



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

Hastane Yönetim ve Randevu Otomasyon Projesi

Eyüp Furkan ARIKAN

PROJE II ÇALIŞMASI

PROJE I DANIŞMANI : Dr.Öğr.Üyesi Salim CEYHAN

BİLECİK
28 Ocak 2019



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

Hastane Yönetim ve Randevu Otomasyon Projesi

Eyüp Furkan ARIKAN

PROJE II ÇALIŞMASI

PROJE I DANIŞMANI : Dr.Öğr.Üyesi Salim CEYHAN

BİLECİK
28 Ocak 2019

ÖZET

Projenin Amacı

Bu projede hastane bazında yönetici tarafından yönetici, branş, doktor ve sekreter kaydını, silmesini ve güncellemesini yapmak, hastanın kendini sisteme kaydetmesini sağlamak, hasta ve yöneticinin bilgilerini düzenlemesini sağlamak, sekreter tarafından pasif yani alınmamış randevu açılmasını sağlamak, güncellemek ve silmek, hasta ve müşteri hizmetleri tarafından hastanın şikayetini yazabilerek randevu almak ve randevu iptal etmek, alınmış veya alınmamış randevuları ilgili koşullara göre ayrı ayrı listelemek, doktor tarafından işi biten randevuları silmek, sekreter tarafından duyuru oluşturmak, doktor, sekreter ve yönetici tarafından duyurulara bakmak ve sekreter tarafından alınmamış randevu oluştururken doktor ve branşlara bakmak amaçlanmıştır.

Projenin Kapsamı

Proje kapsamında kullanılan formlar için C# ve veri tabanı kayıtları için Sql programlama dili kullanılmıştır. Kullanılan programlar olarak ise formlar için Visual Studio 2017 ve veri tabanı kayıtları için MS SQL SERVER 2014 kullanılmıştır. SQL Server Express veri tabanı kayıtları için, Sql Server Managemet Studio veri tabanı kayıtları düzenlemek için kullanılmıştır. Veri tabanından veri çekme, veri ekleme, silme ve güncelleme, çoklu yani birden fazla tablodan veri çekme, veri ekleme, silme ve güncelleme tablo işlemleri, formlardaki tabloları ekleme, silme ve güncelleme işlemlerinden sonra otomatik olarak güncellemek, bütün ekleme, silme güncelleme, randevu alma ve randevuyu iptal etme gibi işlemlerinden sonra kullanıcıyı bilgilendirmek , tablolara ilgili verileri çağırarak, tablolardaki verileri bir kez tıklamak koşuluyla ilgili alanlara aktarmak, ilgili butona tıklandığında ilgili formlar arası geçiş yapmak, gözükmemesi gereken formları görünmez yapmak, işlemler bittiğinde programı sonlandırmak ve formlarda şekil, biçim, renk, büyüklük, yazı biçimi, yazı büyüklüğü, yazı rengi, formlardaki yerleşim, nesnelerin ve yazının neyin üzerinde hangi koşulla duracağı gibi görünümünü tasarlama işlemleri yapmak

programın kapsamı içindedir.

Sonuçlar

Bu proje çalışmasında, hastane bazında yönetici tarafından yönetici, branş, doktor ve sekreter kaydını kaydetmesi, silmesi ve güncellemesi yapılması, hastanın kendini sisteme kaydetmesini sağlaması, hasta ve yöneticinin bilgilerini düzenlemesini sağlaması, sekreter tarafından pasif yani alınmamış randevu açılmasını sağlaması, güncellemesi ve silmesi, hasta ve müşteri hizmetleri tarafından hastanın şikayetini yazabilerek randevu alınması ve randevu iptal etmesi, alınmış veya alınmamış randevuları ilgili koşullara göre ayrı ayrı listelemesi, doktor tarafından işi biten randevuları silmesi, sekreter tarafından duyuru oluşturması, doktor, sekreter ve yönetici tarafından duyurulara bakması ve sekreter tarafından alınmamış randevu oluştururken doktor ve branşlara bakması gerçekleştirilmiştir.

