



**T.C.**  
**BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Hastane Yönetim ve Randevu Otomasyon Projesi**

**Eyüp Furkan ARIKAN**

**PROJE I ÇALIŞMASI**

**PROJE I DANIŞMANI : Dr.Öğr.Üyesi Salim CEYHAN**

**BİLECİK**  
**21 Ocak 2019**



**T.C.**  
**BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Ruby Programlama**

**Eyüp Furkan ARIKAN**

**PROJE I ÇALIŞMASI**

**PROJE I DANIŞMANI : Dr.Öğr.Üyesi Salim CEYHAN**

**BİLECİK**

**21 Ocak 2019**

---

---

## ÖZET

### Projenin Amacı

.....

.....

### Projenin Kapsamı

.....

.....

### Sonuçlar

.....

.....

**ABSTRACT**

**Project Objective**

.....  
.....

**Scope of Project**

.....  
.....

**Results**

.....  
.....

---

---

## TEŞEKKÜR

Bu projenin başından sonuna kadar hazırlanmasında emeği bulunan ve beni bu konuya yönlendiren saygıdeğer hocam ve danışmanım Sayın Doktor Öğretim Üyesi Salim CEYHAN'a tüm katkılarından ve hiç eksiltmediği desteğinden dolayı teşekkür ederim.

**Eyüp Furkan ARIKAN**

21 Ocak 2019

---

---

# İÇİNDEKİLER

ÖZET	ii
ABSTRACT	iii
TEŞEKKÜR	iv
SİMGE LİSTESİ	vi
ŞEKİL LİSTESİ	vii
TABLO LİSTESİ	viii
1 GİRİŞ	1
2 PROJE KOD ANLATIMI ve VERİ TABANI	2
3 PROGRAM ÇALIŞMASININ ANLATIMI ve VERİ TABANI	68
4 SONUÇLAR VE ÖNERİLER	69
5 EKLER	70
KAYNAKLAR	71
ÖZGEÇMİŞ	72

---

---

## SİMGE LİSTESİ

---

---

## ŞEKİL LİSTESİ



---

---

## TABLO LİSTESİ

---

---

# 1 GİRİŞ

---

---

## 2 PROJE KOD ANLATIMI ve VERİ TABANI

**Dosya 1:** Sql Bağlatısı Oluşturma:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data.SqlClient;

namespace Yonetim_Hastane
{
    class SqlBaglantisi
    {
        public SqlConnection baglanti()
        {
            SqlConnection baglan = new SqlConnection("Data
            Source=FURKAN\\SQLEXPRESS;Initial_Catalog=
            Hastane_Yonetim_ve_Randevu_Sistemi;Integrated
            Security=True");
            baglan.Open();
            return baglan;
        }
    }
}
```

Bu dosyadaki komutla sql bağlantısını oluşturup açıyorum ve başka dosyalarda kullanılmak üzere geri döndürüyorum.

**Dosya 2:** İlk Giriş Formu:

```
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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmGirisler : Form
    {
        public FrmGirisler()
        {
            InitializeComponent();

            SqlBaglantisi bgl = new SqlBaglantisi();

            private void button1_Click(object sender, EventArgs e)
            {
                String secim = comboBox1.Text;

                if( secim.Equals("Hasta") )
                {
                    SqlCommand komut = new SqlCommand("select_*_from_Tb
                    komut.Parameters.AddWithValue("@p1", textBox1.Text)
                    komut.Parameters.AddWithValue("@p2", textBox2.Text)
                    SqlDataReader dr = komut.ExecuteReader();
```

---

---

```

        if ( dr . Read () )
        {
            FrmHastaDetay fr = new FrmHastaDetay ();
            fr . tc = textBox1 . Text ;
            fr . Show () ;
            this . Hide () ;
        }
        else
        {
            MessageBox . Show ( "TC_Kimlik_No_ya_da_şifre_hatalı" ) ;
        }
        bg1 . baglanti () . Close () ;
    } else if ( secim . Equals ( "Doktor" ) )
    {
        SqlCommand komut = new SqlCommand ( "select_*_from_Tb"
        komut . Parameters . AddWithValue ( "@d1" , textBox1 . Text )
        komut . Parameters . AddWithValue ( "@d2" , textBox2 . Text )
        SqlDataReader dr = komut . ExecuteReader () ;
        if ( dr . Read () )
        {
            FrmDoktorDetay fr = new FrmDoktorDetay () ;
            fr . DoktorTCsi = textBox1 . Text ;
            fr . Show () ;
            this . Hide () ;
        }
        else
        {
            MessageBox . Show ( "TC_Kimlik_No_ya_da_şifre_hatalı" ) ;
        }
        bg1 . baglanti () . Close () ;
    } else if ( secim . Equals ( "Sekreter" ) )

```

---

---

```

{
    SqlCommand komut = new SqlCommand("select_*_from_Tb
    komut.Parameters.AddWithValue("@p1", textBox1.Text)
    komut.Parameters.AddWithValue("@p2", textBox2.Text)
    SqlDataReader dr = komut.ExecuteReader();
    if (dr.Read())
    {
        FrmSekreterDetay frs = new FrmSekreterDetay();
        frs.SekreterTC = textBox1.Text;
        frs.Show();
        this.Hide();
    }
    else
    {
        MessageBox.Show("TC_Kimlik_No_ya_da_şifre_hatalı");
    }
    bgl.baglanti().Close();
} else if( secim.Equals("Yönetici") )
{
    SqlCommand komut = new SqlCommand("select_*_from_Tb
    komut.Parameters.AddWithValue("@p1", textBox1.Text)
    komut.Parameters.AddWithValue("@p2", textBox2.Text)
    SqlDataReader dr = komut.ExecuteReader();
    if (dr.Read())
    {
        FrmYoneticiDetay frs = new FrmYoneticiDetay();
        frs.YoneticiTC = textBox1.Text;
        frs.Show();
        this.Hide();
    }
    else

```

---



---

```

        {
            MessageBox.Show("TC_Kimlik_No_ya_da_şifre_hatalı");
        }
        bgl.baglanti().Close();
    } else if( secim.Equals("Müşteri_Hizmetleri") )
    {
        SqlCommand komut = new SqlCommand("select_*_from_Tb
        komut.Parameters.AddWithValue("@p1", textBox1.Text);
        komut.Parameters.AddWithValue("@p2", textBox2.Text);
        SqlDataReader dr = komut.ExecuteReader();
        if (dr.Read())
        {
            FrmMusteriHizmetleri frs = new FrmMusteriHizmetleri();
            frs.MusteriHizmetleriTC = textBox1.Text;
            frs.Show();
            this.Hide();
        }
        else
        {
            MessageBox.Show("TC_Kimlik_No_ya_da_şifre_hatalı");
        }
        bgl.baglanti().Close();
    }
}

private void button2_Click(object sender, EventArgs e)
{
    FrmHastaKayit fr = new FrmHastaKayit();
    fr.Show();
}
}

```

---

---

```
}
```

Yukarıdaki Giriş Formu Kodları'nda;

```
SqlBaglantis i bgl = new SqlBaglantis i ();
```

komutuyla Sql bağlantısını çağırdık.

button1\_Click'in içinde;

```
String secim = comboBox1.Text;
```

komutuyla comboBox1'in yazısını string seçim değişkenine aktardım.

```
if ( secim.Equals("Hasta") )
```

```
{
```

```
SqlCommand komut = new SqlCommand("select_*_from_Tb
```

```
komut.Parameters.AddWithValue("@p1", textBox1.Text)
```

```
komut.Parameters.AddWithValue("@p2", textBox2.Text)
```

```
SqlDataReader dr = komut.ExecuteReader();
```

```
if (dr.Read())
```

```
{
```

```
FrmHastaDetay fr = new FrmHastaDetay();
```

```
fr.tc = textBox1.Text;
```

```
fr.Show();
```

```
this.Hide();
```

```
}
```

```
else
```

```
{
```

```
MessageBox.Show("TC_Kimlik_No_ya_da_sifre_hatal
```

```
}
```

```
bgl.baglanti().Close();
```

```
}
```

Seçim eğer Hasta ise Tbl\_Hastalar tablosundan HastaTC ve HastaSifre'ye göre verileri



bağlantıyı sağlayarak getir dedim. p1 ve p2 parametrelerini HastaTC ve HastaSifre için kullandım. Sonra Sql veri okuyucuyu komuta göre çalıştırdım.

Eğer veri okunduysa önce Tc'yi açılan forma gönderdim. FrmHastaDetay formunu aç dedim. Bu formu gösterme dedim. Eğer okunmadıysa "TC Kimlik No ya da şifre hatalıdır." uyarı mesajını gösterdim ve bağlantıyı kapattım.

```
else if ( secim.Equals("Doktor") )
{
    SqlCommand komut = new SqlCommand("select_*_from_Tb
komut.Parameters.AddWithValue("@d1", textBox1.Text)
komut.Parameters.AddWithValue("@d2", textBox2.Text)
SqlDataReader dr = komut.ExecuteReader();
if ( dr.Read() )
{
    FrmDoktorDetay fr = new FrmDoktorDetay();
    fr.DoktorTCsi = textBox1.Text;
    fr.Show();
    this.Hide();
}
else
{
    MessageBox.Show("TC_Kimlik_No_ya_da_şifre_hatalı
}
bgl.baglanti().Close();
}
```

Seçim eğer Doktor ise Tbl\_Doktorlar tablosundan DoktorTC ve DoktorSifre'ye göre verileri bağlantıyı sağlayarak getir dedim. d1 ve d2 parametrelerini DoktorTC ve DoktorSifre için kullandım. Sonra Sql veri okuyucuyu komuta göre çalıştırdım.

Eğer veri okunduysa önce DoktorTCsi'ni açılan forma gönderdim. FrmDoktorDetay formunu aç dedim. Bu formu gösterme dedim. Eğer okunmadıysa "TC Kimlik No ya da şifre

---

---

hatalıdır." uyarı mesajını gösterdim ve bağlantıyı kapattım.

