|  |
| --- |
| RESTORAN OTOMASYONU |
| NESNE TABANLI PROGRAMLAMA PROJE RAPORU |
| HAZIRLAYAN AZİZ EMİR YAVUZ 1355 |

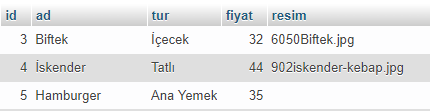
13.05.2022

**VEritabani ve tablolar**

# Tablo #1 “Yemekler”



# Tablo İçinden Örnek :



*Restoran Otomasyonu projemin ilk aşaması bir veritabanı oluşturmaktı.*

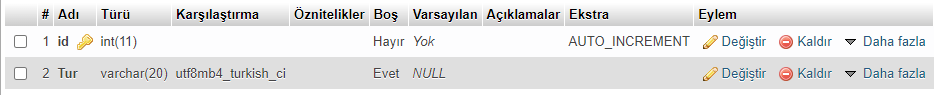
*“dbrestorant “ adında veritabanımı oluşturduktan sonra “Yemekler” adında ilk tablomu hazırladım.*

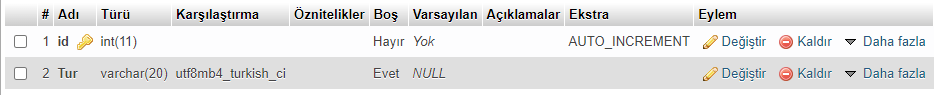
*Ardından sıra içini doldurmaktaydı. “id”,”ad”,”tur”,”fiyat” ve “resim” olucak şekilde kullanacağım nitelikleri hazırladım. İd’ye birincil anahtar ve otomatik artış özellerini verdim.*

*ad,tur ve resim’e 50 karakterlik bir metin özelliği verdim. fiyat’a ise 11 hanelik bir sayı verilebilecek şekilde ayarladım.*

*Ürünün (yemek) tabloya kayıt edildiği zaman bir fotoğrafı olmayabilir bu yüzden “NULL” yani “boş geçilebilir” özelliğini tanımladım.*

# Tablo #2 “Yemek Türü”



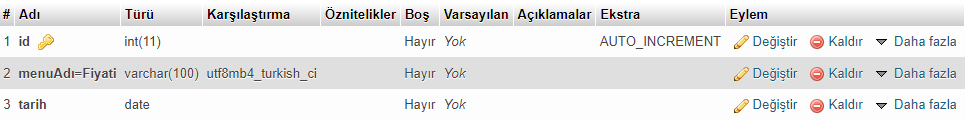


# Tablo İçinden Örnek :

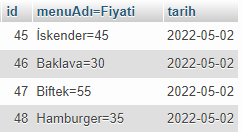


Bu tabloda yemek türünün gerekli tanımlamasını yaptım.

# Tablo #3 “Satislar”



# Tablo İçinden Örnek :



Yapılan satışları görmek amacı ile “satislar” adında tablomu oluşturdum.

Menü adını ve fiyatını tek bir sütünda, tarhihi ise ayrı bir sütüna oluşturdum.

# TAsarım ve kod

Uygulama Tasarımım 4 Formdan oluşuyor.

