trim();

// Open connection to Rceipe table and insert new record

ARSqlconn.Open();

command.CommandText = ("INSERT INTO Receipe VALUES(@R1, @R2, @R3, @R4, @R5, @R6, @R7, @R8, @R9, @R10, @R11, @R12, @R13, @R14)");

command.Parameters.AddWithValue("@R1", NRid);

command.Parameters.AddWithValue("@R2", sta);

command.Parameters.AddWithValue("@R3", Tdt);

command.Parameters.AddWithValue("@R4", txtAddRcpNm.Text.Trim());

command.Parameters.AddWithValue("@R5", Cnam);

command.Parameters.AddWithValue("@R6", Ftyp);

command.Parameters.AddWithValue("@R7", subty);

command.Parameters.AddWithValue("@R8", yld);

command.Parameters.AddWithValue("@R9", ctm);

command.Parameters.AddWithValue("@R10", rat);

command.Parameters.AddWithValue("@R11", ninf);

command.Parameters.AddWithValue("@R12", Rinf);

command.Parameters.AddWithValue("@R13", stp);

command.Parameters.AddWithValue("@R14", " ");

command.ExecuteNonQuery();

// Updating Counter table

command.CommandText = ("UPDATE Counter SET R\_ID=@RD");

command.Parameters.AddWithValue("@RD", Rid);

command.ExecuteNonQuery();

ARSqlconn.Close();

// Updating the Prepration Steps table

nct = 0;

foreach (Control childControl in pnlSteps.Controls)

{

if (childControl is TextBox)

{

nct = nct + 1;

TextBox box = (TextBox)childControl;

//Retrieve and store value from Panel Steps -> childControl.Text

Psteps[nct-1] = box.Text;

}

}

//Open connection to Prepare Steps Table

ARSqlconn.Open();

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

{

command.CommandText = ("INSERT INTO PrepareSteps VALUES(@R1, @R2, @R3)");

command.Parameters.AddWithValue("@R1", NRid);

command.Parameters.AddWithValue("@R2", i+1);

command.Parameters.AddWithValue("@R3", Psteps[i]);

command.ExecuteNonQuery();

}

ARSqlconn.Close();

// Retrieve Qty and Item text box values from Panel Ingredients

//public static string[,] IngrDet = new string[18, 3];

//public static int nct = 1, Qct = 0, Uct = 0, ITct = 0;

List<TextBox> listI = pnlIngrt.Controls.Cast<Control>().Where(i => i.GetType() == typeof(TextBox)).Cast<TextBox>().ToList();

foreach (TextBox tb1 in listI)

{

if (tb1.Name.Contains("ItmsBx"))

{

if (String.IsNullOrEmpty(tb1.Text.Trim()))

{

//Text Box is not empty

}

else

{

IngrDet[ITct, 3] = tb1.Text;

ITct = ITct + 1;

}

}

}

List<TextBox> list = pnlIngrt.Controls.Cast<Control>().Where(i => i.GetType() == typeof(TextBox)).Cast<TextBox>().ToList();

foreach (TextBox tb in list)

{

if (tb.Name.Contains("QtyBx"))

{

if (String.IsNullOrEmpty(tb.Text.Trim()))

{

//Text Box is not empty

}

else

{

IngrDet[Qct, 1] = tb.Text;

MessageBox.Show(IngrDet[Qct, 1]);

Qct = Qct + 1;

}

}

}

// Retrieve Uom combo box values from panel Ingredients

string abv;

foreach (Control childControl in pnlIngrt.Controls)

{

if (childControl is ComboBox)

{

ComboBox Cbox = (ComboBox)childControl;

//Retrieve and store value from Panel Steps -> childControl.Text

if (String.IsNullOrEmpty(Cbox.Text.Trim()))

{

//No task

}

else

{

abv = Cbox.Text.Trim();

// Got the Abbv... with that get the UOM\_ID-Name-Abbv

ARSqlconn.Open();

command.CommandText = ("SELECT UOM\_ID from UOM WHERE Abbv= @Um");

command.Parameters.AddWithValue("@Um", abv.Trim());

command.Prepare();

command.ExecuteNonQuery();

IngrDet[Uct, 2] = (string)command.ExecuteScalar();