```
else if ( secim.Equals(" Sekreter" ) )
{
    SqlCommand komut = new SqlCommand(" select_*_from_Tb
    komut.Parameters.AddWithValue("@p1", textBox1.Text)
    komut.Parameters.AddWithValue("@p2", textBox2.Text)
    SqlDataReader dr = komut.ExecuteReader();
    if ( dr.Read() )
    {
        FrmSekreterDetay frs = new FrmSekreterDetay();
        frs.SekreterTC = textBox1.Text;
        frs.Show();
        this.Hide();
    }
    else
    {
        MessageBox.Show("TC_Kimlik_No_ya_da_şifre_hatalıdır.");
    }
    bgl.baglanti().Close();
}
```

Seçim eğer Sekreter ise Tbl\_Sekreterler tablosundan SekreterTC ve SekreterSifre'ye göre verileri bağlantıyı sağlayarak getir dedim. p1 ve p2 parametrelerini SekreterTC ve SekreterSifre için kullandım. Sonra Sql veri okuyucuyu komuta göre çalıştırdım.

Eğer veri okunduysa önce SekreterTC'yi açılan forma gönderdim. FrmSekreterDetay formunu aç dedim. Bu formu gösterme dedim. Eğer okunmadıysa "TC Kimlik No ya da şifre hatalıdır." uyarı mesajını gösterdim ve bağlantıyı kapattım.

```
else if ( secim.Equals(" Yönetici" ) )
{
    SqlCommand komut = new SqlCommand(" select_*_from_Tb
```

---

---

```

        komut.Parameters.AddWithValue("@p1", textBox1.Text)
        komut.Parameters.AddWithValue("@p2", textBox2.Text)
        SqlDataReader dr = komut.ExecuteReader();
        if (dr.Read())
        {
            FrmYoneticiciDetay frs = new FrmYoneticiciDetay();
            frs.YoneticiciTC = textBox1.Text;
            frs.Show();
            this.Hide();
        }
        else
        {
            MessageBox.Show("TC_Kimlik_No_ya_da_sifre_hatalıdır.");
        }
        bg1.baglanti().Close();
    }

```

Seçim eğer Yönetici ise Tbl\_Yoneticiler tablosundan YoneticiciTC ve YoneticiciSifre'ye göre verileri bağlantıyı sağlayarak getir dedim. p1 ve p2 parametrelerini YoneticiciTC ve YoneticiciSifre için kullandım. Sonra Sql veri okuyucuyu komuta göre çalıştırdım.

Eğer veri okunduysa önce YoneticiciTC'yi açılan forma gönderdim. FrmYoneticiciDetay formunu aç dedim. Bu formu gösterme dedim. Eğer okunmadıysa "TC Kimlik No ya da şifre hatalıdır." uyarı mesajını gösterdim ve bağlantıyı kapattım.

```

else if( secim.Equals("Müşteri_Hizmetleri") )
{
    SqlCommand komut = new SqlCommand("select * from Tbl_Müşteri_Hizmetleri");
    komut.Parameters.AddWithValue("@p1", textBox1.Text)
    komut.Parameters.AddWithValue("@p2", textBox2.Text)
    SqlDataReader dr = komut.ExecuteReader();
    if (dr.Read())

```

```

        {
            FrmMusteriHizmetleri frs = new FrmMusteriHizmetleri();
            frs.MusteriHizmetleriTC = textBox1.Text;
            frs.Show();
            this.Hide();
        }
        else
        {
            MessageBox.Show("TC_Kimlik_No_ya_da_sifre_hatalıdır.");
        }
        bgl.baglanti().Close();
    }

```

Seçim eğer Müşteri Hizmetleri ise Tbl\_Musteri\_Hizmetleri tablosundan MusteriHizmetleriTC ve MusteriHizmetleriSifre'ye göre verileri bağlantıyı sağlayarak getir dedim. p1 ve p2 parametrelerini MusteriHizmetleriTC ve MusteriHizmetleriSifre için kullandım. Sonra Sql veri okuyucuyu komuta göre çalıştırdım.

Eğer veri okunduysa önce MusteriHizmetleriTC'yi açılan forma gönderdim. FrmMusteriHizmetleri formunu aç dedim. Bu formu gösterme dedim. Eğer okunmadıysa "TC Kimlik No ya da şifre hatalıdır." uyarı mesajını gösterdim ve bağlantıyı kapattım.

button2\_Click içinde;

```

    FrmHastaKayit fr = new FrmHastaKayit();
    fr.Show();

```

komutlarıyla FrmHastaKayit formunu açtım.

**Dosya 3:** Hasta Kayıt Formu:

```

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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmHastaKayit : Form
    {
        public FrmHastaKayit()
        {
            InitializeComponent();

            SqlBaglantisi bgl = new SqlBaglantisi();

            private void BtnKayitYap_Click(object sender, EventArgs e)
            {
                if ( TCKontrol(MskTC.Text) ) {
                    SqlCommand komut = new SqlCommand("insert_into_Tbl_
                    komut.Parameters.AddWithValue("@p1", TxtAd.Text);
                    komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);
                    komut.Parameters.AddWithValue("@p3", MskTC.Text);
                    komut.Parameters.AddWithValue("@p4", MskTelefon.Text);
                    komut.Parameters.AddWithValue("@p5", TxtSifre.Text);
                    komut.Parameters.AddWithValue("@p6", CmbCinsiyet.Text);
                    komut.ExecuteNonQuery();
                    bgl.baglanti().Close();
                }
            }
        }
    }
}

```

---

---

```
        MessageBox.Show(" Kaydınız_gerçekleşmiştir._Şifreniz  
    }  
}  
  
public static Boolean TCKontrol(String TC)  
{  
    string tcNo = TC;  
    int toplam = 0; int toplam2 = 0; int toplam3 = 0;  
    if (tcNo.Length == 11)  
    {  
        if (Convert.ToInt32(tcNo[0].ToString()) != 0) // tc  
        {  
            for (int i = 0; i < 10; i++)  
            {  
                toplam = toplam + Convert.ToInt32(tcNo[i].T  
                if (i % 2 == 0)  
                {  
                    if (i != 10)  
                    {  
                        toplam2 = toplam2 + Convert.ToInt32  
                    }  
                }  
                else  
                {  
                    if (i != 9)  
                    {  
                        toplam3 = toplam3 + Convert.ToInt32  
                    }  
                }  
            }  
        }  
    }  
}
```

```

        }
        else
        {
            MessageBox.Show("Tc_Kimlik_Numaranızın_ilk_hane", "Tc_Kimlik_Numaranızın_ilk_hane", MessageBoxButtons.OK);
            return false;
        }
    }
    else
    {
        MessageBox.Show("Tc_Kimlik_Numarınız_11_haneli_olmalı", "Tc_Kimlik_Numarınız_11_haneli_olmalı", MessageBoxButtons.OK);
        return false;
    }

    if (((toplamlar2 * 7) - toplam3) % 10 == Convert.ToInt32(toplam3))
    {
        return true;
    }
    else
    {
        MessageBox.Show("Tc_Kimlik_Numarası_Yanlış!");
        return false;
    }
}

}

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

SqlBaglantisi bgl = new SqlBaglantisi();

```

---

---

komutuyla Sql bağlantısını yaptım.

TCKontrol fonksiyonu içinde;

```
public static Boolean TCKontrol(String TC)
{
    string tcNo = TC;
    int toplam = 0; int toplam2 = 0; int toplam3 = 0;
    if (tcNo.Length == 11)
    {
        if (Convert.ToInt32(tcNo[0].ToString()) != 0) // tc
        {
            for (int i = 0; i < 10; i++)
            {
                toplam = toplam + Convert.ToInt32(tcNo[i].T
                if (i % 2 == 0)
                {
                    if (i != 10)
                    {
                        toplam2 = toplam2 + Convert.ToInt32
                    }
                }
                else
                {
                    if (i != 9)
                    {
                        toplam3 = toplam3 + Convert.ToInt32
                    }
                }
            }
        }
    }
```



```

        else
        {
            MessageBox.Show("Tc_Kimlik_Numaranızın_ilk_hane
            return false;
        }
    }
    else
    {
        MessageBox.Show("Tc_Kimlik_Numarınız_11_haneli_olma
        return false;

    }
    if (((toplamlar2 * 7) - toplam3) % 10 == Convert.ToInt32(tc
    {
        return true;
    }
    else
    {
        MessageBox.Show("Tc_Kimlik_Numarası_Yanlış!");
        return false;
    }
}

```

komutlarıyla şunları yaptım:

Length komutuyla uzunluğun 11 olup olmadığını kontrol ettim. tc kimlik numaranın ilk hanesi 0 değilse dedim ve tc kimlik numaranın ilk hanesinin 0 olup olmadığını kontrol ettim. İndis değişkeni i'yi 0'dan 10'a kadar artırdım. TC formülüne göre yaptığım sistemde tcNo'nun ilk 10 elemanını tamsayıya çevirip topladım ve toplam değişkenine aktardım. tcNo'nun 1,3,5,7 ve 9. elemanlarını 0'dan başlattığımız i 10 olmamak koşuluyla tamsayıya çevirip topladım ve toplam2 değişkenine aktardım. Burada 7 ile çarpılacak sayıları topluyoruz. 2,4,6 ve 8. elemanlarını 0'dan başlattığımız i 10 olmamak koşuluyla tamsa-

yıya çevirip topladım ve toplam3 değişkenine aktardım.

tc kimlik numaranın ilk hanesi 0 değilse koşulunda değilse deyip eğer öyleyse yani tcnin ilk hanesi 0'sa "Tc Kimlik Numaranızın ilk hanesi 0 olamaz." uyarı mesajı verdirdim. Uzunluğun 11 olup olmadığını kontrol ettiğim koşulda değilse deyip eğer tc uzunluğu 11 değilse "Tc Kimlik Numaranız 11 haneli olmak zorunda.Eksik ya da fazla değer girdiniz." uyarı mesajı verdirdim.

Formüle göre toplam2'nin 7 katının toplam3'ten farkının mod 10'dan kalanı tc'nin 10. elemanıya eşitse ve toplam'ın mod 10'dan kalanı tc'nin 11. elemanına eşitse dedim ve doğru değerini döndür dedim. Değilse Tc "Kimlik Numarası Yanlış!" uyarı mesajını göster ve yanlış döndür dedim.

BtnKayitYap\_Click içinde;