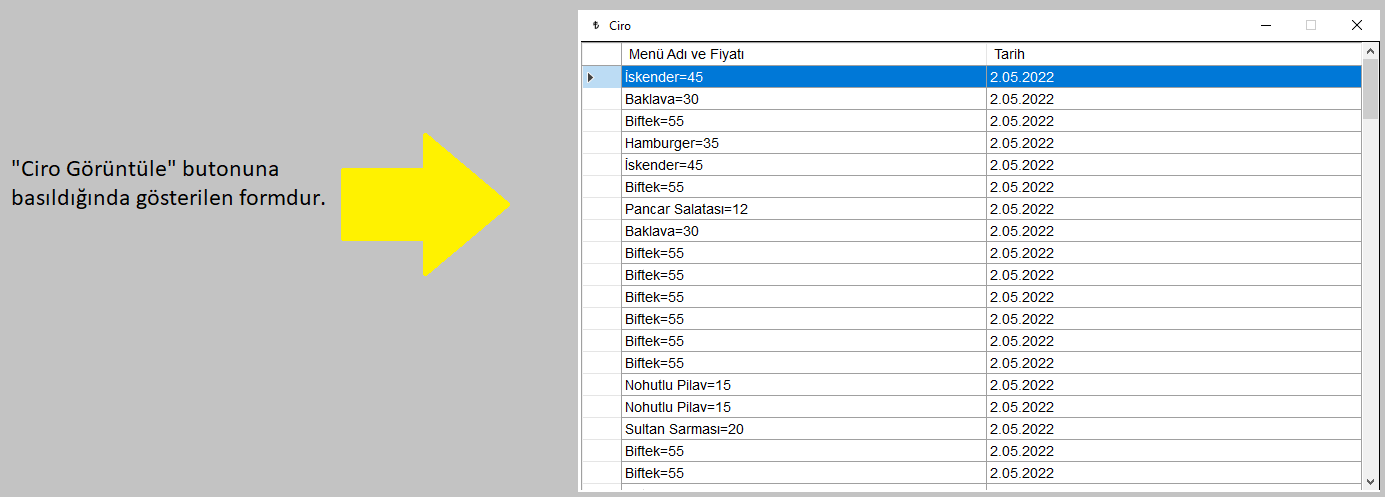
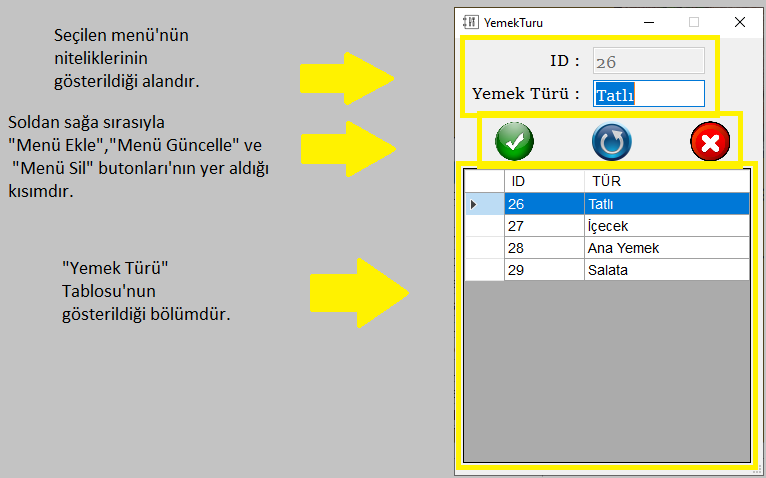
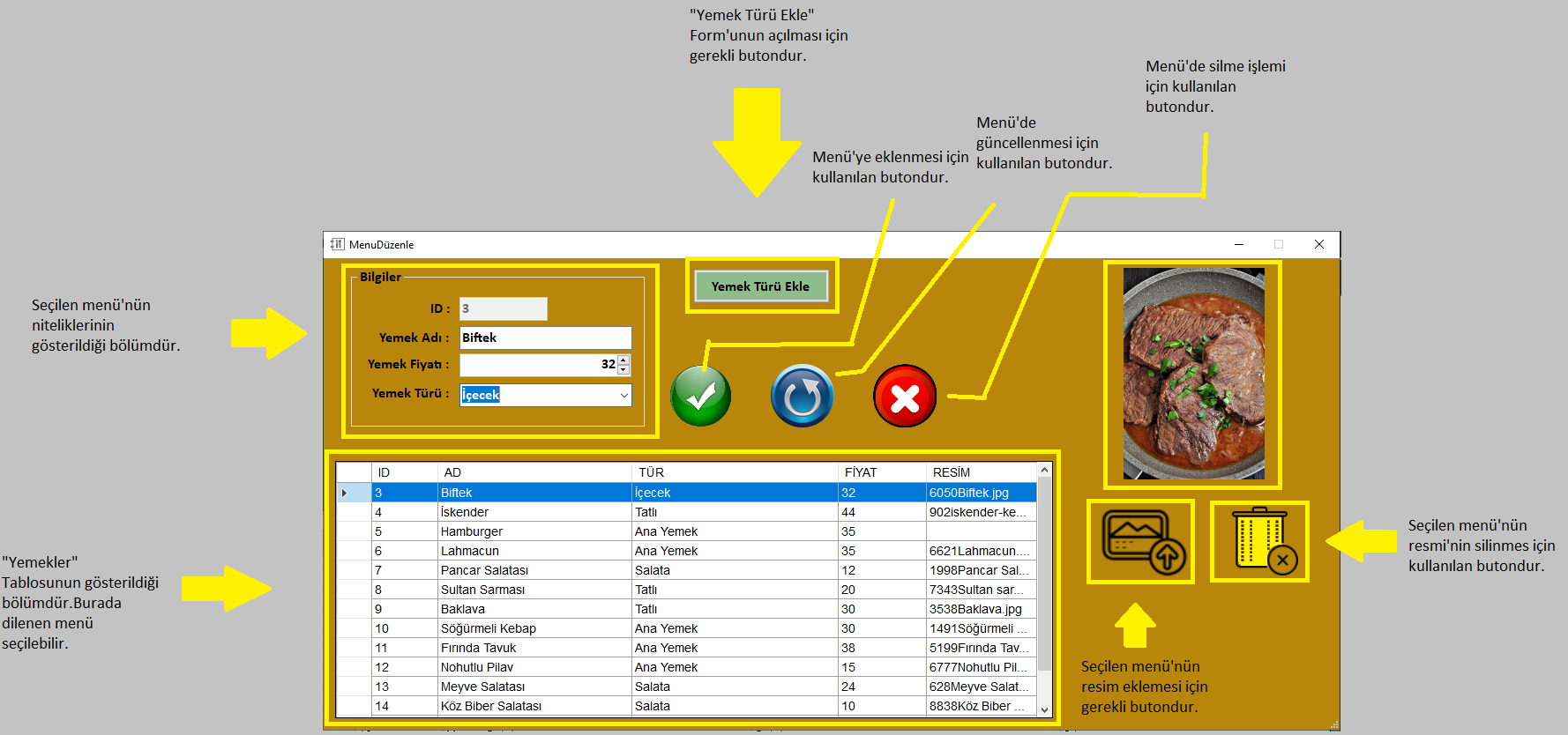
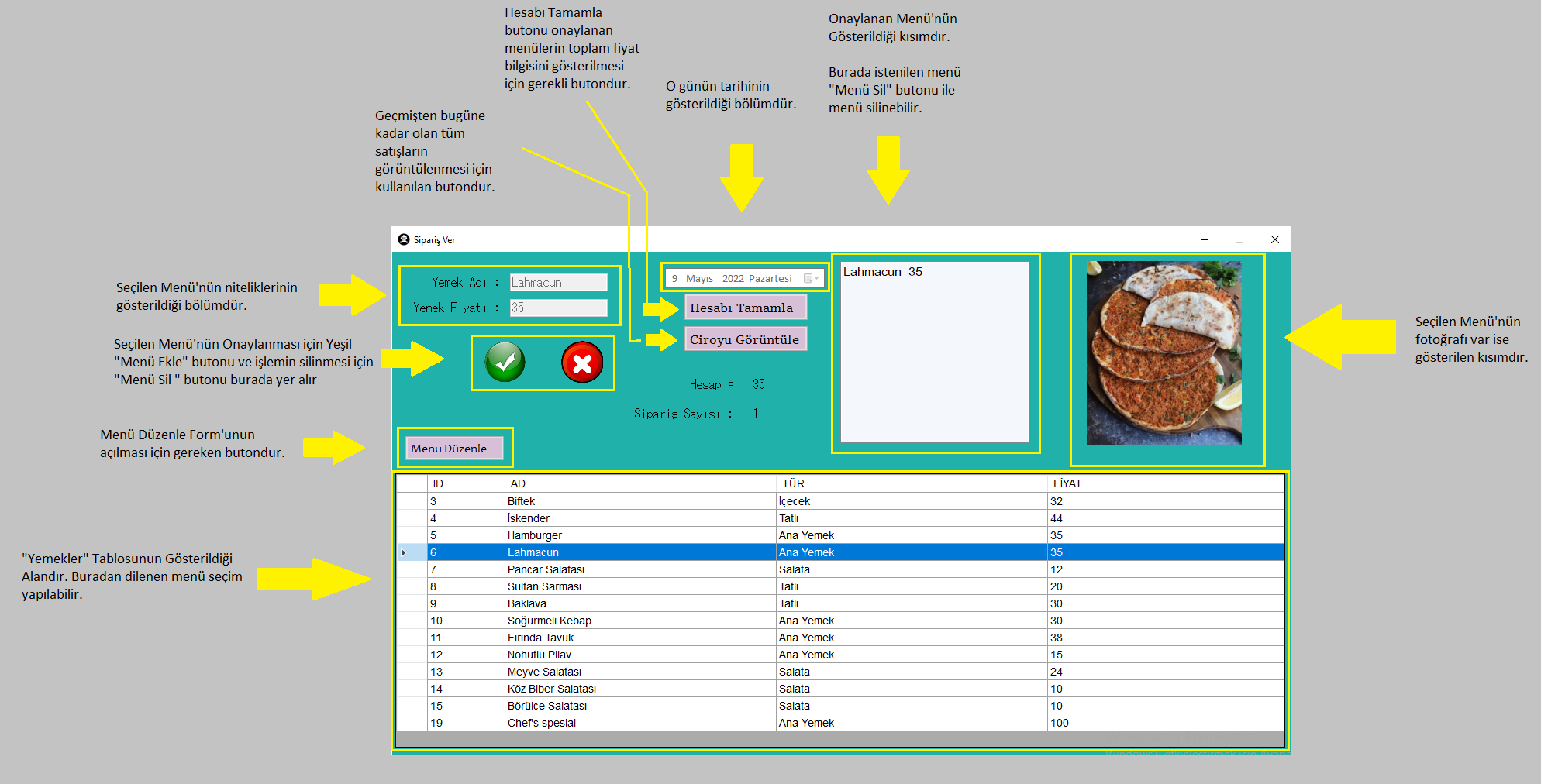
1.Sipariş Ver