ARSqlconn.Close();

Uct = Uct + 1;

}

}

}

// To Insert the gathered values into Ingredients table

ARSqlconn.Open();

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

{

command.CommandText = ("INSERT INTO Ingredients VALUES(@I1, @I2, @I3, @I4)");

command.Parameters.AddWithValue("@I1", NRid);

command.Parameters.AddWithValue("@I2", IngrDet[i, 2]);

command.Parameters.AddWithValue("@I3", IngrDet[i, 1]);

command.Parameters.AddWithValue("@I4", IngrDet[i, 3]);

command.ExecuteNonQuery();

}

ARSqlconn.Close();

MessageBox.Show("The New Receipe "+ NRid + " - "+ txtAddRcpNm.Text.Trim()+ " has been added");

}

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

{

MessageBox.Show("Total Quantity Rows are " + Qct.ToString());

MessageBox.Show("Total Items Rows are " + ITct.ToString());

}

private void cBxRat\_SelectionChangeCommitted(object sender, EventArgs e)

{

string s = cBxRat.SelectedItem.ToString().Trim();

if (s== "\*") rat = 1;

else if (s == "\*\*") rat = 2;

else if (s == "\*\*\*") rat = 3;

else if (s == "\*\*\*\*") rat = 4;

else if (s == "\*\*\*\*\*") rat = 5;

}

}

}

Modify

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Data.SQLite;

using System.Drawing;

using System.Drawing.Drawing2D;

using System.Linq;

using System.Reflection;

using System.Reflection.Emit;

using System.Runtime.Intrinsics.Arm;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Xml.Linq;

using static System.ComponentModel.Design.ObjectSelectorEditor;

using static System.Net.Mime.MediaTypeNames;

using static System.Net.WebRequestMethods;

using static System.Windows.Forms.VisualStyles.VisualStyleElement.ListView;

namespace FoodManagerReceipr

{

public partial class FrmModifyRecipe : Form

{

public string dbfile = "URI=file:FoodManagerReceipe.db";

public SQLiteConnection ARSqlconn;

public SQLiteCommand ARCmd;

public SQLiteDataReader ARDataReader;

public SQLiteDataAdapter ARDap;

public DataTable ARTbl;

public static int rat, stp;

public static string[] Psteps = new string[10];

public static string RccPID;

public static string[,] IngrDet = new string[15, 4];

public static int nct = 0, Qct = 0, Uct = 0, ITct = 0;

public static int idx = 0;

public FrmModifyRecipe()

{

InitializeComponent();

ARSqlconn = new SQLiteConnection(dbfile);

this.WindowState = FormWindowState.Maximized;

}

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

{

msgb();

FillRecpNamGrd();

// Add emtpty to each cells

for (int i =0; i < 15; i++)

for(int j =0; j < 4; j++)

{

IngrDet[i , j] = " ";

}

}

public void FillRecpNamGrd()

{

// Fill Listbox

lBxRcpnam.Items.Clear();

ARSqlconn.Open();

SQLiteCommand command = new SQLiteCommand(ARSqlconn);

// Show Receipe as per selection

//frmFdMgr.CSN, frmFdMgr.CTG, frmFdMgr.TPY

string Cun = frmFdMgr.CSN, Cat = frmFdMgr.CTG, Ty = frmFdMgr.TPY;

if (Cun == "1" && Cat == "1" && Ty == "1") // English - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "1" && Ty == "2") // English - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "1" && Ty == "3") // English - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

//MessageBox.Show(" Display - Non Veg - Pork Selected");

}

else if (Cun == "1" && Cat == "1" && Ty == "4") // English - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "1" && Ty == "5") // English - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "2") // English - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "3") // English - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "1" && Cat == "4") // English - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "2" && Cat == "1" && Ty == "1") // Italian - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "2") // Italian - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "3") // Italian - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C01");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "4") // Italian - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "1" && Ty == "5") // Italian - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "2") // Italian - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "3") // Italian - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "2" && Cat == "4") // Italian - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C02");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "3" && Cat == "1" && Ty == "1") // Indian - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "2") // Indian - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "3") // Indian - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "4") // Indian - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "1" && Ty == "5") // Indian - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID AND Status = @stus ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "2") // Indian - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "3") // Indian - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "3" && Cat == "4") // Indian - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C03");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "4" && Cat == "1" && Ty == "1") // Chinese - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "2") // Chinese - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "3") // Chinese - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "4") // Chinese - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "1" && Ty == "5") // Chinese - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "2") // Chinese - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "3") // Chinese - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "4" && Cat == "5") // Chinese - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "5" && Cat == "1" && Ty == "1") // Japanese - Non Veg - Chicken