```
if ( TCKontrol(MskTC.Text) ) {  
    SqlCommand komut = new SqlCommand("insert_into_Tbl_  
    komut.Parameters.AddWithValue("@p1", TxtAd.Text);  
    komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);  
    komut.Parameters.AddWithValue("@p3", MskTC.Text);  
    komut.Parameters.AddWithValue("@p4", MskTelefon.Text);  
    komut.Parameters.AddWithValue("@p5", TxtSifre.Text);  
    komut.Parameters.AddWithValue("@p6", CmbCinsiyet.Text);  
    komut.ExecuteNonQuery();  
    bgl.baglanti().Close();  
    MessageBox.Show("Kaydınız_gerçekleşmiştir._Şifreniz  
}
```

komutlarıyla tc kontrolü sağlandıktan sonra ad, soyad, tc, telefon, şifre ve cinsiyet parametreleriyle Hastalar tablosuna veri kaydı yaptım. Sorguyu çalıştır dedim ve bağlantıyı kapattım. Bilgi için "Kaydınız gerçekleşmiştir. Şifreniz: " diyerek şifresini de söyleyerek bilgi mesajı verdim.

**Dosya 4:** Hasta Detay Formu:

---

---

```

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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmHastaDetay : Form
    {
        public FrmHastaDetay()
        {
            InitializeComponent();

            public string tc;

            SqlBaglantisi bgl = new SqlBaglantisi();

            private void FrmHastaDetay_Load(object sender, EventArgs e)
            {
                LblTC.Text = tc;
                //Ad Soyad Çekme
                SqlCommand komut = new SqlCommand("select_HastaAd ,HastaSoyad", bgl.baglanti());
                komut.Parameters.AddWithValue("@p1", tc);
                SqlDataReader dr = komut.ExecuteReader();
            }
        }
    }
}

```

---

---

```

        while (dr.Read())
        {
            LblAdSoyad.Text = dr[0] + "_" + dr[1];
        }
        bgl.baglanti().Close();

        /*//Randevu Geçmişi
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_*_from_T",
        da.Fill(dt);
        dataGridView1.DataSource = dt; */

        //Branş Çekme

        SqlCommand komut2 = new SqlCommand("select_BransAd_from",
        SqlDataReader dr2 = komut2.ExecuteReader();
        while (dr2.Read())
        {
            CmbBrans.Items.Add(dr2[0]);
        }
        bgl.baglanti().Close();

        DataTable dt2 = new DataTable();
        SqlDataAdapter da2 = new SqlDataAdapter("SELECT_Tbl_Ran",
        da2.Fill(dt2);
        dataGridView1.DataSource = dt2;
    }

    private void CmbBrans_SelectedIndexChanged(object sender, EventArgs e)
    {
        CmbDoktor.Items.Clear();
    }

```

---



---

```

        SqlCommand komut3 = new SqlCommand("SELECT_DoktorAd , Dok
        komut3.Parameters.AddWithValue("@p1", CmbBrans.Text);
        SqlDataReader dr3 = komut3.ExecuteReader();
        while (dr3.Read())
        {
            CmbDoktor.Items.Add(dr3[0] + "_" + dr3[1]);
        }
        bgl.baglanti().Close();
    }

    private void CmbDoktor_SelectedIndexChanged(object sender,
    {
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_Randevuic
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

    private void LnkBilgiDuzenle_LinkClicked(object sender, Lin
    {
        FrmBilgiDuzenle fr = new FrmBilgiDuzenle();
        fr.TCno = LblTC.Text;
        fr.Show();
    }

    private void dataGridView2_CellClick(object sender, DataGri
    {
        int secilen = dataGridView2.SelectedCells[0].RowIndex;
        Txtid.Text = dataGridView2.Rows[secilen].Cells[0].Value
    }

```

---

---

```
public string RandevuTarihi , RandevuSaati;

private void dataGridView2_CellContentClick(object sender ,
{

}

private void button1_Click(object sender , EventArgs e)
{
    SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevula
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Randevu_iptal_edildi.", "Bilgi", Messa

    TabloGuncelle();

}

private void dataGridView1_CellClick(object sender , DataGri
{
    int secilen = dataGridView1.SelectedCells[0].RowIndex;
    Txtid.Text = dataGridView1.Rows[secilen].Cells[0].Value
}

private void TabloGuncelle()
{
    //Tablo Güncelleme
    DataTable dt2 = new DataTable();
    SqlDataAdapter da2 = new SqlDataAdapter("SELECT_Tbl_Ran
    da2.Fill(dt2);
    dataGridView1.DataSource = dt2;
```

---



---

```

        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_RandevuTarihi");
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }

    private void BtnRandevuAl_Click(object sender, EventArgs e)
    {
        SqlCommand komut = new SqlCommand("UPDATE_Tbl_RandevuTarihi");
        komut.Parameters.AddWithValue("@r2", RchSikayet.Text);
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_oluşturuldu.", "Bilgi", MessageBoxButtons.OK);

        TabloGuncelle();

        /*SqlCommand komut = new SqlCommand("insert_into_Tbl_RandevuTarihi");
        komut.Parameters.AddWithValue("@r2", RchSikayet.Text);
        komut.Parameters.AddWithValue("@r3", Txtid.Text);
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();

        // dataGridView2_CellClick özelliği çalışmadığı için RandevuAl_Click
        SqlCommand komut2 = new SqlCommand("select_RandevuTarihi");
        da2 = new SqlDataAdapter(komut2, bgl.baglanti());
        dt2 = new DataTable();
        da2.Fill(dt2);
        dataGridView2.DataSource = dt2;
    }

```

```

        komut2.Parameters.AddWithValue("@r4", Txtid.Text);
        SqlDataReader dr = komut2.ExecuteReader();
        RandevuTarihi = dr[1].ToString();
        RandevuSaati = dr[2].ToString();
        MessageBox.Show(CmbBrans.Text + "_bölümünden_" + CmbDol
        bgl.baglanti().Close();*/
        //Buraya kadar!
    }
}
}

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

SqlBaglantisi bgl = new SqlBaglantisi();

komutuyla Sql bağlantısını yaptım.

FrmHastaDetay_Load içinde;

LblTC.Text = tc;

        //Ad Soyad Çekme
        SqlCommand komut = new SqlCommand("select_HastaAd ,Hasta
        komut.Parameters.AddWithValue("@p1", tc);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())
        {
            LblAdSoyad.Text = dr[0] + "_" + dr[1];
        }
        bgl.baglanti().Close();

komutlarında

public string tc;

```



komutuyla aldığımız tc'yi etiket tc'nin yazısına yapıştırdım. Tc'yi gösterdim. Sonra ad soyad çekme yaptım. Hasta ad ve soyadı Hastalar tablosundan HastaTC'ye göre getirdim. dr isimli sql veri okuyucuyu çalıştırdım. Eğer okunduysa ad soyad etiketine arada boşluk bırakarak yazdırdım ve sql bağlantısını kapattım.

```
SqlCommand komut2 = new SqlCommand(" select _BransAd_from _Tbl_Bransla  
    SqlDataReader dr2 = komut2.ExecuteReader();  
    while (dr2.Read())  
    {  
        CmbBrans.Items.Add(dr2[0]);  
    }  
    bgl.baglanti().Close();
```

komutlarında Branşlar tablosundan branş adlarını çektim. dr2 isimli sql veri okuyucuyu çalıştırdım. Eğer okunduysa branş combobox'ının içine yazdırdım ve bağlantıyı kapattım.

```
DataTable dt2 = new DataTable();  
    SqlDataAdapter da2 = new SqlDataAdapter("SELECT _Tbl_Ran  
    da2.Fill(dt2);  
    dataGridView1.DataSource = dt2;
```

komutlarında dt2 isimli veri tablosu oluşturdum. da2 isimli sql veri uyarlayıcısına sql bağlantısını yapıp komutu yazdım. Sql komutunda randevu id, doktor ad ve soyad, randevu tarih, randevu saat, randevu durum ve randevu şikayetini HastaTC'sine göre randevu durumu aktif olan randevuları getir dedim. Sql sorgusuna göre dt2 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'nin veri kaynağında tabloyu gösterdim.

CmbBrans\_SelectedIndexChanged içinde;

```
CmbDoktor.Items.Clear();  
    SqlCommand komut3 = new SqlCommand("SELECT _DoktorAd , Dok  
    komut3.Parameters.AddWithValue("@p1", CmbBrans.Text);  
    SqlDataReader dr3 = komut3.ExecuteReader();  
    while (dr3.Read())
```

```

{
    CmbDoktor.Items.Add(dr3[0] + " " + dr3[1]);
}
bgl.baglanti().Close();

```

komutlarında branş combobox'ını seçtiğimizde doktor combobox'ında eski doktor isim ve soyisimleri olmaması için temizledim. Doktor combobox'ına doktor ad ve soyadlarını Doktorlar tablosundan branş combobox'ında seçilen branş adına göre getirdim. dr3 isimli sql veri okuyucuyu komutla çalıştırdım. Okunurken doktor combobox'ına doktor ad ve soyadlarını arada boşluk bırakarak yazdırdım ve bağlantıyı kapattım.

CmbDoktor\_SelectedIndexChanged içinde;

```

DataTable dt = new DataTable();
SqlDataAdapter da = new SqlDataAdapter("select_Randevu");
da.Fill(dt);
dataGridView2.DataSource = dt;

```

komutlarında dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcısına komutu yazdım. Randevu id, randevu tarih ve randevu saati Randevular tablosundan branş adına, doktor adına ve soyadına göre pasif olan randevuları sql bağlantısını sağlayarak getirdim. da isimli sql veri uyarlayıcısıyla dt isimli veri tablosunu doldur dedim. dt isimli veri tablosunu da veri kafesi görünümü 2'nin veri kaynağına aktar dedim.

LnkBilgiDuzenle\_LinkClicked içinde;

```

FrmBilgiDuzenle fr = new FrmBilgiDuzenle();
fr.TCno = LblTC.Text;
fr.Show();

```

komutlarında FrmBilgiDuzenle formunun TCno'suna etiketteki TC'yi aktardım ve bilgi düzenle formunu gösterdim.

dataGridView2\_CellClick içinde;

```
int secilen = dataGridView2.SelectedCells[0].RowIndex;
    Txtid.Text = dataGridView2.Rows[secilen].Cells[0].Value;
```

komutlarında alınmamış randevuların yani pasif randevuların id'sini Txtid'nin yazısına getirdim. Bunu veri kafesi görünümü 2'den yaptım.

TabloGuncelle fonksiyonu içinde;

```
private void TabloGuncelle()
{
    //Tablo Güncelleme
    DataTable dt2 = new DataTable();
    SqlDataAdapter da2 = new SqlDataAdapter("SELECT Tbl_Ran
da2.Fill(dt2);
    dataGridView1.DataSource = dt2;
```

komutlarında dt2 isimli veri tablosu oluşturdum. da2 isimli sql veri uyarlayıcıya sql komutunu yazdım. Randevu id, doktor ad ve soyad, randevu tarihi, randevu saati, randevu durumu ve randevu şikayetini Randevular ve Doktorlar tablosundan HastaTC'ye göre randevu durumu aktif olan randevuları sql bağlantısını sağlayarak getir dedim. da2 isimli sql veri uyarlayıcıya dt2 isimli veri tablosunu doldur dedim ve veri kafes görünümü 1'in veri kaynağına dt2 isimli veri tablosunu aktar dedim.

TabloGuncelle fonksiyonu içinde;