ABSTRACT

Project Objective

In this project, the manager, branch, doctor and secretary records, deletes and updates by the administrator on the basis of the hospital, to enable the patient to register himself / herself in the system, to arrange the information of the patient and the manager, to enable the secretary to open, update and delete the appointment so that the appointment has not been received by the secretary, patient and customer service to make an appointment by writing the complaint of the patient and to cancel the appointment, to list the appointments which are received or not taken according to the relevant conditions, to delete the appointments by the doctor, to make an announcement by the secretary, to look for the announcements by the doctor, secretary and manager and to create an appointment not taken by the secretary and branches are intended to look at.

Scope of Project

For the forms used within the scope of the project, Sql programming language is used for C# and database records. Visual Studio 2017 for forms and MS SQL SERVER 2014 for database records were used as the programs used. For SQL Server Express database records, Sql Server was used to edit the Managemet Studio database records. Data extraction from the database, data adding, deleting and updating, multi-ie multiple data from the table, data insertion, deletion and update table operations, add, delete and update tables automatically after the process of adding all the delete, update, appointment and informing the user after the procedures such as canceling the appointment, calling the related data to the tables, transferring the data to the related fields by clicking on the data in the tables, clicking the relevant button, switching between the related forms, making the forms not invisible, terminating the program when the processes are finished and color, size, typeface, font size, text color, form placement, objects and the condition on which to stand out what conditions to design the appearance of the program is within the scope of the program.

Results

In this project study, the manager, branch, doctor and secretary record, record and update the manager, branch, doctor and secretary record, provide the patient and the administrator to edit his / her information, provide the secretary by the secretary to open, update and delete the appointment. and the customer service to write the patient's complaint and cancel an appointment, list appointments that are received or not received separately according to the relevant conditions, delete appointments by the doctor, make an announcement by the secretary, look for announcements by the doctor, secretary and administrator and make an appointment not received by the secretary. to look at the doctors and branches.

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

28 Ocak 2019

İÇİNDEKİLER

ÖZET	ii
ABSTRACT	iv
TEŞEKKÜR	vi
SİMGE LİSTESİ	viii
ŞEKİL LİSTESİ	ix
TABLO LİSTESİ	x
1 GİRİŞ	1
2 PROJE C# KOD ANLATIMI	2
3 PROGRAM VERİ TABANI (SQL) ÇALIŞMASININ ANLATIMI	124
4 SONUÇLAR VE ÖNERİLER	132
5 EKLER	134
KAYNAKLAR	135
ÖZGEÇMİŞ	136

SİMGE LİSTESİ

ŞEKİL LİSTESİ

TABLO LİSTESİ

1 GİRİŞ

Bu proje çalışmasında, hastane bazında yönetici tarafından yönetici, branş, doktor ve sekreter kaydını kaydetmesi, silmesi ve güncellemesi yapılması, hastanın kendini sisteme kaydetmesini sağlaması, hasta ve yöneticinin bilgilerini düzenlemesini sağlaması, sekreter tarafından pasif yani alınmamış randevu açılmasını sağlaması, güncellemesi ve silmesi, hasta ve müşteri hizmetleri tarafından hastanın şikayetini yazabilerek randevu alması ve randevu iptal etmesi, alınmış veya alınmamış randevuları ilgili koşullara göre ayrı ayrı listelemesi, doktor tarafından işi biten randevuları silmesi, sekreter tarafından duyuru oluşturmaları, doktor, sekreter ve yönetici tarafından duyurulara bakması ve sekreter tarafından alınmamış randevu oluştururken doktor ve branşlara bakması gerçekleştirilmiştir.

Proje kapsamında kullanılan formlar için C# ve veri tabanı kayıtları için Sql programlama dili kullanılmıştır. Kullanılan programlar olarak ise formlar için Visual Studio 2017 ve veri tabanı kayıtları için MS SQL SERVER 2014 kullanılmıştır. SQL Server Express veri tabanı kayıtları için, Sql Server Managemet Studio veri tabanı kayıtları düzenlemek için kullanılmıştır.