2.Menü Düzenle

3.Yemek Türü

4.Ciro

Bu formların tasarım görüntüleri ve hangi nesnenin ne işe yaradığı aşağıdaki gibidir :



Oluşturduğum veritabanı’nı bağlamak için kullandığım kod bloğu aşağıdaki gibidir :

using MySql.Data;

using MySql.Data.MySqlClient;

class DBislemleri

{

MySqlConnection baglanti;

public MySqlConnection Baglan() // class içinde baglan adında fonksiyon oluşturuldu.

{

baglanti = new MySqlConnection("Server = localhost; Database = dbrestorant; Uid = root; Pwd = ;");

// Bağlantı yolu verildi.

baglanti.Open();// Bağlantı açıldı.

MySqlConnection.ClearPool(baglanti);// Bundan önceki bağlantılar temizlendi.

return baglanti;// Çağrıldığı yere bağlantı gönderildi.

}

public void BaglantiyiKapat()

{

baglanti.Close();

}

}

Kalan kod kısmı için gerekli kütüphaneleri çağırıyoruz

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using MySql.Data;

using MySql.Data.MySqlClient;

using System.IO;

using System.Collections;

İşlemlerde kullanacağım global değişkenlerimi tanımladım.

//Global Değişkenler

DBislemleri bag = new DBislemleri();// Bağlantı için oluşturulan class’tan türetilen bir nesne

DataTable dtYemekler = new DataTable();//Tablolara yüklemek için bir adet dtBirinci değişkeni tanımlandı.

DataTable dtMenu = new DataTable();//Tablolara yüklemek için bir adet dtBirinci değişkeni tanımlandı.

int siraNo = 0;

ArrayList yemekler = new ArrayList();

ArrayList fiyatlar = new ArrayList();

MySqlCommand kmt = new MySqlCommand();//İşlemler için bir MySqlCommand nesnesi tanımlandı.