```
DataTable dt = new DataTable();
    SqlDataAdapter da = new SqlDataAdapter("select_Randevu
da.Fill(dt);
    dataGridView2.DataSource = dt;
}
```

komutlarında dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcıya sql komutunu yazdım. Randevu id, randevu tarihi ve randevu saatini Randevular tablosundan branş adı ve doktor ad ve soyadına göre randevu durumu pasif yani alınmamış olan randevuları

sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim ve veri kafes görünümü 2'nin veri kaynağına dt isimli veri tablosunu aktar dedim.

Bu TabloGuncelle fonksiyonu diğer tıklamalarda kullanılmak için yazıldı.

button1\_Click içinde;

```
SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevular_SET_RandevuDurumu=0 WHERE RandevuID=@id", bgl.baglanti());
komut.ExecuteNonQuery();
bgl.baglanti().Close();
MessageBox.Show("Randevu iptal edildi.", "Bilgi", MessageBoxButtons.OK);

TabloGuncelle();
```

Randevuyu iptal et butonunda randevular tablosundan randevu durumunu pasif ve tablolar id'lerle birbirine bağlı olduğundan Hastaid'sini null olarak Randevuid'ye göre sql bağlantısını sağlayarak güncelle dedim. Komutu çalıştırdım ve sql bağlantısını kapattım. "Randevu iptal edildi." şeklinde randevu iptal bilgi mesajı gösterdim ve tabloları güncelledim.

dataGridView1\_CellClick içinde;

```
int secilen = dataGridView1.SelectedCells[0].RowIndex;
Txtid.Text = dataGridView1.Rows[secilen].Cells[0].Value;
```

komutlarında alınmış randevuların yani aktif randevuların id'sini Txtid'nin yazısına getirdim. Bunu veri kafesi görünümü 1'den yaptım.

button2\_Click içinde;

```
Application.Exit();
```

komutuyla uygulamadan çık dedim.

BtnRandevuAl\_Click içinde;

```
SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevular_SET_Randevu
    komut.Parameters.AddWithValue("@r2", RchSikayet.Text);
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Randevu_oluşturuldu.", "Bilgi", Messag

    TabloGuncelle();
```

Randevu al butonunda randevular tablosundan randevu durumunu aktif ve HastaTC'sine göre Hastaid'sini ve şikayeti Randevuid'sine göre sql bağlantısını sağlayarak güncelledim. Şikayeti komuta parametre olarak gönderdim. Komutu çalıştırıp sql bağlantısını kapattım. "Randevu oluşturuldu." şeklinde bilgi mesajı verdim ve tabloları güncelledim.

**Dosya 5:** Bilgi Düzenle Formu:

```
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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmBilgiDuzenle : Form
    {
        public FrmBilgiDuzenle()
        {
```

---

---

```

        InitializeComponent();
    }

    public string TCno;

    SqlBaglantisi bgl = new SqlBaglantisi();
    private void FrmBilgiDuzenle_Load(object sender, EventArgs e)
    {
        MskTC.Text = TCno;
        SqlCommand komut = new SqlCommand("select_*_from_Tbl_Hasta", bgl.baglanti());
        komut.Parameters.AddWithValue("@p1", MskTC.Text);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())
        {
            TxtAd.Text = dr[1].ToString();
            TxtSoyad.Text = dr[2].ToString();
            MskTelefon.Text = dr[4].ToString();
            TxtSifre.Text = dr[5].ToString();
            CmbCinsiyet.Text = dr[6].ToString();
        }
        bgl.baglanti().Close();
    }

    private void BtnBilgiGuncelle_Click(object sender, EventArgs e)
    {
        if (FrmHastaKayit.TCKontrol(MskTC.Text)) {
            SqlCommand komut2 = new SqlCommand("update_Tbl_Hasta", bgl.baglanti());
            komut2.Parameters.AddWithValue("@p1", TxtAd.Text);
            komut2.Parameters.AddWithValue("@p2", TxtSoyad.Text);
            komut2.Parameters.AddWithValue("@p3", MskTelefon.Text);
            komut2.Parameters.AddWithValue("@p4", TxtSifre.Text);
            komut2.ExecuteNonQuery();
            bgl.baglanti().Close();
        }
    }

```

```

        komut2.Parameters.AddWithValue("@p5", CmbCinsiyet.Text);
        komut2.Parameters.AddWithValue("@p6", MskTC.Text);
        komut2.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Bilgileriniz_güncellendi.", "Bilgi");
    }
}
}
}

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

SqlBaglantisi bgl = new SqlBaglantisi();

komutuyla Sql bağlantısını yaptım.

public string TCno;

komutuyla her yerden ulaşılabilir TCno'yu tanımladım.

FrmBilgiDuzenle_Load içinde;

MskTC.Text = TCno;

        SqlCommand komut = new SqlCommand("select *_from_Tbl_Ha
        komut.Parameters.AddWithValue("@p1", MskTC.Text);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())
        {
            TxtAd.Text = dr[1].ToString();
            TxtSoyad.Text = dr[2].ToString();
            MskTelefon.Text = dr[4].ToString();
            TxtSifre.Text = dr[5].ToString();

```

```

        CmbCinsiyet.Text = dr[6].ToString();
    }
    bgl.baglanti().Close();

```

komutlarında TCno'yu MskTC'nin yazısına aktardım. Hastalar tablosundan HastaTC'ye göre MskTC'nin yazısını parametre olarak verileri getiren komutu yazdım. dr isimli sql veri okuyucuyu komuta göre çalıştırdım. Okunurken ad, soyad ve şifreyi text'e, telefonu masked text'e ve cinsiyeti combobox'a bilgileri aktardım. Sonra bağlantıyı kapattım.

BtnBilgiGuncelle\_Click içinde;

```

if (FrmHastaKayit.TCKontrol(MskTC.Text)) {
    SqlCommand komut2 = new SqlCommand("update_Tbl_Hasta
    komut2.Parameters.AddWithValue("@p1", TxtAd.Text);
    komut2.Parameters.AddWithValue("@p2", TxtSoyad.Text);
    komut2.Parameters.AddWithValue("@p3", MskTelefon.Text);
    komut2.Parameters.AddWithValue("@p4", TxtSifre.Text);
    komut2.Parameters.AddWithValue("@p5", CmbCinsiyet.Text);
    komut2.Parameters.AddWithValue("@p6", MskTC.Text);
    komut2.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Bilgileriniz_güncellendi.", "Bilgi")
}

```

komutlarında TC gerçekse işlemleri yapacak. Önce TC kontrolü yaptım. Sql komutunda Hastalar tablosunda hasta ad, hasta soyad, hasta telefon, hasta şifre ve hasta cinsiyeti HastaTC'ye göre ilgili parametreleri olarak ve sql bağlantısını yaparak yazdım ve çalıştırdım. Sonra sql bağlantısını kapatıp "Bilgileriniz güncellendi." şeklinde bilgi mesajı verdirdim.

**Form 6:** Doktor Detay Formu:

```

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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmDoktorDetay : Form
    {
        public FrmDoktorDetay()
        {
            InitializeComponent();

            SqlBaglantisi bgl = new SqlBaglantisi();

            public string DoktorTCsi;
            private void FrmDoktorDetay_Load(object sender, EventArgs e)
            {
                LblTC.Text = DoktorTCsi;
                //Doktor Ad Soyad Çekme
                SqlCommand komut = new SqlCommand("select _DoktorAd ,DoktorSoyad from _Doktorlar where TC=@d1", bgl);
                komut.Parameters.AddWithValue("@d1", LblTC.Text);
                SqlDataReader dr = komut.ExecuteReader();

                //Doktor Ad, Soyad ve Randevu Listesi
                while (dr.Read())
                {
```

---

---

```

        String DoktorAd = dr[0].ToString();
        String DoktorSoyad = dr[1].ToString();
        LblAdSoyad.Text = DoktorAd + "_" + DoktorSoyad;
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_Tbl_
        da.Fill(dt);
        dataGridView1.DataSource = dt;
    }
    bgl.baglanti().Close();
}

private void BtnGuncelle_Click(object sender, EventArgs e)
{
    FrmDoktorBilgiDuzenle dbd = new FrmDoktorBilgiDuzenle()
    dbd.DoktorTCsidir = LblTC.Text;
    dbd.Show();
}

private void BtnDuyurular_Click(object sender, EventArgs e)
{
    FrmDuyurular d = new FrmDuyurular();
    d.Show();
}

private void BtnCikis_Click(object sender, EventArgs e)
{
    Application.Exit();
}

private void dataGridView1_CellClick(object sender, DataGri
{

```

---



---

```

        int secilen = dataGridView1.SelectedCells[0].RowIndex;
        RchSikayet.Text = dataGridView1.Rows[secilen].Cells[4].Text;
        txtRandevuid.Text = dataGridView1.Rows[secilen].Cells[0].Text;
    }

    private void dataGridView1_CellContentClick(object sender, EventArgs e)
    {

    }

    private void btnTamamla_Click(object sender, EventArgs e)
    {
        SqlCommand komut2 = new SqlCommand("delete_from_Tbl_Randevu", conn);
        komut2.Parameters.AddWithValue("@p1", txtRandevuid.Text);
        komut2.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_Tamamlandı.", "Bilgi", MessageBoxButtons.OK);
        TabloGuncelle();
    }

    private void TabloGuncelle()
    {
        LblTC.Text = DoktorTCsi;
        //Doktor Ad Soyad Çekme
        SqlCommand komut = new SqlCommand("select_DoktorAd,DoktorSoyad", conn);
        komut.Parameters.AddWithValue("@d1", LblTC.Text);
        SqlDataReader dr = komut.ExecuteReader();

        while (dr.Read())
        {
            String DoktorAd = dr[0].ToString();

```

```

        String DoktorSoyad = dr[1].ToString();
        LblAdSoyad.Text = DoktorAd + "_" + DoktorSoyad;
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_Tbl_
da.Fill(dt);
        dataGridView1.DataSource = dt;
    }
    bgl.baglanti().Close();
}
}
}
}

```

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

```
SqlBaglantisi bgl = new SqlBaglantisi();
```

komutuyla Sql bağlantısını yaptım.