Veri tabanından veri çekme, veri ekleme, silme ve güncelleme, çoklu yani birden fazla tablodan veri çekme, veri ekleme, silme ve güncelleme tablo işlemleri, formlardaki tabloları ekleme, silme ve güncelleme işlemlerinden sonra otomatik olarak güncellemek, bütün ekleme, silme güncelleme, randevu alma ve randevuyu iptal etme gibi işlemlerinden sonra kullanıcıyı bilgilendirmek , tablolara ilgili verileri çağırmak, tablolardaki verileri bir kez tıklamak koşuluyla ilgili alanlara aktarmak, ilgili butona tıklandığında ilgili formlar arası geçiş yapmak, gözükmemesi gereken formları görünmez yapmak, işlemler bittiğinde programı sonlandırmak ve formlarda şekil, biçim, renk, büyüklük, yazı biçimi, yazı büyüklüğü, yazı rengi, formlardaki yerleşim, nesnelerin ve yazının neyin üzerinde hangi koşulla duracağı gibi görünümü tasarlama işlemleri yapmak programın çalışmasının içindedir.

Bu proje bir hastane sisteminde yönetim işlemleri ve randevu alma veya iptal etme işlemleri için kullanılabilir.

2 PROJE C# KOD ANLATIMI

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_Tbl_Hastalar_where_HastaTC=@p1_and
                    _____HastaSifre=@p2", bgl.baglanti());

```

```

        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_hatalıdır.", "Uyarı",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        }
        bgl.baglanti().Close();
    } else if( secim.Equals("Doktor") )
    {
        SqlCommand komut = new SqlCommand("select_*
        .....from_Tbl_Doktorlar_where_DoktorTC=@d1_and
        .....DoktorSifre=@d2", bgl.baglanti());
        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_hatalıdır.", "Uyarı",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        }
        bgl.baglanti().Close();
    } else if( secim.Equals("Sekreter") )
    {
        SqlCommand komut = new SqlCommand("select_*
        .....from_Tbl_Sekreterler_where_SekreterTC=@p1
        .....and_SekreterSifre=@p2", bgl.baglanti());
        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.", "Uyarı",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
    bgl.baglanti().Close();
} else if( secim.Equals("Yönetici") )
{
    SqlCommand komut = new SqlCommand("select_*
    from_Tbl_Yoneticiler_where_YoneticiciTC=@p1
    and_YoneticiciSifre=@p2", bgl.baglanti());
    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

```

```

.....şifre_hatalıdır.", "Uyarı",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
    bgl.baglanti().Close();
} else if( secim.Equals("Müşteri_Hizmetleri") )
{
    SqlCommand komut = new SqlCommand("select_*
.....from_Tbl_Musteri_Hizmetleri_where
.....MusteriHizmetleriTC=@p1_and
.....MusteriHizmetleriSifre=@p2",
        bgl.baglanti());
    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_hatalıdır.", "Uyarı",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
}

```

```

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

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

```

Yukarıdaki Giriş Formu Kodları'nda;

```
SqlBaglantisi bg1 = new SqlBaglantisi();
```

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_Tbl_Hastalar_where_HastaTC=@p1_and
        _____HastaSifre=@p2", bg1.baglanti());
        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_hatalıdır.", "Uyarı",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        }
        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_Tbl_Doktorlar_where_DoktorTC=@d1_and
    _____DoktorSifre=@d2", bgl.baglanti());