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST01");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "2") // Japanese - Non Veg - Beef

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C04");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST04");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "3") // Japanese - Non Veg - Pork

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST06");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "4") // Japanese - Non Veg - Lamb

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST03");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "1" && Ty == "5") // Japanese - Non Veg - Seafood

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T03");

command.Parameters.AddWithValue("@sID", "ST05");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "2") // Japanese - Veg

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T01");

command.Parameters.AddWithValue("@sID", "ST07");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "3") // Japanese - Vegan

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T02");

command.Parameters.AddWithValue("@sID", "ST08");

command.Parameters.AddWithValue("@stus", "Active");

}

else if (Cun == "5" && Cat == "5") // Japanese - Dessert

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C05");

command.Parameters.AddWithValue("@tID", "T04");

command.Parameters.AddWithValue("@sID", "ST09");

command.Parameters.AddWithValue("@stus", "Active");

}

if (Cun == "6") // Drink

{

command.CommandText = "select RecipeName from Receipe Where Cus\_ID = @cID AND Typ\_ID = @tID AND SubCat\_ID = @sID ORDER BY RecipeName ASC"; ;

command.Parameters.AddWithValue("@cID", "C06");

command.Parameters.AddWithValue("@tID", "T05");

command.Parameters.AddWithValue("@sID", "ST10");

command.Parameters.AddWithValue("@stus", "Active");

}

command.ExecuteNonQuery();

SQLiteDataReader reader = command.ExecuteReader();

while (reader.Read())

{

lBxRcpnam.Items.Add(reader["RecipeName"]);

}

ARSqlconn.Close();

}

//On double click - Display the Receipe details

public void msgb()

{

if (frmFdMgr.CSN == "1") lblcus.Text = "ENGLISH";

else if (frmFdMgr.CSN == "2") lblcus.Text = "ITALIAN";

else if (frmFdMgr.CSN == "3") lblcus.Text = "INDIAN";

else if (frmFdMgr.CSN == "4") lblcus.Text = "CHINESE";

else if (frmFdMgr.CSN == "5") lblcus.Text = "JAPANESE";

else if (frmFdMgr.CSN == "6") lblcus.Text = "DRINKS";

if (frmFdMgr.CTG == "1") lblCat.Text = "NON VEG";

else if (frmFdMgr.CTG == "2") lblCat.Text = "VEG";

else if (frmFdMgr.CTG == "3") lblCat.Text = "VEGAN";

else if (frmFdMgr.CTG == "4") lblCat.Text = "DESSERT";

if (frmFdMgr.TPY == "1") lbltyp.Text = "CHICKEN";

else if (frmFdMgr.TPY == "2") lbltyp.Text = "BEEF";

else if (frmFdMgr.TPY == "3") lbltyp.Text = "PORK";

else if (frmFdMgr.TPY == "4") lbltyp.Text = "LAMB";

else if (frmFdMgr.TPY == "5") lbltyp.Text = "SEAFOOD";

}

//int RwId;

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

{

// When double click the row in the Grid table it should fetch the values in the text box

// Blank All Label Cell

Text1.Text = "";

Text2.Text = "";

Text3.Text = "";

Text4.Text = "";

Text5.Text = "";

Text6.Text = "";

Text7.Text = "";

Text8.Text = "";

Text9.Text = "";

Text10.Text = "";

Text1.Visible = false;

Text2.Visible = false;

Text3.Visible = false;

Text4.Visible = false;

Text5.Visible = false;

Text6.Visible = false;

Text7.Visible = false;

Text8.Visible = false;

Text9.Visible = false;

Text10.Visible = false;

//File the textbox / label box and radio button values with the row selected

if (lBxRcpnam.SelectedItem != null)

{

TxtRnm.Text = lBxRcpnam.SelectedItem.ToString();