```
public string DoktorTCsi;
```

komutuyla her yerden ulaşılacak TCno'yu tanımladım.

FrmDoktorDetay\_Load içinde;

```
LblTC.Text = DoktorTCsi;
```

```
//Doktor Ad Soyad Çekme
```

```
SqlCommand komut = new SqlCommand("select_DoktorAd,Dokt
```

```
komut.Parameters.AddWithValue("@d1", LblTC.Text);
```

```
SqlDataReader dr = komut.ExecuteReader();
```

```
//Doktor Ad, Soyad ve Randevu Listesi
```

```
while (dr.Read())
```

```
{
```

```
String DoktorAd = dr[0].ToString();
String DoktorSoyad = dr[1].ToString();
LblAdSoyad.Text = DoktorAd + " " + DoktorSoyad;
```

DoktorTCsi'ni LblTC'nin yazısına aktardım. Sql komutunda doktor ad, doktor soyad ve doktor id'yi Doktorlar tablosundan DoktorTC'ye göre LblTC'nin yazısını parametre olarak ve sql bağlantısını yaparak yazdım ve çalıştırdım.

dr isimli sql veri okuyucuyu komutla çalıştırdım. Okurken Doktor ad ve soyadı aldım ve etikete aktardım.

```
DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select _Tbl_
        da.Fill(dt);
        dataGridView1.DataSource = dt;
    }
    bgl.baglanti().Close();
```

dt isimli veri tablosunu oluşturdum. Sql veri uyarlayıcıda sql komutunu yazdım. Sql komutunda randevu id, hasta ad ve soyad, randevu tarihi, randevu saati, randevu durum ve randevu şikayeti randevular ve hastalar tablosundan doktor id'ye göre randevu durumu aktif olan randevuları sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcısına dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım ve sql bağlantısını kapattım.

BtnDuyurular\_Click içinde;

```
FrmDuyurular d = new FrmDuyurular();
        d.Show();
```

komutlarında FrmDuyurular formunu açtım.

BtnCikis\_Click içinde;

```
Application.Exit();
```

---

---

komutuyla uygulamadan çık dedim.

dataGridView1\_CellClick içinde;

```
int secilen = dataGridView1.SelectedCells[0].RowIndex;  
RchSikayet.Text = dataGridView1.Rows[secilen].Cells[6].  
txtRandevuid.Text = dataGridView1.Rows[secilen].Cells[0].
```

komutlarında randevu id ve şikayeti ilgili yerlere veri kafesi görünümü 1'de tıklayınca aktardım.

TabloGuncelle fonksiyonu içinde;

```
LblTC.Text = DoktorTCsi;  
//Doktor Ad Soyad Çekme  
SqlCommand komut = new SqlCommand("select _DoktorAd ,DoktorSoyad from Doktorlar where DoktorTC=@d1", bgl.baglanti());  
komut.Parameters.AddWithValue("@d1", LblTC.Text);  
SqlDataReader dr = komut.ExecuteReader();  
  
while (dr.Read())  
{  
    String DoktorAd = dr[0].ToString();  
    String DoktorSoyad = dr[1].ToString();  
    LblAdSoyad.Text = DoktorAd + "_" + DoktorSoyad;  
    DataTable dt = new DataTable();  
    SqlDataAdapter da = new SqlDataAdapter("select _TblRandevu from Randevu where DoktorTC=@d1", bgl.baglanti());  
    da.Fill(dt);  
    dataGridView1.DataSource = dt;  
}  
bgl.baglanti().Close();
```

DoktorTCsi'ni LblTC'nin yazısına aktardım. Sql komutunda doktor ad, doktor soyad ve doktor id'yi Doktorlar tablosundan DoktorTC'ye göre LblTC'nin yazısını parametre alarak ve sql bağlantısını yaparak yazdım ve çalıştırdım.

dr isimli sql veri okuyucuyu komutla çalıştırdım. Okurken Doktor ad ve soyadı aldım ve etikete aktardım.

dt isimli veri tablosunu oluşturdum. Sql veri uyarlayıcıda sql komutunu yazdım. Sql komutunda randevu id, hasta ad ve soyad, randevu tarihi, randevu saati, randevu durum ve randevu şikayeti randevular ve hastalar tablosundan doktor id'ye göre randevu durumu aktif olan randevuları sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcısına dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım ve sql bağlantısını kapattım.

btnTamamla\_Click içinde;

```
SqlCommand komut2 = new SqlCommand("delete_from_Tbl_Randevular_where  
komut2.Parameters.AddWithValue("@p1", txtRandevuid.Text  
komut2.ExecuteNonQuery();  
bgl.baglanti().Close();  
MessageBox.Show("Randevu_Tamamlandı.", "Bilgi", Message  
TabloGuncelle());
```

komutlarında Sql komutunu yazdım. Sql komutunda randevular tablosundan Randevuid'ye göre txtRandevuid'nin yazısını parametre olarak randevuyu sildim ve sorguyu çalıştırdım. Sonra bağlantıyı kapatıp "Randevu Tamamlandı." şeklinde bilgi mesajı verdirdim. En son aktif randevu tablosunu güncelledim.

**Form 7:** Duyurular Formu:

```
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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmDuyurular : Form
    {
        public FrmDuyurular()
        {
            InitializeComponent();

            SqlBaglantisi bgl = new SqlBaglantisi();

            private void FrmDuyurular_Load(object sender, EventArgs e)
            {
                DataTable dt = new DataTable();
                SqlDataAdapter da = new SqlDataAdapter("select _Duyuruid
da.Fill(dt);
dataGridView1.DataSource = dt;
            }
        }
    }
}
```

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

komutuyla Sql komutlarının çalışması için SqlClient kütüphanesini ekledim.

```
SqlBaglantisi bgl = new SqlBaglantisi();
```

komutuyla Sql bağlantısını yaptım.

FrmDuyurular\_Load içinde;



```

DataTable dt = new DataTable();

        SqlDataAdapter da = new SqlDataAdapter("select _Duyuruid
da.Fill(dt);

dataGridView1.DataSource = dt;

```

dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutunda Duyurular tablosundan Duyuruid ve Duyuru'yu sql bağlantısını da yaparak çağırdım. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi Görünümü 1'in veri kaynağına dt isimli veri tablosunu aktar dedim.

#### **Dosya 8:** Sekreter Detay Formu:

```

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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmSekreterDetay : Form
    {
        public FrmSekreterDetay()
        {
            InitializeComponent();
        }
    }
}

```

---



---

```

        public string SekreterTC;

        SqlBaglantisi bgl = new SqlBaglantisi();

        private void FrmSekreterDetay_Load(object sender, EventArgs e)
        {
            LblTC.Text = SekreterTC;
            // Sekreter AdSoyad Çekme
            SqlCommand komut = new SqlCommand("select _SekreterAd , SekreterTC from Tbl_Sekreter where SekreterTC=@p1", bgl.baglanti());
            komut.Parameters.AddWithValue("@p1", LblTC.Text);
            SqlDataReader dr1 = komut.ExecuteReader();
            while (dr1.Read())
            {
                LblAdSoyad.Text = dr1[0].ToString() + " " + dr1[1].ToString();
            }
            bgl.baglanti().Close();

            //Randevuları Datagridview1'e aktarma
            DataTable dt = new DataTable();
            SqlDataAdapter da = new SqlDataAdapter("select _Tbl_Randevu from Tbl_Randevu", bgl.baglanti());
            da.Fill(dt);
            dataGridView1.DataSource = dt;

            // Branşı Combobox'a aktarma
            SqlCommand komut2 = new SqlCommand("select _BransAd from Tbl_Brans", bgl.baglanti());
            SqlDataReader dr2 = komut2.ExecuteReader();
            while (dr2.Read())
            {
                CmbBrans.Items.Add(dr2[0]);
            }
            bgl.baglanti().Close();
        }
    }

```

---

---

```

    }

    private void TabloyuGuncelle()
    {
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select _Tbl_Randevu", conn);
        da.Fill(dt);
        dataGridView1.DataSource = dt;
    }

    private void BtnKaydet_Click(object sender, EventArgs e)
    {
        SqlCommand komutkaydet = new SqlCommand("insert_into_Tbl_Randevu", conn);
        komutkaydet.Parameters.AddWithValue("@p1", MskTarih.Text);
        komutkaydet.Parameters.AddWithValue("@p2", MskSaat.Text);
        komutkaydet.Parameters.AddWithValue("@p3", CmbBrans.Text);
        komutkaydet.Parameters.AddWithValue("@p4", CmbDoktor.Text);
        komutkaydet.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_oluşturuldu.", "Bilgi", MessageBoxButtons.OK);
        TabloyuGuncelle();
    }

    private void CmbBrans_SelectedIndexChanged(object sender, EventArgs e)
    {
        CmbDoktor.Items.Clear();
        SqlCommand komut = new SqlCommand("select _DoktorAd, DoktorAd", conn);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())
        {
            CmbDoktor.Items.Add(dr[0] + "_" + dr[1]);
        }
    }

```

---

---

```

        }
        bg1.baglanti().Close();
    }

    private void BtnDuyuruOlustur_Click(object sender, EventArgs e)
    {
        SqlCommand komut = new SqlCommand("insert_into_Tbl_Duyuru", bg1.baglanti());
        komut.Parameters.AddWithValue("@dl", RchDuyuru.Text);
        komut.ExecuteNonQuery();
        bg1.baglanti().Close();
        MessageBox.Show("Duyuru_oluřturuldu.", "Bilgi", MessageBoxButtons.OK);
    }

    private void BtnDoktorPaneli_Click(object sender, EventArgs e)
    {
        FrmDoktorPaneli drp = new FrmDoktorPaneli();
        drp.Show();
    }

    private void BtnBransPaneli_Click(object sender, EventArgs e)
    {
        FrmBransPaneli frb = new FrmBransPaneli();
        frb.Show();
    }

    private void BtnRandevuListe_Click(object sender, EventArgs e)
    {
        FrmRandevuListesi rl = new FrmRandevuListesi();
        rl.Show();
    }

```

---

---

```
private void button1_Click(object sender, EventArgs e)
{
    FrmDuyurular frd = new FrmDuyurular();
    frd.Show();
}

private void BtnGuncelle_Click(object sender, EventArgs e)
{
    //SqlCommand komut2 = new SqlCommand("update_Tbl_Randevu