```

```

        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_hatalıdır.", "Uyarı",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        }
        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_Tbl_Sekreterler_where_SekreterTC=@p1
        _____and_SekreterSifre=@p2", bgl.baglanti());
        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.", "Uyarı",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        }
        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_Tbl_Yoneticiler_where_YoneticiciTC=@p1
    _____and_YoneticiciSifre=@p2", bgl.baglanti());
    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_şifre
        _____hatalıdır.", "Uyarı",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
    bgl.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 Yone-

ticiSifre için kullandım. Sonra Sql veri okuyucuyu komuta göre çalıştırdım.

Eğer veri okunduysa önce YoneticiTC'yi açılan forma gönderdim. FrmYoneticiDetay 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_Musteri_Hizmetleri_where
    MusteriHizmetleriTC=@p1_and
    MusteriHizmetleriSifre=@p2",
    bgl.baglanti());
    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_hatalıdır.", "Uyarı",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
}
```



```
}  
    bg1 . 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_Hastalar (HastaAd ,HastaSoyad ,HastaTC ,
HastaTelefon ,HastaSifre ,HastaCinsiyet )
values (@p1 ,@p2 ,@p3 ,@p4 ,@p5 ,@p6) " ,
bgl.baglanti());
                    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:_" + TxtSifre.Text, "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
    }
}

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 kimlik numaranın ilk hanesi 0
        değilse
        {
            for (int i = 0; i < 10; i++)
            {
                toplam = toplam +
                Convert.ToInt32(tcNo[i].
                ToString());
                if (i % 2 == 0)
                {
                    if (i != 10)
                    {
                        toplam2 = toplam2 +

```

```
Convert.ToInt32(tcNo[i].
ToString()); // 7 ile
çarpılacak sayıları
topluyoruz
    }

}
else
{
    if (i != 9)
    {
        toplam3 = toplam3 +
            Convert.ToInt32(tcNo[i].
                ToString());
    }
}
}
else
{
    MessageBox.Show("Tc_Kimlik_Numaranızın
    .....ilk_hanesi_0_olamaz.");
    return false;
}
}
else
{
    MessageBox.Show("Tc_Kimlik_Numarınız_11
    .....haneli_olmak_zorunda.Eksik_ya_da_fazla
    .....değer_girdiniz.");
    return false;
}
```

```
int toplam3 = 0;
if (tcNo.Length == 11)
{
    if (Convert.ToInt32(tcNo[0].ToString()) !=
0) //tc kimlik numaranın ilk hanesi 0
değilse
    {
        for (int i = 0; i < 10; i++)
        {
            toplam = toplam +
                Convert.ToInt32(tcNo[i].
                    ToString());
            if (i % 2 == 0)
            {
                if (i != 10)
                {
                    toplam2 = toplam2 +
                        Convert.ToInt32(tcNo[i].
                            ToString()); // 7 ile
                        çarpılacak sayıları
                        topluyoruz
                }
            }
        }
        else
        {
            if (i != 9)
            {
                toplam3 = toplam3 +
                    Convert.ToInt32(tcNo[i].
                        ToString());
            }
        }
    }
}
```

```

    }

    }

    }

    else
    {
        MessageBox.Show("Tc_Kimlik_Numaranızın
        .....ilk_hanesi_0_olamaz.");
        return false;
    }
}

else
{
    MessageBox.Show("Tc_Kimlik_Numarınız_11
    .....haneli_olmak_zorunda.Eksik_ya_da_fazla
    .....değer_girdiniz.");
    return false;

}

if (((toplamlar2 * 7) - toplam3) % 10 ==
    Convert.ToInt32(tcNo[9].ToString()) &&
    toplam % 10 == Convert.ToInt32(tcNo[10]
    .ToString()))
{
    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 tamsayıya 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_Hastalar ( HastaAd , HastaSoyad , HastaTC ,  
    _____HastaTelefon , HastaSifre , HastaCinsiyet )  
    _____values ( @p1 , @p2 , @p3 , @p4 , @p5 , @p6 ) " ,  
    _____bgl.baglanti ( ) );
```



```

        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:_" + TxtSifre.Text, "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
    }

```

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 _from_ Tbl_Hastalar _where
                _____HastaTC=@p1", 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_Tbl_Randevular_where_Hastaid="+tc ,
    bgl.baglanti());
    da.Fill(dt);
    dataGridView1.DataSource = dt; */