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

       {

           dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

           dtpTarih.Value = DateTime.Now;

           dtpTarih.Enabled = false;

           txtFiyat.Enabled = false;

           txtYemek.Enabled = false;

           DataGridiDoldur();

           ToolTip btn\_aciklama = new ToolTip();

           btn\_aciklama.SetToolTip(btnSil, "Menu Sil");

           btn\_aciklama.SetToolTip(btnMenuEkle, "Menu Ekle");

           dgw1.Columns.Remove("resim");

           dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

       }

Form ekranım açılırken yapmak istediğim işlemleri Form\_load kısmına ekliyorum.Burada yapmak istediğim işlemler ; tarih göstergesini güncellemek,seçilen menü’nün niteliklerinin dışardan erişilebilir olmasını engellemek, Yemekler tablosu’nun veritabanından verileri çekmesini sağlamak ve butonların üzerine gelindiğinde ne işe yaradığına dair küçük bir yazı yazmaktır.

private void DataGridiDoldur()

{

dgw1.RowHeadersVisible = true;//Satır başlıkları görünmez yapılır.

dgw1.Font = new Font("Arial", 10);// Datagridview için font ayarlaması yapılır.

//dtBirinci.Columns.Clear();//Tablonun önceki kullanımından kalan sütunlar temizlenir.

dtYemekler.Clear(); // Tablo temizlenir.

MySqlDataAdapter adtr = new MySqlDataAdapter("select \* from yemekler", bag.Baglan());//Komut ve bağlantı yüklenen DataAdapter nesnesi

adtr.Fill(dtYemekler);//DataAdapter nesnesine yüklenen bilgiler tabloya aktarılır.

dgw1.DataSource = dtYemekler; //Datagridview bileşeni data kaynağı olarak dtBirinci seçilir.

dgw1.Columns["id"].HeaderText = "ID";

dgw1.Columns["ad"].HeaderText = "AD";

dgw1.Columns["ad"].Width = 350;

dgw1.Columns["fiyat"].HeaderText = "FİYAT";

dgw1.Columns["fiyat"].Width = 305;

dgw1.Columns["tur"].HeaderText = "TÜR";

dgw1.Columns["tur"].Width = 350;

adtr.Dispose();

try

{

dgw1.Rows[siraNo].Selected = true;

}

catch (Exception)

{

MessageBox.Show("Menü Boş", "Uyarı", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

VerileriDoldur();

bag.BaglantiyiKapat();

}

Verileri getirmek için kullandığım metot yukarıdaki şekilde gösterilmiştir.

private void VerileriDoldur()

{

if (siraNo >= 0 && siraNo < dtYemekler.Rows.Count)

{

txtYemek.Text = dtYemekler.Rows[siraNo]["ad"].ToString();

txtFiyat.Text = dtYemekler.Rows[siraNo]["fiyat"].ToString();

dtMenu.Columns.Clear();//Tablonun önceki kullanımından kalan sütunlar temizlenir.

dtMenu.Clear(); // Tablo temizlenir.

string resim = Directory.GetParent(Environment.CurrentDirectory).Parent.FullName + "\\Resimler\\";

if (dtYemekler.Rows[siraNo]["resim"].ToString() == "")

{

resim += "no-image.jpg";

pictureBox1.Image = Image.FromFile(resim);

}

else

{

resim += dtYemekler.Rows[siraNo]["resim"].ToString();

pictureBox1.Image = Image.FromFile(resim);

}

MySqlDataAdapter adtr = new MySqlDataAdapter("select \* from yemekler", bag.Baglan());//Komut ve bağlantı yüklenen DataAdapter nesnesi

adtr.Fill(dtMenu);

bag.BaglantiyiKapat();

}

}

Yemekler tablosunda seçim yapılan menü’nün nitelikleri’nin gösterilmesi için kullandığım VerileriDoldur metotudur.

private void dataGridView1\_SelectionChanged(object sender, EventArgs e)

{

if (dgw1.CurrentRow != null)

{

siraNo = dgw1.CurrentRow.Index;

dgw1.Rows[siraNo].Selected = true;

if (siraNo >= 0)

{

VerileriDoldur();

}

}

}

Seçilen satırın tamamının seçilmesi için kullanılan kod bu şekildedir.

private void dataGridView1\_SelectionChanged(object sender, EventArgs e)

{

if (dgw1.CurrentRow != null)

{

siraNo = dgw1.CurrentRow.Index;

dgw1.Rows[siraNo].Selected = true;

if (siraNo >= 0)

{

VerileriDoldur();

}

}

}

2.Formumuz Menü Düzenle’yi göstermek için kullandığım kod aşağıda verilmiştir.

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

{

MenuDüzenle menudüzenle = new MenuDüzenle();

menudüzenle.ShowDialog();

DataGridiDoldur();

}

---------------------------------------------------------------------------------------------------------------------------------------

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

{

yemekler.Add(txtYemek.Text);

fiyatlar.Add(txtFiyat.Text);

Menulistele();

index = fiyatlar.Count;

hesap += Convert.ToDecimal(fiyatlar[index - 1]);

lblHesap.Text = hesap.ToString();

}

Sipariş Ver formundaki menü ekle butonunun çalışması için gerekli kod bloğu yukarda verildiği gibidir. Burada yapmak istediğim şey seçilen menülerin toplu bir yerde gösterilmesi ve toplam fiyat bilgisini öğrenmektir.

private void Menulistele()

{

lbxMenu.Items.Clear();

for (int i = 0; i < yemekler.Count; i++)

{

lbxMenu.Items.Add(yemekler[i] + "=" + fiyatlar[i]);

}

lblSiparisSayisi.Text = lbxMenu.Items.Count.ToString();

}

Menü ekle butonunda kullandığım menülistele metotu yukarıda verilmiştir. Toplam sipariş sayısını öğrenmek amacıyla yapılmıştır.