    SqlCommand komut2 = new SqlCommand("update_Tbl_Randevu
    komut2.Parameters.AddWithValue("@p1", MskTarih.Text);
    komut2.Parameters.AddWithValue("@p2", MskSaat.Text);
    komut2.Parameters.AddWithValue("@p3", CmbBrans.Text);
    komut2.Parameters.AddWithValue("@p4", DoktorAd);
    komut2.Parameters.AddWithValue("@p5", DoktorSoyad);
    komut2.Parameters.AddWithValue("@p6", Txtid.Text);
    komut2.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Bilgileriniz_güncellendi.", "Bilgi", M
    TabloyuGuncelle();
}

String DoktorAd, DoktorSoyad;
private void dataGridView1_CellClick(object sender, DataGri
{
    int secilen = dataGridView1.SelectedCells[0].RowIndex;
    Txtid.Text = dataGridView1.Rows[secilen].Cells[0].Value
    MskTarih.Text = dataGridView1.Rows[secilen].Cells[4].Va
    MskSaat.Text = dataGridView1.Rows[secilen].Cells[5].Val
    CmbBrans.Text = dataGridView1.Rows[secilen].Cells[3].Va
```

```

        CmbDoktor.Text = dataGridView1.Rows[secilen].Cells[1].Value;
        DoktorAd = dataGridView1.Rows[secilen].Cells[1].Value.ToString();
        DoktorSoyad = dataGridView1.Rows[secilen].Cells[2].Value.ToString();
    }

    private void button2_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        FrmDoktorVeBranslar frd = new FrmDoktorVeBranslar();
        frd.Show();
    }

    private void button4_Click(object sender, EventArgs e)
    {
        SqlCommand komut = new SqlCommand("DELETE_FROM_Tbl_Randevu", bgl.baglanti());
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu silindi.", "Bilgi", MessageBoxButtons.OK);
        TabloyuGuncelle();
    }
}

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

SqlBaglantisi bgl = new SqlBaglantisi();

```

---

---

komutuyla Sql bağlantısını yaptım.

```
public string SekreterTC ;
```

komutuyla SekreterTC değişkenine Sekreter TC'yi almak üzere diğer formlardan ulaşılacak şekilde oluşturdum.

FrmSekreterDetay\_Load içinde;

```
LblTC . Text = SekreterTC ;
```

```
    // Sekreter AdSoyad Çekme
```

```
    SqlCommand komut = new SqlCommand(" select _SekreterAd , Se
```

```
    komut . Parameters . AddWithValue ( "@p1" , LblTC . Text );
```

```
    SqlDataReader dr1 = komut . ExecuteReader ();
```

```
    while ( dr1 . Read () )
```

```
    {
```

```
        LblAdSoyad . Text = dr1 [ 0 ] . ToString () + " _ " + dr1 [ 1 ] .
```

```
    }
```

```
    bgl . baglanti () . Close ();
```

komutlarında SekreterTC'yi LblTC'nin yazısında gösterdim. Sql komutta Sekreter ad ve soyadını Sekreterler tablosundan SekreterTC'ye göre LblTC'nin yazısını parametre alarak ve sql bağlantısını sağlayarak yazdım. dr1 isimli sql veri okuyucuyu komutu okuyarak çalıştırdım. Okurken Sekreter ad ve soyadı arada boşluk bırakarak ad soyad etiketine yazdırdım ve bağlantıyı kapattım.

```
// Randevuları Datagridview1'e aktarma
```

```
DataTable _dt = new DataTable ();
```

```
SqlDataAdapter _da = new SqlDataAdapter ( " select _Tbl_Rand
```

```
_da . Fill ( dt );
```

```
dataGridView1 . DataSource = _dt ;
```

dt isimli veri tablosunu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta randevu id'si, doktor adı ve soyadı, branş adı, randevu tarihi, randevu saati, randevu durumu ve şikayeti randevular, doktorlar ve branşlar tablosundan sql bağlantısını

---

---

yaparak çektim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım.

```
// Branşı Combobox'a aktarma
        SqlCommand_komut2=_new_SqlCommand(" select _BransAd_from,
        SqlDataReader_dr2=_komut2 . ExecuteReader ();
        while_( dr2 . Read ())
        {
            CmbBrans . Items . Add( dr2 [ 0 ] );
        }
        bgl . baglanti () . Close ();
```

komutlarında sql komutu yazdım. Sql komutta Branşlar tablosundan branş adlarını sql bağlantısını yaparak çektim. dr2 isimli sql veri okuyucuyu komutu okutarak çalıştırdım. Okurken branş combobox'ına branşları ekledim ve bağlantıyı kapattım.

TabloyuGuncelle fonksiyonu içinde;

```
DataTable dt = new DataTable ();
        SqlDataAdapter da = new SqlDataAdapter(" select _Tbl_Randevular
        da . Fill ( dt );
        dataGridView1 . DataSource = dt ;
```

dt isimli veri tablosunu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta randevu id'si, doktor adı ve soyadı, branş adı, randevu tarihi, randevu saati, randevu durumu ve şikayeti randevular, doktorlar ve branşlar tablosundan sql bağlantısını yaparak çektim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım.

BtnKaydet\_Click içinde;

```
SqlCommand komutkaydet = new SqlCommand(" insert _into _Tbl_Randevular
        komutkaydet . Parameters . AddWithValue ( "@p1" , MskTarih . Text
        komutkaydet . Parameters . AddWithValue ( "@p2" , MskSaat . Text
        komutkaydet . Parameters . AddWithValue ( "@p3" , CmbBrans . Text
```



```

        komutkaydet . Parameters . AddWithValue ( "@p4" , CmbDoktor . Text );
        komutkaydet . ExecuteNonQuery ();
        bgl . baglanti () . Close ();
        MessageBox . Show ( "Randevu_olusturuldu . " , " Bilgi " , MessageBoxButtons . Ok );
        TabloyuGuncelle ();

```

komutlarında sql komutu yazdım. Sql komutta randevular tablosuna randevu tarihi, randevu saati, branş adına göre branş id'yi, doktor ad ve soyadına göre doktor id'yi ve randevu durumunu pasif olarak sql bağlantısını sağlayıp p1 için MskTarih yazısını, p2 için MskSaat yazısını, p3 için CmbBrans yazısını ve p4 için CmbDoktor yazısını parametre olarak alıp kaydettim ve sql komutu çalıştırdım. Sonra bağlantıyı kapatıp "Randevu oluşturuldu." bilgi mesajı verdim ve randevular tablosunu güncelledim.

CmbBrans\_SelectedIndexChanged içinde;

CmbDoktor . Items . Clear ();

```

        SqlCommand komut = new SqlCommand("select _DoktorAd , DoktorSoyad from Doktorlar");
        SqlDataReader dr = komut . ExecuteReader ();
        while ( dr . Read ())
        {
            CmbDoktor . Items . Add ( dr [0] + " _ " + dr [1] );
        }
        bgl . baglanti () . Close ();

```

komutlarında branş combobox'ını seçtiğimizde doktor combobox'ını temizledim. Sql komutunda doktor ad ve soyadını Doktorlar tablosundan branş adına göre branş combobox'ını parametre olarak ve sql bağlantısını yapıp getir dedim. dr adlı sql veri okuyucuyu komutu okuyarak çalıştır dedim. Okurken doktor combobox'ına ad ve soyadı arada boşluk bırakarak ekle dedim ve bağlantıyı kapattım.

BtnDuyuruOlustur\_Click içinde;

```

        SqlCommand komut = new SqlCommand("insert_into _Tbl_Duyurular (DuyuruTarihi , DuyuruMetni , DuyuruDurumu) values (@d1 , @d2 , @d3)");
        komut . Parameters . AddWithValue ( "@d1" , RchDuyuru . Text );

```

```

        komut . ExecuteNonQuery ();
        bgl . baglanti (). Close ();
        MessageBox . Show ( " Duyuru _ oluřturuldu . " , " Bilgi " , MessageBo

```

Sql komutta Duyurular tablosuna richtextbox'ın yazısını parametre olarak ve sql bağlantısını yapıp duyuruyu kaydettim. Sql komutunu çalıştırdım ve bağlantıyı kapattım. Sonra "Duyuru oluşturuldu." şeklinde bilgi mesajı verdirdim.

BtnRandevuListe\_Click içinde;

```

FrmRandevuListesi r1 = new FrmRandevuListesi ();
        r1 . Show ();

```

komutunda RandevuListesi formunu açtım.

button1\_Click içinde;

```

FrmDuyurular frd = new FrmDuyurular ();
        frd . Show ();

```

komutunda Duyurular formunu açtım.

BtnGuncelle\_Click içinde;

```

SqlCommand komut2 = new SqlCommand ( " update _ Tbl _ Randevular _ set _ Rande
        komut2 . Parameters . AddWithValue ( "@p1 " , MskTarih . Text );
        komut2 . Parameters . AddWithValue ( "@p2 " , MskSaat . Text );
        komut2 . Parameters . AddWithValue ( "@p3 " , CmbBrans . Text );
        komut2 . Parameters . AddWithValue ( "@p4 " , DoktorAd );
        komut2 . Parameters . AddWithValue ( "@p5 " , DoktorSoyad );
        komut2 . Parameters . AddWithValue ( "@p6 " , Txtid . Text );
        komut2 . ExecuteNonQuery ();
        bgl . baglanti (). Close ();
        MessageBox . Show ( " Bilgileriniz _ güncellendi . " , " Bilgi " , M
        TabloyuGuncelle ();

```

---

---

komutlarında sql komutunu yazdım. Sql komutunda randevular tablosunda randevu tarihi ve randevu saati randevu id'ye göre MskTarih'in yazısını ve MskSaat'in yazısını parametre olarak ve sql bağlantısını da yaparak güncelledim. Komutu çalıştırıp sql bağlantısını kapattım. Sonra "Bilgileriniz güncellendi." şeklinde bilgi mesajı verip randevular tablosunu güncelledim.

dataGridView1\_CellClick içinde;

```
int secilen = dataGridView1.SelectedCells[0].RowIndex;  
    Txtid.Text = dataGridView1.Rows[secilen].Cells[0].Value;  
    MskTarih.Text = dataGridView1.Rows[secilen].Cells[4].Value;  
    MskSaat.Text = dataGridView1.Rows[secilen].Cells[5].Value;  
    CmbBrans.Text = dataGridView1.Rows[secilen].Cells[3].Value;  
    CmbDoktor.Text = dataGridView1.Rows[secilen].Cells[1].Value;
```

komutlarında randevu id'yi Txtid'nin yazısına, randevu tarihi MskTarih'in yazısına, randevu saaati MskSaat'in yazısına, randevu branşı branş combobox'ın yazısına ve randevu doktorunun ad ve soyadını arada boşluk bırakarak doktor combobox'ının yazısına tabloda satıra tek tıklatınca aktardım.

button2\_Click içinde;