    //Branş Çekme

    SqlCommand komut2 = new SqlCommand("select
    .....BransAd_from_Tbl_Branslar",bgl.baglanti());
    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_Randevular.Randevuid ,_Tbl_Doktorlar .
    .....DoktorAd ,_Tbl_Doktorlar.DoktorSoyad ,
    .....Tbl_Randevular.RandevuTarih ,_Tbl_Randevular .
    .....RandevuSaat ,_Tbl_Randevular.RandevuDurum ,
    .....Tbl_Randevular.Sikayet_FROM_Tbl_Randevular ,
    .....Tbl_Doktorlar_WHERE_Hastaid_IN_( select

```

```

        Hastaid_from_Tbl_Hastalar_where_HastaTC_=
        ' " + tc + " ' _and_RandevuDurum='True ' ) _AND
        Tbl_Randevular.Doktorid_=
        Tbl_Doktorlar.Doktorid" , bgl.baglanti ());
        da2.Fill(dt2);
        dataGridView1.DataSource = dt2;
    }

    private void CmbBrans_SelectedIndexChanged( object
sender , EventArgs e)
    {
        CmbDoktor.Items.Clear();
        SqlCommand komut3 = new SqlCommand("SELECT
        DoktorAd ,DoktorSoyad_FROM_Tbl_Doktorlar
        WHERE_Bransid_IN_( select_Bransid_from
        tbl_Branslar_where_BransAd_LIKE_@p1)" ,
        bgl.baglanti ());
        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 , EventArgs e)
    {
        DataTable dt = new DataTable ();
    }

```

```

        SqlDataAdapter da = new SqlDataAdapter("select
        _____Randevid ,RandevuTarih ,RandevuSaat_from
        _____Tbl_Randevular_where_Bransid_IN_(select
        _____Bransid_FROM_Tbl_Branslar_WHERE_BransAd_LIKE
        _____'" + CmbBrans.Text + "')_and_Doktorid_IN
        _____(select_Doktorid_FROM_Tbl_Doktorlar_WHERE
        _____DoktorAd_+_ ' ' _+_DoktorSoyad_LIKE_' " +
        CmbDoktor.Text + "'_and_RandevuDurum='False ')
        _____", bgl.baglanti());
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

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

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

    public string RandevuTarihi , RandevuSaati;

```

```

        private void dataGridView2_CellContentClick(object
sender , DataGridViewCellEventArgs e)
        {

        }

        private void button1_Click(object sender , EventArgs
e)
        {
            SqlCommand komut = new SqlCommand("UPDATE
            Tbl_Randevular_SET_RandevuDurum=' False ',
            Hastaid=null_WHERE_Randevuid='" + Txtid.Text +
            "'_", bgl.baglanti());
            komut.ExecuteNonQuery();
            bgl.baglanti().Close();
            MessageBox.Show("Randevu iptal edildi.",
            "Bilgi", MessageBoxButtons.OK,
            MessageBoxIcon.Information);

            TabloGuncelle();