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

{

int index = lbxMenu.SelectedIndex;

if (index < 0)

{

MessageBox.Show("Silinecek Menü'yü Seçiniz", "Uyarı!", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

hesap -= Convert.ToDecimal(fiyatlar[index]);

yemekler.RemoveAt(index);

fiyatlar.RemoveAt(index);

Menulistele();

lblHesap.Text = hesap.ToString();

}

}

Siparişte istenmeyen bir menü olduğunda silmek için kullandığım kodlar yukarıdaki şekildedir.

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

if (lbxMenu.Items.Count != 0)

{

string sql = "INSERT INTO `satislar`(`menuAdı=Fiyati`, `tarih`) VALUES (@menuAdıFiyati,@tarih)";

kmt = new MySqlCommand(sql, bag.Baglan());

foreach (var kategori in lbxMenu.Items)

{

kmt = new MySqlCommand(sql, bag.Baglan());

kmt.Parameters.AddWithValue("@menuAdıFiyati", kategori.ToString());

kmt.Parameters.AddWithValue("@tarih", dtpTarih.Value.ToString("yy-MM-dd"));

kmt.ExecuteNonQuery();

//siraNo = dtBirinci.Rows.Count;

DataGridiDoldur();

}

DialogResult seçenek = MessageBox.Show("Hesabı Tamamlamak İstiyor musunuz ?", "Uyarı!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (seçenek == DialogResult.Yes)

{

MessageBox.Show("Hesap : " + hesap.ToString() + " TL", "Hesap Tutarı");

//Görüntüyü Sıfırlamak için

lbxMenu.Items.Clear();

lblSiparisSayisi.Text = "";

lblHesap.Text = "";

//Kodda üstüne eklenmemesi için

yemekler.Clear();

fiyatlar.Clear();

hesap = 0;

}

else

{

return;

}

}

else

{

MessageBox.Show("Kayıt Edilmiş Menu Yok", "Uyarı!",MessageBoxButtons.OK,MessageBoxIcon.Warning);

}

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

bag.BaglantiyiKapat();

}

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Siparişler tamamlanmak istendiğinde kullanılan butondur.Burada yapmak istediğim şey tamamlanan siparişlerin Ciro Tablosuna eklenmesi ve daha sonra istenilen zamanda Ciro görüntüle butonuna tıklanarak hangi tarihte hangi satışların olduğunu öğrenmektir. Daha sonra tamamlanan siparişin toplam tutar bilgisini ekranda gösterilir ve sipariş tamamlanarak bir sonraki siparişler için temizlenerek hazır hale getirilir.

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

{

Ciro form3 = new Ciro();

form3.ShowDialog();

}

Yapılan siparişlerin tamamının gösterilmesi için olan Ciro görüntüle ekranının gelmesi için olan kod yukarıdadır.

Ve burada Sipariş Ver formu’nun kod kısmını tamamlamış oluyoruz.

2.Formumuz olan Menü Düzenle formunda yazacağımız kodlar için kullanılan kütüphaneler aşağıdaki gibidir.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using MySql.Data;

using MySql.Data.MySqlClient;

using System.IO;

Yazdığımız kodlarda kullanılan Global değişkenleri tanımlıyoruz.

DBislemleri bag = new DBislemleri();// Bağlantı için oluşturulan class’tan türetilen bir nesne

DataTable dtYemekler = new DataTable();//Tablolara yüklemek için bir adet dtBirinci değişkeni tanımlandı.

DataTable dtMenu = new DataTable();//Tablolara yüklemek için bir adet dtBirinci değişkeni tanımlandı.

MySqlCommand kmt = new MySqlCommand();//İşlemler için bir MySqlCommand nesnesi tanımlandı.

int siraNo = 0;

Kullanacağımız 3 metot’tan biri DataGridiDoldur metotudur.Burada yapılmak istenen “Yemekler” tablosun daki verilerin ekrana yansıtılmasıdır.

private void DataGridiDoldur()

{

dgw1.RowHeadersVisible = true;//Satır başlıkları görünmez yapılır.

dgw1.Font = new Font("Arial", 10);// Datagridview için font ayarlaması yapılır.

dtYemekler.Columns.Clear();//Tablonun önceki kullanımından kalan sütunlar temizlenir.

dtYemekler.Clear(); // Tablo temizlenir.

MySqlDataAdapter adtr = new MySqlDataAdapter("select \* from yemekler", bag.Baglan());//Komut ve bağlantı yüklenen DataAdapter nesnesi

adtr.Fill(dtYemekler);//DataAdapter nesnesine yüklenen bilgiler tabloya aktarılır.

dgw1.DataSource = dtYemekler; //Datagridview bileşeni data kaynağı olarak dtBirinci seçilir.

dgw1.Columns["id"].HeaderText = "ID";

dgw1.Columns["id"].Width = 75;

dgw1.Columns["ad"].HeaderText = "AD";

dgw1.Columns["ad"].Width = 220;

dgw1.Columns["fiyat"].HeaderText = "FİYAT";

dgw1.Columns["fiyat"].Width = 100;

dgw1.Columns["tur"].HeaderText = "TÜR";

dgw1.Columns["tur"].Width = 234;

dgw1.Columns["resim"].HeaderText = "RESİM";

dgw1.Columns["resim"].Width = 125;

try

{

dgw1.Rows[siraNo].Selected = true;

}

catch (Exception)

{

MessageBox.Show("Menü Boş", "Uyarı", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

VerileriDoldur();

bag.BaglantiyiKapat();

}

2.Metotumuz ComboboxDoldur metotudur.