```
Application.Exit();
```

komutunda uygulamadan çık dedim.

button3\_Click içinde;

```
FrmDoktorVeBranslar frd = new FrmDoktorVeBranslar();  
    frd.Show();
```

komutunda DoktorVeBranslar formunu açtım.

button4\_Click içinde;

```
SqlCommand komut = new SqlCommand("DELETE_FROM_Tbl_Randevular WHERE  
    komut.ExecuteNonQuery();
```

---

---

```
        bgl.baglanti().Close();  
        MessageBox.Show("Randevu_silindi.", "Bilgi", MessageBoxButtons.OK);  
        TabloyuGuncelle();
```

komutunda sql komutu yazdım. Randevu sil butonunda yazdım. Sql komutunda Randevular tablosundan Randevuid'ye göre sql bağlantısını yaparak sildim ve komutu çalıştırdım. Sonra bağlantıyı kapatıp "Randevu silindi." şeklinde bilgi mesajı verdirdim. En son randevular tablosunu güncelledim.

**Dosya 9:** Randevu Listesi Formu:

```
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 System.Data.SqlClient;  
  
namespace Yonetim_Hastane  
{  
    public partial class FrmRandevuListesi : Form  
    {  
        public FrmRandevuListesi()  
        {  
            InitializeComponent();  
        }  
  
        SqlBaglantisi bgl = new SqlBaglantisi();
```

---

---

```

        private void FrmRandevuListesi_Load(object sender, EventArgs e)
        {
            DataTable dt = new DataTable();
            SqlDataAdapter da = new SqlDataAdapter("select * from Tbl_Randevu", conn);
            da.Fill(dt);
            dataGridView1.DataSource = dt;
        }

        public int secilen;
        private void dataGridView1_CellDoubleClick(object sender, DataGridViewCellEventArgs e)
        {
            secilen = dataGridView1.SelectedCells[0].RowIndex;
        }
    }
}

using System.Data.SqlClient;

```

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

```
SqlBaglantisi bgl = new SqlBaglantisi();
```

komutuyla Sql bağlantısını yaptım.

FrmRandevuListesi\_Load içinde;

```

DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select * from Tbl_Randevu", conn);
        da.Fill(dt);
        dataGridView1.DataSource = dt;

```

dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutunda randevu id'si, hasta adı ve soyadı, branş adı, doktor adı ve soyadı, randevu tarihi,

randevu saati, randevu durumu ve şikayeti Randevular, Branşlar, Doktorlar ve Hastalar tablosundan gerekli sql bağlantısını yaparak çektim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım.

```
public int secilen;
```

komutunda seçilene diğer formlardan ve tıklamalardan da erişebilecek şekilde tamsayı olarak ayarladım.

dataGridView1\_CellDoubleClick içinde;

```
secilen = dataGridView1.SelectedCells[0].RowIndex;
```

komutunda seçilene veri kafesi görünümü 1'in seçili 0. hücrenin satır indeksini aktardım. Bu komutla aynı zamanda form açıldığında seçili satırın ilk hücrenin seçili olmasını sağladım.

**Dosya 10:** Doktor ve Branşlar Formu:

```
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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmDoktorVeBranşlar : Form
    {
```

---

---

```
public FrmDoktorVeBranslar()
{
    InitializeComponent();

    SqlBaglantisi bgl = new SqlBaglantisi();

    private void DoktorVeBranslar_Load(object sender, EventArgs e)
    {
        DataTable dt1 = new DataTable();
        SqlDataAdapter da1 = new SqlDataAdapter("select _Doktorid
        da1.Fill(dt1);
        dataGridView1.DataSource = dt1;

        DataTable dt2 = new DataTable();
        SqlDataAdapter da2 = new SqlDataAdapter("select _Bransid
        da2.Fill(dt2);
        dataGridView2.DataSource = dt2;
    }
}
```

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

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

```
SqlBaglantisi bgl = new SqlBaglantisi();
```

komutuyla Sql bağlantısını yaptım.

DoktorVeBranslar\_Load içinde;

```
DataTable dt1 = new DataTable();
```

```
SqlDataAdapter da1 = new SqlDataAdapter("select _Doktorid
```

```
da1.Fill(dt1);
dataGridView1.DataSource = dt1;
```

dt1 isimli veri tablosu oluşturdum. da1 isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta doktor id'si, doktor ad'ı ve soyadı ve branş adını Doktorlar ve Branşlar tablosundan gerekli sql bağlantısını da yaparak çektim. da1 isimli sql veri uyarlayıcıya dt1 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt1 isimli veri tablosunu aktar dedim.

```
DataTable dt2 = new DataTable();
SqlDataAdapter da2 = new SqlDataAdapter("select Bransid,
da2.Fill(dt2);
dataGridView2.DataSource = dt2;
```

dt2 isimli veri tablosu oluşturdum. da2 isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta branş id'sini ve branş adını Branşlar tablosundan gerekli sql bağlantısını da yaparak çektim. da2 isimli sql veri uyarlayıcıya dt2 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 2'in veri kaynağına dt2 isimli veri tablosunu aktar dedim.

#### **Dosya 11: Müşteri Hizmetleri Formu:**

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

```
namespace Yonetim_Hastane
```



---

---

```

{
    public partial class FrmMusteriHizmetleri : Form
    {
        public FrmMusteriHizmetleri()
        {
            InitializeComponent();
        }

        SqlBaglantisi bgl = new SqlBaglantisi();

        public string MusteriHizmetleriTC;
        private void FrmMusteriHizmetleri_Load(object sender, EventArgs e)
        {
            SqlCommand komut2 = new SqlCommand("select _BransAd_from", bgl.baglanti());
            SqlDataReader dr2 = komut2.ExecuteReader();
            while (dr2.Read())
            {
                CmbBrans.Items.Add(dr2[0]);
            }
            bgl.baglanti().Close();

            lblMusteriHizmetleriTC.Text = MusteriHizmetleriTC;

            //Musteri Hizmetleri AdSoyad Çekme
            SqlCommand komut = new SqlCommand("select _MusteriHizmetAdSoyad", bgl.baglanti());
            komut.Parameters.AddWithValue("@pl", MusteriHizmetleriTC);
            SqlDataReader dr1 = komut.ExecuteReader();
            while (dr1.Read())
            {
                lblAdiSoyadi.Text = dr1[0].ToString() + "_" + dr1[1].ToString();
            }
        }
    }
}

```

---

---

```

        bgl.baglanti().Close();
    }

    private void TabloGuncelle()
    {
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_Randevuic
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

    private void CmbBrans_SelectedIndexChanged_1(object sender,
    {
        CmbDoktor.Items.Clear();
        SqlCommand komut3 = new SqlCommand("SELECT_DoktorAd , Dok
        komut3.Parameters.AddWithValue("@p1", CmbBrans.Text);
        SqlDataReader dr3 = komut3.ExecuteReader();
        while (dr3.Read())
        {
            CmbDoktor.Items.Add(dr3[0] + "_" + dr3[1]);
        }
        bgl.baglanti().Close();
    }

    private void CmbDoktor_SelectedIndexChanged(object sender,
    {
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select_Randevuic
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

```

---

---

```

private void BtnRandevuAl_Click(object sender, EventArgs e)
{
    SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevulari", bgl.baglanti());
    komut.Parameters.AddWithValue("@r2", RchSikayet.Text);
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Randevu_oluşturuldu.", "Bilgi", MessageBoxButtons.OK);
    TabloGuncelle();
}

private void dataGridView2_CellClick(object sender, DataGridViewCellEventArgs e)
{
    int secilen = dataGridView2.SelectedCells[0].RowIndex;
    Txtid.Text = dataGridView2.Rows[secilen].Cells[0].Value;
}

private void btnRandevularıGetir_Click(object sender, EventArgs e)
{
    DataTable dt = new DataTable();
    SqlDataAdapter da = new SqlDataAdapter("select_Randevulari", bgl.baglanti());
    da.Fill(dt);
    dataGridView2.DataSource = dt;
}

private void btnRandevuSil_Click(object sender, EventArgs e)
{
    SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevulari", bgl.baglanti());
    komut.Parameters.AddWithValue("@p1", Txtid.Text);
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
}

```

```

        MessageBox.Show("Randevu_Silindi.", "Bilgi", MessageBoxButtons.OK);
        TabloGuncelle();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}

```

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

SqlBaglantisi bgl = new SqlBaglantisi();

komutuyla Sql bağlantısını yaptım.

public string MusteriHizmetleriTC;

komutlarında Müşteri Hizmetlerinin sisteme girişte girdiği TC'sini almak için diğer formlardan da ulaşılabilir şekilde MusteriHizmetleriTC'yi string biçiminde tanımladım.

FrmMusteriHizmetleri\_Load içinde;

```

SqlCommand komut2 = new SqlCommand("select_BransAd_from_Tbl_Branslar");
SqlDataReader dr2 = komut2.ExecuteReader();
while (dr2.Read())
{
    CmbBrans.Items.Add(dr2[0]);
}
bgl.baglanti().Close();

```

Sql komutta brans adlarını Branslar tablosundan sql bağlantısını sağlayarak getir dedim.

dr2 isimli sql veri okuyucuyu komutu okutarak çalıştırdım. Okurken branş combobox'ına ekledim ve bağlantıyı kapattım.

```
lblMusteriHizmetleriTC.Text = MusteriHizmetleriTC;
```

komutunda MusteriHizmetleriTC'yi MusteriHizmetleriTC etiketiinin yazısına aktardım.