        }

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

```

```

    }

    private void TabloGuncelle()
    {
        //Tablo Güncelleme
        DataTable dt2 = new DataTable();
        SqlDataAdapter da2 = new SqlDataAdapter("SELECT
        Tbl_Randevular.Randevuid , Tbl_Doktorlar .
        DoktorAd , Tbl_Doktorlar . DoktorSoyad ,
        Tbl_Randevular . RandevuTarih , Tbl_Randevular .
        RandevuSaat , Tbl_Randevular . RandevuDurum ,
        Tbl_Randevular . Sikayet FROM Tbl_Randevular ,
        Tbl_Doktorlar WHERE Hastaid IN ( select
        Hastaid from Tbl_Hastalar where HastaTC =
        '" + tc + "' and RandevuDurum='True ') AND
        Tbl_Randevular . Doktorid = Tbl_Doktorlar .
        Doktorid", bgl.baglanti());
        da2.Fill(dt2);
        dataGridView1.DataSource = dt2;
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select
        Randevuid , RandevuTarih , RandevuSaat from
        Tbl_Randevular where Bransid IN ( select
        Bransid FROM Tbl_Branslar WHERE BransAd LIKE
        '" + CmbBrans.Text + "' ) and Doktorid IN
        ( select Doktorid FROM Tbl_Doktorlar WHERE
        DoktorAd + ' ' + DoktorSoyad LIKE '" +
        CmbDoktor.Text + "' and RandevuDurum='False ')
        ", bgl.baglanti());
        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_Randevular SET_RandevuDurum='True ',
        Hastaid=(SELECT_Hastaid FROM_Tbl_Hastalar
        WHERE_HastaTC_=' " + tc + " '),Sikayet=@r2
        WHERE_Randevuid=' " + Txtid.Text + " ',
        bgl.baglanti());
        komut.Parameters.AddWithValue("@r2",
        RchSikayet.Text);
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_oluřturuldu.",
        "Bilgi", MessageBoxButtons.OK,
        MessageBoxIcon.Information);

        TabloGuncelle();

        /*SqlCommand komut = new SqlCommand("insert
        into_Tbl_Randevular

```



```

(RandevuDurum , Hastaid , Sikayet , Doktorid ,
Bransid ) _SELECT_1 ,(SELECT_ Hastaid _FROM
Tbl_Hastalar _WHERE_ HastaTC_ = ' " + tc + " ' ),
@r2 , _ (SELECT_ Doktorid _from_ Tbl_Doktorlar
WHERE_ DoktorAd + ' _ ' + DoktorSoyad_ = ' " +
CmbDoktor . Text + " ' ) , (SELECT_ Bransid _from
Tbl_Branslar _WHERE_ BransAd_ = ' " +
CmbBrans . Text + " ' ) " , bgl . baglanti ());
komut . Parameters . AddWithValue ( " @r2 " ,
RchSikayet . Text );
komut . Parameters . AddWithValue ( " @r3 " ,
Txtid . Text );
komut . ExecuteNonQuery ();
bgl . baglanti ( ) . Close ();

// dataGridView2_CellClick özelliği çalışmadığı
için Randevuid gelmiyor ve aşağıdaki kodda
veri gelirken hata veriyor .
SqlCommand komut2 = new SqlCommand ( " select
RandevuTarih , RandevuSaat _from_ Tbl_Randevular
where_ Randevuid = @r4 " , bgl . baglanti ());
komut2 . Parameters . AddWithValue ( " @r4 " ,
Txtid . Text );
SqlDataReader dr = komut2 . ExecuteReader ();
RandevuTarihi = dr [1] . ToString ();
RandevuSaati = dr [2] . ToString ();
MessageBox . Show (CmbBrans . Text + " _bölümünden_ "
+ CmbDoktor . Text + " _doktordan_ \n " +
RandevuTarihi + " _tarihinde_ " + RandevuSaati +
" _ve_ saatinde_ randevunuz_ alınmıştır . " ,
" Bilgi " , MessageBoxButtons . OK,

```

```

        MessageBoxIcon.Information);
        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 ,HastaSoyad_from_Tbl_Hastalar_where
_____HastaTC=@p1", bgl.baglanti());
```

```
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_Branslar", bgl.baglanti());  
    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_Randevular.Randevuid, _Tbl_Doktorlar.  
_____DoktorAd, _Tbl_Doktorlar.DoktorSoyad,  
_____Tbl_Randevular.RandevuTarih, _Tbl_Randevular.  
_____RandevuSaat, _Tbl_Randevular.RandevuDurum,  
_____Tbl_Randevular.Sikayet FROM _Tbl_Randevular,  
_____Tbl_Doktorlar WHERE _Hastaid_IN( select  
_____Hastaid_from _Tbl_Hastalar_where _HastaTC_=  
_____'" + tc + "'_and_RandevuDurum='True ' )_AND  
_____Tbl_Randevular.Doktorid_=  
_____Tbl_Doktorlar.Doktorid", bgl.baglanti());  
    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, ran-