Burada yapılmak istenen ise “Yemek Türü” tablosundan verilerin combobox’a getirilmesidir.

private void ComboboxDoldur()

{

dtMenu.Clear(); // Tablo temizlenir.

MySqlDataAdapter adtr2 = new MySqlDataAdapter("select \* from yemekturu", bag.Baglan());//Komut ve bağlantı yüklenen DataAdapter nesnesi

adtr2.Fill(dtMenu);

cbYemekTuru.Items.Clear();

for (int i = 0; i < dtMenu.Rows.Count; i++)

{

cbYemekTuru.Items.Add(dtMenu.Rows[i]["Tur"].ToString());

}

}

3.Metotumuz verileriDoldur metotudur.

Seçilen menü’nün bilgilerinin ekrana yansıtılmasıdır.

private void VerileriDoldur()

{

if (siraNo >= 0 && siraNo < dtYemekler.Rows.Count)

{

txtYemekAdı.Text = dtYemekler.Rows[siraNo]["ad"].ToString();

nYemekFiyatı.Text = dtYemekler.Rows[siraNo]["fiyat"].ToString();

txtid.Text = dtYemekler.Rows[siraNo]["id"].ToString();

cbYemekTuru.Text = dtYemekler.Rows[siraNo]["tur"].ToString();

string resim = Directory.GetParent(Environment.CurrentDirectory).Parent.FullName + "\\Resimler\\";

if (dtYemekler.Rows[siraNo]["resim"].ToString() == "")

{

resim += "no-image.jpg";

pictureBox1.Image = Image.FromFile(resim);

}

else

{

resim += dtYemekler.Rows[siraNo]["resim"].ToString();

pictureBox1.Image = Image.FromFile(resim);

}

}

}

2.Formumuz olan Menü Düzenle açılırken yapıcaklarımız bu 3 metotu çağırmak ve ekrandaki butonlar üzerine gelindiği zaman ne işe yaradığına dair küçük bir bilgilendirme notu bırakmaktır.

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

DataGridiDoldur();

ComboboxDoldur();

ToolTip btn\_aciklama = new ToolTip();

btn\_aciklama.SetToolTip(btnSil, "Menu Sil");

btn\_aciklama.SetToolTip(btnEkle, "Menu Ekle");

btn\_aciklama.SetToolTip(btnGuncelle, "Menu Güncelle");

btn\_aciklama.SetToolTip(btnResimGuncelle, "Resimi Güncelle");

btn\_aciklama.SetToolTip(btnResimSil, "Resimi Sil");

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Menü üzerinde değişiklikler yapmak üzere kullandığım ;

Menu Ekle butonu

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

string sorgu = "INSERT INTO `yemekler` ( `ad`, `tur`, `fiyat`) VALUES ( @ad, @tur, @fiyat)";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@ad", txtYemekAdı.Text.Trim());

kmt.Parameters.AddWithValue("@tur", cbYemekTuru.Text.Trim());

kmt.Parameters.AddWithValue("@fiyat", nYemekFiyatı.Text.Trim().ToString());

kmt.ExecuteNonQuery();

siraNo = dtYemekler.Rows.Count;

DataGridiDoldur();

MessageBox.Show("Kayıt Başarılı", "Kayıt İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Menü Güncelle butonu

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

string sorgu = "UPDATE `yemekler` SET `ad`=@ad,`tur`=@tur,`fiyat`=@fiyat WHERE `yemekler`.`id`=@id";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@ad", txtYemekAdı.Text.Trim());

kmt.Parameters.AddWithValue("@tur", cbYemekTuru.Text.Trim());

kmt.Parameters.AddWithValue("@fiyat", nYemekFiyatı.Text.Trim().ToString());

kmt.Parameters.AddWithValue("@id", txtid.Text.Trim().ToString());

kmt.ExecuteNonQuery();

siraNo = dgw1.CurrentRow.Index;

DataGridiDoldur();

MessageBox.Show("Güncelleme Başarılı", "Güncelleme İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Ve Menü Sil butonları fotoğraflardaki gibidir.

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

string sorgu = "DELETE FROM `yemekler` WHERE `yemekler`.`id` = @id; ";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@id", txtid.Text.Trim().ToString());

DialogResult cevap = MessageBox.Show("Kayıt Silinecek !", "Uyarı", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (cevap == DialogResult.OK)

{

kmt.ExecuteNonQuery();

siraNo = 0;

DataGridiDoldur();

}

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Seçilen menü’nün nitelikleri’nin ekrana yansıtılması için gerekli kod aşağıdaki fotoğrafta verilmiştir.

private void dataGridView1\_SelectionChanged(object sender, EventArgs e)

{

if (dgw1.CurrentRow != null)

{

siraNo = dgw1.CurrentRow.Index;

dgw1.Rows[siraNo].Selected = true;

if (siraNo >= 0)

{

VerileriDoldur();

}

}

}

Menü’de bulunan yemeğe resim eklemek istediğimiz de proje dosyamızdaki Resimler klasörü oluşturuyoruz. Ve ardından gerekli kodları yazıyoruz.