}

string rcname = TxtRnm.Text.Trim();

ARSqlconn.Open();

ARCmd = new SQLiteCommand(ARSqlconn);

ARCmd.CommandText = "SELECT Rcp\_ID from Receipe WHERE RecipeName = @RCiD";

ARCmd.Parameters.AddWithValue("@RCiD", rcname);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

string Rcp = (string)ARCmd.ExecuteScalar();

RccPID = Rcp;

ARDap = new SQLiteDataAdapter("select \* from Receipe WHERE Rcp\_ID='" + Rcp + "'", ARSqlconn);

ARSqlconn.Close();

SQLiteCommandBuilder ARCdB1 = new SQLiteCommandBuilder(ARDap);

ARTbl = new DataTable();

ARDap.Fill(ARTbl);

dGWRcnm.DataSource = ARTbl;

string Cui = dGWRcnm.Rows[0].Cells["Cus\_ID"].Value.ToString();

string Cta = dGWRcnm.Rows[0].Cells["Typ\_ID"].Value.ToString();

string Tpy = dGWRcnm.Rows[0].Cells["SubCat\_ID"].Value.ToString();

TxtCal.Text = dGWRcnm.Rows[0].Cells["NutritionInfo"].Value.ToString();

TxtYld.Text = dGWRcnm.Rows[0].Cells["Yield"].Value.ToString();

TxtCt.Text = dGWRcnm.Rows[0].Cells["CookTime"].Value.ToString();

int rtr = int.Parse(dGWRcnm.Rows[0].Cells["Rating"].Value.ToString());

string rrat = dGWRcnm.Rows[0].Cells["Rating"].Value.ToString().Trim();

//rat = rtr;

//MessageBox.Show("Rating is " + rtr.ToString());

if (rrat == "1")

cBxRat.SelectedIndex = cBxRat.FindStringExact("\*");

else if (rrat == "2")

cBxRat.SelectedIndex = cBxRat.FindStringExact("\*\*");

else if (rrat == "3")

cBxRat.SelectedIndex = cBxRat.FindStringExact("\*\*\*");

else if (rrat == "4")

cBxRat.SelectedIndex = cBxRat.FindStringExact("\*\*\*\*");

else if (rrat == "5")

cBxRat.SelectedIndex = cBxRat.FindStringExact("\*\*\*\*\*");

TxtRI.Text = dGWRcnm.Rows[0].Cells["receipeInfo"].Value.ToString();

pBxRcp.ImageLocation = @"D:\SHU\FoodManagerReceipr\data resorces\" + dGWRcnm.Rows[0].Cells["Pic"].Value.ToString();

txtimgpic.Text = dGWRcnm.Rows[0].Cells["Pic"].Value.ToString();

// Display Ingredients details for selected Receipe

SQLiteCommand cmd = new SQLiteCommand(

@"select

Ingredients.OUM,

UOM.Abbv,

Ingredients.Items

from Ingredients

inner join UOM on Ingredients.UOM\_ID =

UOM.UOM\_ID

where Ingredients.Rcp\_ID= @RiD", ARSqlconn);

cmd.Parameters.AddWithValue("@RiD", Rcp);

cmd.CommandType = System.Data.CommandType.Text;

SQLiteDataAdapter sda = new SQLiteDataAdapter(cmd);

SQLiteCommandBuilder ARCdB2 = new SQLiteCommandBuilder(sda);

ARTbl = new DataTable();

sda.Fill(ARTbl);

dGVIngdt.DataSource = ARTbl;

Int32 index = dGVIngdt.Rows.Count - 1;

int vi = 0;

//Copy Gridview details inot string array

foreach (DataGridViewRow r in dGVIngdt.Rows)

{

if (vi < index)

{

IngrDet[vi, 0] = r.Cells[0].Value.ToString();

IngrDet[vi, 1] = r.Cells[1].Value.ToString();

IngrDet[vi, 2] = r.Cells[2].Value.ToString();

vi = vi + 1;

}

}

// Add Dynamic test and combo box and dispplay current ingredients list

int pX1 = 0, pX2 = 66, pX3 = 134;

int pY = 3, i = 0;

TextBox[] Q = new TextBox[25];

ComboBox[] U = new ComboBox[25];