devu 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 , DoktorSoyad FROM Tbl_Doktorlar  
____WHERE Bransid IN (select Bransid from  
____tbl_Branslar where BransAd LIKE @p1) ",  
bgl.baglanti());  
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  
____Randevuid , RandevuTarih , RandevuSaat from  
____Tbl_Randevular where Bransid IN (select  
____Bransid FROM Tbl_Branslar WHERE BransAd LIKE
```

```

.....'" + CmbBrans.Text + "'')_and_Doktorid_IN
.....( select_Doktorid_FROM_Tbl_Doktorlar_WHERE
.....DoktorAd+_+'_'+DoktorSoyad_LIKE_' " +
.....CmbDoktor.Text + "'_and_RandevuDurum=' False ')
....." , bgl.baglanti ());
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 doldurdum. dt isimli veri tablosunu da veri kafesi görünümü 2'nin veri kaynağına aktardım.

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.ToString ();

```

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_Randevular.Randevuid, _____Tbl_Doktorlar.
    _____DoktorAd, _____Tbl_Doktorlar.DoktorSoyad,
    _____Tbl_Randevular.RandevuTarih, _____Tbl_Randevular.
    _____RandevuSaat, Tbl_Randevular.RandevuDurum,
    _____Tbl_Randevular.Sikayet _____FROM _____Tbl_Randevular,
    _____Tbl_Doktorlar _____WHERE _____Hastaid _____IN _____(select _____Hastaid
    _____from _____Tbl_Hastalar _____where _____HastaTC _____=
    _____'" + tc + "' _____and _____RandevuDurum='True') _____AND
    _____Tbl_Randevular.Doktorid _____=
    _____Tbl_Doktorlar.Doktorid", bgl.baglanti());
    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
    _____Randevuid, RandevuTarih, RandevuSaat _____from
    _____Tbl_Randevular _____where _____Bransid _____IN _____(select
    _____Bransid _____FROM _____Tbl_Branslar _____WHERE _____BransAd _____LIKE
    _____'" + CmbBrans.Text + "' _____) _____and _____Doktorid _____IN

```

```

.....( select _Doktorid _FROM _Tbl_Doktorlar _WHERE
.....DoktorAd _+ _' _' _+ _DoktorSoyad _LIKE _' " +
.....CmbDoktor .Text + " _and _RandevuDurum=' False ' )
....." , bgl . baglanti ());
.....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 brans 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 _RandevuDurum=' False ' , _Hastaid=null _WHERE _Randevuid=' "
+ Txtid .Text + " _' _" , bgl . baglanti ());
.....komut . ExecuteNonQuery ( );
.....bgl . baglanti ( ) . Close ( );
.....MessageBox . Show( "Randevu _iptal _edildi . " ,
....." Bilgi " , MessageBoxButtons . OK ,
.....MessageBoxIcon . Information );
.....
.....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.ToString();
```

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_RandevuDurum='True',Hastaid=(SELECT_Hastaid_FROM  
Tbl_Hastalar_WHERE_HastaTC_=' " + tc + " '),Sikayet=@r2  
WHERE_Randevuid=' " + Txtid.Text + "'_", bgl.baglanti());  
    komut.Parameters.AddWithValue("@r2",  
        RchSikayet.Text);  
    komut.ExecuteNonQuery();  
    bgl.baglanti().Close();  
    MessageBox.Show("Randevu_oluşturuldu.",  
        "Bilgi", MessageBoxButtons.OK,  
        MessageBoxIcon.Information);  
  