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

OpenFileDialog op = new OpenFileDialog();

string yol = Directory.GetParent(Environment.CurrentDirectory).Parent.FullName + "\\Resimler\\";

op.Title = "Resim Seç"; // filedialog penceresinin başlığında gösterilecek yazı atanır

op.Filter = "Dosya Türleri|\*.jpg;\*.jpeg;\*.png|" +

"JPEG (\*.jpg;\*.jpeg)|\*.jpg;\*.jpeg|" +

"Portable Network Graphic (\*.png)|\*.png";

DialogResult? kontrol = op.ShowDialog();

if (kontrol != null && kontrol == DialogResult.OK)

{

pictureBox1.Image = Image.FromFile(op.FileName);

if (!Directory.Exists(yol))

{

Directory.CreateDirectory(yol);

}

int rSayi = new Random().Next(10000);

string resimYolu = yol + rSayi + Path.GetFileName(op.FileName);

File.Copy(op.FileName, resimYolu, true);

string[] diziResim = resimYolu.Split('\\');

//MessageBox.Show(diziResim[diziResim.Length - 1].ToString());

try

{

string sorgu = "UPDATE `yemekler` SET `resim` = @resim WHERE `yemekler`.`id` = @id; ";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@id", txtid.Text.Trim().ToString());

kmt.Parameters.AddWithValue("@resim", diziResim[diziResim.Length - 1].ToString());

kmt.ExecuteNonQuery();

DataGridiDoldur();

MessageBox.Show("Resim Güncelleme Başarılı", "Resim Güncelleme İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

}

else

{

MessageBox.Show("Resim seçilmedi!", "Uyarı", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Dilediğimizde bu resimleri silmemiz için Resmi Sil butonuna gerekli kodları aşağıdaki gibi yazıyoruz.

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

{

dgw1.SelectionChanged -= dataGridView1\_SelectionChanged;

if (dtYemekler.Rows[siraNo]["resim"].ToString().Trim() != "")

{

try

{

string sorgu = "UPDATE `yemekler` SET `resim` = @resim WHERE `yemekler`.`id` = @id; ";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@id", txtid.Text.Trim().ToString());

kmt.Parameters.AddWithValue("@resim", "");

kmt.ExecuteNonQuery();

string resim = Directory.GetParent(Environment.CurrentDirectory).Parent.FullName + "\\Resimler\\" + dtYemekler.Rows[siraNo]["resim"].ToString();

if (File.Exists(resim))

{

string resimNoImage = Directory.GetParent(Environment.CurrentDirectory).Parent.FullName + "\\Resimler\\no-image.jpg";

pictureBox1.Image = Image.FromFile(resimNoImage);

GC.Collect();

GC.WaitForPendingFinalizers();

File.Delete(resim);

}

MessageBox.Show("Resim Silme Başarılı", "Resim Silme İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

DataGridiDoldur();

}

}

else

{

MessageBox.Show("Henüz bir resim tanımlanmadı !", "Uyarı!", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

dgw1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Yemek türü ekleyeceğimiz zaman 3.Formumuz olan YemekTürüEkle’yi açmak için butonumuza kodları yazıyoruz.

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

{

YemekTuru form3 = new YemekTuru();

form3.ShowDialog();

ComboboxDoldur();

}

Ve bu son kod ile Menü Düzenle formumuzun kodlarını bitiriyoruz.

3.Formumuz Yemek Türü Ekle için gerekli kütüphaneleri çağırıyoruz.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using MySql.Data.MySqlClient;

Global değişkenlerimizi tanımlıyoruz.

DBislemleri bag = new DBislemleri();

DataTable dTur = new DataTable();

MySqlCommand kmt = new MySqlCommand();

int siraNo = 0;

Formumuz açıldığı zaman yapıcağımız işlemleri Form\_Load kısmına yazıyoruz.

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

{

DataGridiDoldur();

ToolTip btn\_aciklama = new ToolTip();

btn\_aciklama.SetToolTip(btnSil, "Menu Sil");

btn\_aciklama.SetToolTip(btnEkle, "Menu Ekle");

btn\_aciklama.SetToolTip(btnGuncelle, "Menu Güncelle");

}

Bu kısımda yapmak istediklerim; ”YemekTürü” tablosu’nun verilerini ekranda göstermek ve DataGridiDoldur metotu’nun içindeki VerileriDoldur metotu ile seçilen Türün niteliklerinin ekranda gösterilmesidir. Kullanıcı butonların üzerine geldiğinde ne işe yaradıklarına dair küçük birer not bırakmaktır.

Kullandığım DataGridiDoldur metotu aşağıdaki fotoğrafta gösterilmiştir.

private void DataGridiDoldur()

{

dataGridView1.SelectionChanged -= dataGridView1\_SelectionChanged;

dataGridView1.RowHeadersVisible = true;

dataGridView1.Font = new Font("Arial", 10);

dTur.Columns.Clear();

dTur.Clear();

MySqlDataAdapter adtr = new MySqlDataAdapter("select \* from yemekturu", bag.Baglan());

adtr.Fill(dTur);

dataGridView1.DataSource = dTur;

dataGridView1.Columns["id"].HeaderText = "ID";

dataGridView1.Columns["id"].Width = 80;

dataGridView1.Columns["Tur"].HeaderText = "TÜR";

dataGridView1.Columns["tur"].Width = 165;

try

{

dataGridView1.Rows[siraNo].Selected = true;

}

catch (Exception)

{

MessageBox.Show("Yemek Türü Tablosu Boş","Uyarı",MessageBoxButtons.OK,MessageBoxIcon.Warning);