TextBox[] Itm = new TextBox[25];

for (i = 0; i < 15; i++)

{

// Add new text box for Quantity input

Q[i] = new TextBox();

Q[i].AutoSize = false;

Q[i].Location = new Point(pX1, pY);

Q[i].Size = new System.Drawing.Size(60, 25);

Q[i].Name = "QtyBx" + i.ToString();

Q[i].Text = IngrDet[i, 0];

pnlIngrt.Controls.Add(Q[i]);

// Add text box for Items input

Itm[i] = new TextBox();

Itm[i].AutoSize = false;

Itm[i].Location = new Point(pX3, pY);

Itm[i].Size = new System.Drawing.Size(175, 25);

Itm[i].Name = "ItmsBx" + i.ToString();

Itm[i].Text = IngrDet[i, 2];

pnlIngrt.Controls.Add(Itm[i]);

// Add Combo box for selecting UOM

U[i] = new ComboBox();

U[i].Location = new Point(pX2, pY);

U[i].Size = new System.Drawing.Size(68, 25);

U[i].Name = "UOMBx" + i.ToString();

ARSqlconn.Open();

SQLiteCommand command = new SQLiteCommand(ARSqlconn);

command.CommandText = "SELECT Abbv from UOM";

command.ExecuteNonQuery();

SQLiteDataReader reader = command.ExecuteReader();

while (reader.Read())

{

U[i].Items.Add(reader["Abbv"]);

}

U[i].SelectedIndex = U[i].FindStringExact(IngrDet[i, 1]);

ARSqlconn.Close();

pnlIngrt.Controls.Add(U[i]);

pnlIngrt.Show();

pY += 30;

}

ARSqlconn.Open();

ARCmd = new SQLiteCommand(ARSqlconn);

ARCmd.CommandText = "SELECT CuisineName FROM Cuisines WHERE Cuisines.Cus\_ID = @CiD";

ARCmd.Parameters.AddWithValue("@CiD", Cui);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

lblCui.Text = (string)ARCmd.ExecuteScalar();

ARCmd.CommandText = "SELECT FoodType FROM Types WHERE Types.Typ\_ID = @TiD";

ARCmd.Parameters.AddWithValue("@TiD", Cta);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

lblCatg.Text = (string)ARCmd.ExecuteScalar();

ARCmd.CommandText = "SELECT SubCat FROM Sub\_Category WHERE Sub\_Category.SubCat\_ID = @SiD";

ARCmd.Parameters.AddWithValue("@SiD", Tpy);

ARCmd.Prepare();

ARCmd.ExecuteNonQuery();

lblTp.Text = (string)ARCmd.ExecuteScalar();

ARCmd.CommandText = "SELECT count(\*) FROM PrepareSteps WHERE PrepareSteps.Rcp\_ID = @RiD";

ARCmd.Parameters.AddWithValue("@RiD", Rcp);

int cnt = Convert.ToInt32(ARCmd.ExecuteScalar());

ARDap = new SQLiteDataAdapter("SELECT Preparation\_Details FROM PrepareSteps WHERE PrepareSteps.Rcp\_ID ='" + Rcp + "'", ARSqlconn);

ARSqlconn.Close();

SQLiteCommandBuilder ARCdB = new SQLiteCommandBuilder(ARDap);

ARTbl = new DataTable();

ARDap.Fill(ARTbl);

dGWStp.DataSource = ARTbl;

Text1.Visible = true;

Text2.Visible = true;

Text3.Visible = true;

Text4.Visible = true;

Text5.Visible = true;

Text6.Visible = true;

Text7.Visible = true;

Text8.Visible = true;

Text9.Visible = true;

Text10.Visible = true;

foreach (DataGridViewRow row in dGWStp.Rows)

{

if (idx == 0 && idx < cnt) Text1.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 1 && idx < cnt) Text2.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 2 && idx < cnt) Text3.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 3 && idx < cnt) Text4.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 4 && idx < cnt) Text5.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 5 && idx < cnt) Text6.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 6 && idx < cnt) Text7.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 7 && idx < cnt) Text8.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 8 && idx < cnt) Text9.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

else if (idx == 9 && idx < cnt) Text10.Text = dGWStp.Rows[idx].Cells["Preparation\_Details"].Value.ToString();

idx++;