```
// Musteri Hizmetleri AdSoyad Çekme
```

```
SqlCommand komut = new SqlCommand("select _MusteriHizmetleriTC  
komut.Parameters.AddWithValue("@pl", MusteriHizmetleriTC  
SqlDataReader dr1 = komut.ExecuteReader();  
while (dr1.Read())  
{  
    lblAdiSoyadi.Text = dr1[0].ToString() + "_" + dr1[1].ToString()  
}  
bgl.baglanti().Close();
```

Sql komutta müşteri hizmetleri adı ve müşteri hizmetleri soyadını müşteri hizmetleri tablosundan MusteriHizmetleriTC'ye göre sql bağlantısını sağlayarak MusteriHizmetleriTC'yi parametre alıp getir dedim. dr1 isimli sql veri okuyucuyu komutu okutarak çalıştırdım. Okurken müşteri hizmetleri adı ve müşteri hizmetleri soyadını arada boşluk bırakarak formdaki ad soyad etiketine aktar dedim ve bağlantıyı kapattım.

TabloGuncelle fonksiyonu içinde;

```
DataTable dt = new DataTable();
```

```
SqlDataAdapter da = new SqlDataAdapter("select _Randevular  
da.Fill(dt);  
dataGridView2.DataSource = dt;
```

dt isimli veri tablosunu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta randevu id'yi, randevu tarihi, randevu saati ve şikayeti Randevular tablosundan branş combobox'ın seçili olduğu branş adına ve doktor combobox'ının seçili olduğu doktor ad ve soyadına göre randevu durumu pasif yani alınmamış randevuları sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldurdum.

dedim. Veri kafesi görünümü 2'nin veri kaynağına dt isimli veri tablosunu aktardım.

CmbBrans\_SelectedIndexChanged\_1 içinde;

CmbDoktor.Items.Clear();

```
SqlCommand komut3 = new SqlCommand("SELECT_DoktorAd , Dok  
komut3.Parameters.AddWithValue("@p1", CmbBrans.Text);  
SqlDataReader dr3 = komut3.ExecuteReader();  
while (dr3.Read())  
{  
    CmbDoktor.Items.Add(dr3[0] + " " + dr3[1]);  
}  
bgl.baglanti().Close();
```

komutlarında branş combobox'ı seçildiğinde önce doktor combobox'ını temizledim. Sql komutunda doktor adı ve soyadını Doktorlar tablosundan branş adına göre branş combobox'ını parametre alıp sql bağlantısını sağlayarak getir dedim. dr3 isimli sql veri okuyucu komutu okutarak çalıştır dedim. Okurken doktor combobox'ına tüm doktor ad ve soyadlarını arada boşluk bırakarak ekledim ve sql bağlantısını kapattım.

CmbDoktor\_SelectedIndexChanged içinde;

DataTable dt = new DataTable();

```
SqlDataAdapter da = new SqlDataAdapter("select_Randevu  
da.Fill(dt);  
dataGridView2.DataSource = dt;
```

dt isimli veri tablosunu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta randevu id'yi, randevu tarihi, randevu saati ve şikayeti Randevular tablosundan branş combobox'ın seçili olduğu branş adına ve doktor combobox'ının seçili olduğu doktor ad ve soyadına göre randevu durumu pasif yani alınmamış randevuları sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldurdum. Veri kafesi görünümü 2'nin veri kaynağına dt isimli veri tablosunu aktardım.

BtnRandevuAl\_Click içinde;

```
SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevular_SET_Randevu
    komut.Parameters.AddWithValue("@r2", RchSikayet.Text);
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Randevu_oluşturuldu.", "Bilgi", Messag
    TabloGuncelle();
```

Sql komutta Randevular tablosunda randevu durumu doğru olarak, hasta id'yi HastaTC'ye göre, şikayeti richtextbox'ı parametre olarak randevu id'ye göre sql bağlantısını yapıp güncelledim ve komutu çalıştırdım. Sonra sql bağlantısını kapatıp "Randevu oluşturuldu." şeklinde bilgi mesajı verdirdim. En son yaz... tablosunu güncelledim.

dataGridView2\_CellClick içinde;

```
int secilen = dataGridView2.SelectedCells[0].RowIndex;
    Txtid.Text = dataGridView2.Rows[secilen].Cells[0].Value
```

komutta aktif veya pasif randevular tablosunda randevu id'yi tek tıklayınca ilgili yazı kutusuna aldım.

btnRandevularıGetir\_Click içinde;

```
DataTable dt = new DataTable();
    SqlDataAdapter da = new SqlDataAdapter("select_Randevu
    da.Fill(dt);
    dataGridView2.DataSource = dt;
```

dt isimli veri tablosunu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta randevu id'yi, randevu tarihi, randevu saati ve şikayeti Randevular tablosundan HastaTC'ye göre randevuları sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 2'nin veri kaynağına dt isimli veri tablosunu aktardım.

btnRandevuSil\_Click içinde;

```
SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevular_SET_Randevu
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Randevu iptal edildi.", "Bilgi", Messa
    TabloGuncelle();
```

Randevuyu iptal et butonunda randevular tablosundan randevu durumunu pasif ve tablolar id'lerle birbirine bağlı olduğundan Hastaid'sini null olarak Randevuid'ye göre sql bağlantısını sağlayarak güncelle dedim. Komutu çalıştırdım ve sql bağlantısını kapattım. "Randevu iptal edildi." şeklinde randevu iptal bilgi mesajı gösterdim ve aktif veya pasif randevular tablosunu güncelledim.

button1\_Click içinde;

```
Application.Exit();
```

komutunda uygulamadan çık dedim.

#### **Dosya 11:** Yönetici Detay Formu:

```
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 System.Data.SqlClient;

namespace Yonetim_Hastane
{
    public partial class FrmYoneticiDetay : Form
```



---

---

```

{
    public FrmYoneticiDetay ()
    {
        InitializeComponent ();
    }

    public String YoneticiTC;
    SqlBaglantisi bgl = new SqlBaglantisi ();

    private void FrmYoneticiDetay_Load(object sender, EventArgs e)
    {
        LblTC.Text = YoneticiTC;
        SqlCommand komut = new SqlCommand("select_YoneticiAd , YoneticiTC", bgl.baglanti());
        komut.Parameters.AddWithValue("@pl", YoneticiTC);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())
        {
            LblAdSoyad.Text = dr[0] + "_" + dr[1];
        }
        bgl.baglanti().Close();
    }

    private void LnkBilgiDuzenle_LinkClicked(object sender, LinkLabelEventArgs e)
    {
        FrmYoneticiBilgiDuzenle fr = new FrmYoneticiBilgiDuzenle();
        fr.TCno = LblTC.Text;
        fr.Show();
    }

    private void label1_Click(object sender, EventArgs e)
    {

```

---

---

```

    }

    private void BtnDoktorPaneli_Click(object sender, EventArgs e)
    {
        FrmDoktorPaneli drp = new FrmDoktorPaneli();
        drp.Show();
    }

    private void BtnBransPaneli_Click(object sender, EventArgs e)
    {
        FrmBransPaneli frb = new FrmBransPaneli();
        frb.Show();
    }

    private void BtnRandevuListe_Click(object sender, EventArgs e)
    {
        FrmRandevuListesi rl = new FrmRandevuListesi();
        rl.Show();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        FrmDuyurular frd = new FrmDuyurular();
        frd.Show();
    }

    private void linkYoneticiEkle_LinkClicked(object sender, LinkLabelClickedEventArgs e)
    {
        FrmYoneticiEkle fr = new FrmYoneticiEkle();
        fr.Show();
    }

```

```

    }

    private void linkSekreterEkle_LinkClicked(object sender, Li
    {
        FrmSekreterEkle fr = new FrmSekreterEkle();
        fr.Show();
    }

    private void button2_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

SqlBaglantisi bgl = new SqlBaglantisi();

komutuyla Sql bağlantısını yaptım.

public String YoneticiTC;

komutlarında yöneticinin sisteme girişte girdiği TC'sini almak için diğer formlardan da
ulaşılabilir şekilde YoneticiTC'sini string biçiminde tanımladım.

FrmYoneticiDetay_Load içinde;

LblTC.Text = YoneticiTC;

        SqlCommand komut = new SqlCommand("select_YoneticiAd , Yo
        komut.Parameters.AddWithValue("@p1", YoneticiTC);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())

```

---

---

```
{  
    LblAdSoyad.Text = dr[0] + "_" + dr[1];  
}  
bgl.baglanti().Close();
```

---

---

### **3 PROGRAM ÇALIŞMASININ ANLATIMI ve VERİ TABANI**

---

---

## 4 SONUÇLAR VE ÖNERİLER

---

---

## 5 EKLER

## KAYNAKLAR

- [1] CTAN,[http://zelmanov.ptep-online.com/ctan/lshort\\_turkish.pdf](http://zelmanov.ptep-online.com/ctan/lshort_turkish.pdf) [Ziyaret Tarihi: 4 Kasım 2011]—> Kaynak yazarı bilinmeyen yabancı bir çalışmadan alınmış ise:
- [2] Anonymous, 1989, Farm accountancy data network, an A-Z of methodology, Commission Report of the EC, Brussels, 16-19.
- [3] <http://akgul.bilkent.edu.tr/Yunus/lshort.pdf> —> Kaynak kongreden alınmış ise:
- [4] Calvalho, M. ve Ludermir, T.B., 2007, Particle Swarm Optimization of Neural Network Architectures and Weights, Seventh International Conference on Hybrid Intelligent Systems, Almanya, 336-339. —> Kaynak aktüel dergi ve gazete haberinden alınmış ise:
- [5] Şevkli, M., ve Yenisey, M. M., 2006, Atölye Tipi Çizelgeleme Problemleri için Parçacık Sürü Optimizasyonu Yöntemi, İtÜdergisi/d Mühendislik, Cilt 5, Sayı 2(1), 58-68. —> Kaynak aktüel dergi ve gazete haberinden alınmış ise:
- [6] <http://kisi.deu.edu.tr/umit.akinci/latexseminer.pdf>—> Kaynak yazarı bilinmeyen ulusal bir çalışmadan alınmış ise:
- [7] Anonim, 2006, Tarım istatistikleri özeti, DİE Yayınları, No;12, Ankara, 22-23. —> Kaynak yazarı bilinmeyen ulusal bir çalışmadan alınmış ise:



## **ÖZGEÇMİŞ**

### **KİŞİSEL BİLGİLER**

**Adı Soyadı** :

**Uyruğu** :

**Doğum Yeri ve Tarihi:**

**Adres** :

**Telefon** :

**e-mail** :

### **EĞİTİM DURUMU**

**Lisans Öğrenimi** : BŞEÜ Bilgisayar Mühendisliği Bölümü

**Bitirme Yılı** :

**Lise** :

### **İŞ DENEYİMLERİ**

**Yıl** :

**Kurum** :

**Stajlar** :

### **İLGİ ALANLARI:**

### **YABANCI DİLLER:**

### **BELİRTMEK İSTEDİĞİNİZ DİĞER ÖZELLİKLER:**