}

VerileriDoldur();

bag.BaglantiyiKapat();

dataGridView1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Ve VerileriDoldur metotu da aşağıda verilmiştir.

private void VerileriDoldur()

{

if (siraNo >= 0 && siraNo < dTur.Rows.Count)

{

txtID.Text = dTur.Rows[siraNo]["id"].ToString();

txtYemekTuru.Text = dTur.Rows[siraNo]["Tur"].ToString();

}

}

Seçilelen menü’nün nitelikleri’nin ekrana yansıtılması için kullanılan kod aşağıda verilmiştir.

private void dataGridView1\_SelectionChanged(object sender, EventArgs e)

{

if (dataGridView1.CurrentRow != null)

{

siraNo = dataGridView1.CurrentRow.Index;

dataGridView1.Rows[siraNo].Selected = true;

if (siraNo >= 0)

{

VerileriDoldur();

}

}

}

Yemek türünü eklemek için kullandığım kod bloğu bu şekildedir.

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

{

dataGridView1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

string sorgu = "INSERT INTO `yemekturu` ( `Tur` ) VALUES ( @Tur ); ";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@Tur", txtYemekTuru.Text.Trim());

kmt.ExecuteNonQuery();

siraNo = dTur.Rows.Count;

DataGridiDoldur();

MessageBox.Show("Kayıt Başarılı", "Kayıt İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

dataGridView1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Aynı şekilde Yemek türünü güncellemek için:

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

{

dataGridView1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

string sorgu = "UPDATE `yemekturu` SET `Tur` = @Tur WHERE `id` = @id; ";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@Tur", txtYemekTuru.Text.Trim());

kmt.Parameters.AddWithValue("@id", txtID.Text.Trim().ToString());

kmt.ExecuteNonQuery();

siraNo = dataGridView1.CurrentRow.Index;

DataGridiDoldur();

MessageBox.Show("Güncelleme Başarılı", "Güncelleme İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

dataGridView1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Ve Dilediğimiz zaman türün kaldırılması için kullandığım Menü Sil kodlarıdır.

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

{

dataGridView1.SelectionChanged -= dataGridView1\_SelectionChanged;

try

{

string sorgu = "DELETE FROM `yemekturu` WHERE `id` = @id; ";

kmt = new MySqlCommand(sorgu, bag.Baglan());

kmt.Parameters.AddWithValue("@id", txtID.Text.Trim().ToString());

kmt.ExecuteNonQuery();

siraNo = 0;

DataGridiDoldur();

MessageBox.Show("Silme Başarılı", "Silme İşlemi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception hata)

{

MessageBox.Show("HATA : " + hata.Message);

}

finally

{

kmt.Connection.Close();

bag.BaglantiyiKapat();

}

dataGridView1.SelectionChanged += dataGridView1\_SelectionChanged;

}

Burada 3.Formumuz “YemekTürü”nün kodlarınız bitirmiş oluyoruz.

4. ve Son formumuz olan Ciro ekranı için kütüphaneleri ekleyerek başlıyoruz.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using MySql.Data.MySqlClient;

Global Değişkenlerimizi tanımlıyoruz.

DBislemleri bag = new DBislemleri();// Bağlantı için oluşturulan class’tan türetilen bir nesne

DataTable dtCiro = new DataTable();//Tablolara yüklemek için bir adet dtBirinci değişkeni tanımlandı.

int siraNo = 0;

Bu formda yapmak istediklerimiz gayet basit ve sadedir. Ciro tablosunu görüntülemek. Bu amaç için DataGridiDoldur metotu oluşturacağız ve bunu form açılırken göstereceğiz yani Form\_Load kısmına metotumuzu ekleyeceğiz.

private void DataGridiDoldur()

{

dataGridView1.RowHeadersVisible = true;//Satır başlıkları görünmez yapılır.

dataGridView1.Font = new Font("Arial", 10);// Datagridview için font ayarlaması yapılır.

dtCiro.Columns.Clear();//Tablonun önceki kullanımından kalan sütunlar temizlenir.

dtCiro.Clear(); // Tablo temizlenir.

MySqlDataAdapter adtr = new MySqlDataAdapter("select \* from satislar", bag.Baglan());//Komut ve bağlantı yüklenen DataAdapter nesnesi

adtr.Fill(dtCiro);//DataAdapter nesnesine yüklenen bilgiler tabloya aktarılır.

dataGridView1.DataSource = dtCiro; //Datagridview bileşeni data kaynağı olarak dtBirinci seçilir.

dataGridView1.Columns.Remove("id");

dataGridView1.Columns["menuAdı=Fiyati"].HeaderText = "Menü Adı ve Fiyatı";

dataGridView1.Columns["menuAdı=Fiyati"].Width = 365;

dataGridView1.Columns["tarih"].HeaderText = "Tarih";

dataGridView1.Columns["tarih"].Width = 375;

try

{

dataGridView1.Rows[siraNo].Selected = true;

}

catch (Exception)

{

MessageBox.Show(" Tablo Boş","Uyarı",MessageBoxButtons.OK,MessageBoxIcon.Warning);

}

bag.BaglantiyiKapat();

}

Kullandığım DataGridiDoldur metotudur.

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

{

DataGridiDoldur();

}

Ve bu metotu formumuz yüklenirken göstermek için yazdığım kod yukarıda verilmiştir.

4.Formumuz burada tamamlamış bulunuyoruz.

Projemiz burada sona ermiştir.

İncelediğiniz için Teşekkür ederim…