}

idx = idx - 1;

//MessageBox.Show("Total Steps are " + idx.ToString());

}

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

{

// To update the modified details to receipe/Prepration Steps and Ingrdients table

// Updating the Prepration Steps table

int nnct = 0;

if (Text1.Text.Length != 0) { Psteps[0] = Text1.Text; nnct = 0; }

if (Text2.Text.Length != 0) { Psteps[1] = Text2.Text; nnct = 1; }

if (Text3.Text.Length != 0) { Psteps[2] = Text3.Text; nnct = 2; }

if (Text4.Text.Length != 0) { Psteps[3] = Text4.Text; nnct = 3; }

if (Text5.Text.Length != 0) { Psteps[4] = Text5.Text; nnct = 4; }

if (Text6.Text.Length != 0) { Psteps[5] = Text6.Text; nnct = 5; }

if (Text7.Text.Length != 0) { Psteps[6] = Text7.Text; nnct = 6; }

if (Text8.Text.Length != 0) { Psteps[7] = Text8.Text; nnct = 7; }

if (Text9.Text.Length != 0) { Psteps[8] = Text9.Text; nnct = 8; }

if (Text10.Text.Length != 0) { Psteps[9] = Text10.Text; nnct = 9; }

nct = nnct;

//MessageBox.Show("Total Steps = " + int.Parse((nct+1).ToString()));

ARSqlconn.Open();

SQLiteCommand command = new SQLiteCommand(ARSqlconn);

//Find Cuisine Name

string Cnam = " ", Ftyp = " ", subty = " ";

if (frmFdMgr.CSN == "1") Cnam = "C01";

else if (frmFdMgr.CSN == "2") Cnam = "C02";

else if (frmFdMgr.CSN == "3") Cnam = "C03";

else if (frmFdMgr.CSN == "4") Cnam = "C04";

else if (frmFdMgr.CSN == "5") Cnam = "C05";

else if (frmFdMgr.CSN == "6")

{

Cnam = "C06";

Ftyp = "T05";

subty = "ST10";

}

if (frmFdMgr.CTG == "1") Ftyp = "T03";

else if (frmFdMgr.CTG == "2")

{

Ftyp = "T01";

subty = "ST07";

}

else if (frmFdMgr.CTG == "3")

{

Ftyp = "T02";

subty = "ST08";

}

else if (frmFdMgr.CTG == "4")

{

Ftyp = "T04";

subty = "ST09";

}

if (frmFdMgr.TPY == "1") subty = "ST01";

else if (frmFdMgr.TPY == "2") subty = "ST04";

else if (frmFdMgr.TPY == "3") subty = "ST06";

else if (frmFdMgr.TPY == "4") subty = "ST03";

else if (frmFdMgr.TPY == "5") subty = "ST05";

string yld = TxtYld.Text.Trim();

string ctm = TxtCt.Text.Trim();

string ninf = TxtCal.Text.Trim();

string Rinf = TxtRI.Text.Trim();

// Open connection to Rceipe table and insert new record

command.CommandText = ("UPDATE Receipe SET RecipeName=@R4, Cus\_ID=@R5, Typ\_ID=@R6, SubCat\_ID=@R7, Yield=@R8, CookTime=@R9, Rating=@R10, NutritionInfo=@R11, receipeInfo=@R12, Steps=@R13, Pic=@R14 where RCP\_ID = @R15");

command.Parameters.AddWithValue("@R4", TxtRnm.Text.Trim());

command.Parameters.AddWithValue("@R5", Cnam);

command.Parameters.AddWithValue("@R6", Ftyp);

command.Parameters.AddWithValue("@R7", subty);

command.Parameters.AddWithValue("@R8", yld);

command.Parameters.AddWithValue("@R9", ctm);

command.Parameters.AddWithValue("@R10", rat);

command.Parameters.AddWithValue("@R11", ninf);

command.Parameters.AddWithValue("@R12", Rinf);

command.Parameters.AddWithValue("@R13", nct+1);

command.Parameters.AddWithValue("@R14", txtimgpic.Text.Trim());

command.Parameters.AddWithValue("@R15", RccPID);

command.ExecuteNonQuery();

ARSqlconn.Close();

//Open connection to Prepare Steps Table

//First delete previous Receipe steps entries

ARSqlconn.Open();

command.CommandText = ("DELETE FROM PrepareSteps WHERE RCP\_ID=@R1");

command.Parameters.AddWithValue("@R1", RccPID);

command.ExecuteNonQuery();

//Now add the modifed prepration steps

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

{

command.CommandText = ("INSERT INTO PrepareSteps VALUES(@R1, @R2, @R3)");

command.Parameters.AddWithValue("@R1", RccPID);

command.Parameters.AddWithValue("@R2", i + 1);

command.Parameters.AddWithValue("@R3", Psteps[i]);

command.ExecuteNonQuery();

}

ARSqlconn.Close();

// Retrieve Qty and Item text box values from Panel Ingredients

//public static string[,] IngrDet = new string[18, 3];

//public static int nct = 1, Qct = 0, Uct = 0, ITct = 0;

List<TextBox> listI = pnlIngrt.Controls.Cast<Control>().Where(i => i.GetType() == typeof(TextBox)).Cast<TextBox>().ToList();

foreach (TextBox tb1 in listI)

{

if (tb1.Name.Contains("ItmsBx"))

{

if (String.IsNullOrEmpty(tb1.Text.Trim()))

{

//Text Box is not empty

}

else

{

IngrDet[ITct, 3] = tb1.Text;

ITct = ITct + 1;

}

}

}

List<TextBox> list = pnlIngrt.Controls.Cast<Control>().Where(i => i.GetType() == typeof(TextBox)).Cast<TextBox>().ToList();

foreach (TextBox tb in list)

{

if (tb.Name.Contains("QtyBx"))

{

if (String.IsNullOrEmpty(tb.Text.Trim()))

{

//Text Box is not empty

}

else

{

IngrDet[Qct, 1] = tb.Text;

//MessageBox.Show(IngrDet[Qct, 1]);

Qct = Qct + 1;

}

}

}

// Retrieve Uom combo box values from panel Ingredients

string abv;

foreach (Control childControl in pnlIngrt.Controls)

{

if (childControl is ComboBox)

{

ComboBox Cbox = (ComboBox)childControl;

//Retrieve and store value from Panel Steps -> childControl.Text

if (String.IsNullOrEmpty(Cbox.Text.Trim()))

{

//No task

}

else

{

abv = Cbox.Text.Trim();

// Got the Abbv... with that get the UOM\_ID-Name-Abbv

ARSqlconn.Open();

command.CommandText = ("SELECT UOM\_ID from UOM WHERE Abbv= @Um");

command.Parameters.AddWithValue("@Um", abv.Trim());

command.Prepare();

command.ExecuteNonQuery();

IngrDet[Uct, 2] = (string)command.ExecuteScalar();

ARSqlconn.Close();

Uct = Uct + 1;

}

}

}

// To Insert the gathered values into Ingredients table

ARSqlconn.Open();

// !st delete the existing receipt Ingredients details then add again with modified details

command.CommandText = ("DELETE FROM Ingredients WHERE RCP\_ID=@I1");

command.Parameters.AddWithValue("@I1", RccPID);

command.ExecuteNonQuery();

// Now add the modified Ingredients details

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

{

command.CommandText = ("INSERT INTO Ingredients VALUES(@I1, @I2, @I3, @I4)");

command.Parameters.AddWithValue("@I1", RccPID);

command.Parameters.AddWithValue("@I2", IngrDet[i, 2]);

command.Parameters.AddWithValue("@I3", IngrDet[i, 1]);

command.Parameters.AddWithValue("@I4", IngrDet[i, 3]);

command.ExecuteNonQuery();

}

ARSqlconn.Close();

MessageBox.Show("The Receipe " + RccPID + " - " + TxtRnm.Text.Trim() + " has been modified successfuly");

}

private void cBxRat\_SelectionChangeCommitted(object sender, EventArgs e)

{

string s = cBxRat.SelectedItem.ToString().Trim();

if (s == "\*") rat = 1;

else if (s == "\*\*") rat = 2;

else if (s == "\*\*\*") rat = 3;

else if (s == "\*\*\*\*") rat = 4;

else if (s == "\*\*\*\*\*") rat = 5;

}

}

}