    TabloGuncelle();
```

Randevu al butonunda randevular tablosundan randevu durumunu aktif ve HastaTC'sine göre Hastaid'sini ve şikayeti Randevuid'sine göre sql bağlantısını sağlayarak güncelle-

dim. 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_Hastalar_where_HastaTC=@p1",
        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_Hastalar_set_HastaAd=@p1,
            HastaSoyad=@p2,_HastaTelefon=@p3,
            HastaSifre=@p4,_HastaCinsiyet=@p5_where
            HastaTC=@p6", 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 . Parameters . AddWithValue ( "@p5" ,
        CmbCinsiyet . Text );
        komut2 . Parameters . AddWithValue ( "@p6" ,
        MskTC . Text );
        komut2 . ExecuteNonQuery ();
        bgl . baglanti (). Close ();
        MessageBox . Show ( " Bilgileriniz _ güncellendi . "
        , " Bilgi " , MessageBoxButtons . OK ,
        MessageBoxIcon . Information );
    }
}
}
}
}

```

using System . Data . SqlConnection ;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

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

komutuyla Sql bağlantısını yaptım.

```
public string TCno;
```

komutuyla her yerden ulaşılacak TCno'yu tanımladım.

FrmBilgiDuzenle_Load içinde;

```
MskTC . Text = TCno;
```

```

        SqlCommand komut = new SqlCommand ( " select _ *
        from _ Tbl _ Hastalar _ where _ HastaTC = @p1 " ,

```

```

        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();

```

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_Hastalar_set_HastaAd=@p1,
        _____HastaSoyad=@p2, _HastaTelefon=@p3,
        _____HastaSifre=@p4, _HastaCinsiyet=@p5_where
        _____HastaTC=@p6", 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 . Parameters . AddWithValue ( "@p5" ,
        CmbCinsiyet . Text );
        komut2 . Parameters . AddWithValue ( "@p6" ,
        MskTC . Text );
        komut2 . ExecuteNonQuery ();
        bgl . baglanti (). Close ();
        MessageBox . Show ( " Bilgileriniz _ güncellendi . "
        , " Bilgi " , MessageBoxButtons . OK ,
        MessageBoxIcon . Information );
    }

```

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 alarak 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 ,Doktorid_from
                _____Tbl_Doktorlar_where_DoktorTC=@d1",
                bgl.baglanti());
                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_Randevular.Randevuid ,
        _____ Tbl_Hastalar.HastaAd ,
        _____ Tbl_Hastalar.HastaSoyad ,
        _____ Tbl_Randevular.RandevuTarih ,
        _____ Tbl_Randevular.RandevuSaat ,
        _____ Tbl_Randevular.RandevuDurum ,
        _____ Tbl_Randevular.Sikayet_from
        _____ Tbl_Randevular , _Tbl_Hastalar_where
        _____ Doktorid='" + dr[2].ToString() + "'
        _____ and_RandevuDurum='True' _AND
        _____ Tbl_Randevular.Hastaid_=
        _____ Tbl_Hastalar.Hastaid", bgl.baglanti());
        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,
DataGridViewCellEventArgs e)
{
    int secilen = dataGridView1.SelectedCells[0].
RowIndex;
    RchSikayet.Text = dataGridView1.Rows[secilen].
Cells[4].Value.ToString();
    txtRandevuid.Text = dataGridView1.
Rows[secilen].Cells[0].Value.ToString();
}

private void dataGridView1_CellContentClick(object
sender, DataGridViewCellEventArgs e)
{
}

private void btnTamamla_Click(object sender,
EventArgs e)
```



```

        {
            SqlCommand komut2 = new SqlCommand("delete_from
            _____Tbl_Randevular_where_Randevuid=@p1",
            bgl.baglanti());
            komut2.Parameters.AddWithValue("@p1",
            txtRandevuid.Text);
            komut2.ExecuteNonQuery();
            bgl.baglanti().Close();
            MessageBox.Show("Randevu_Tamamlandı.", "Bilgi",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
            TabloGuncelle();
        }

        private void TabloGuncelle()
        {
            LblTC.Text = DoktorTCsi;
            //Doktor Ad Soyad Çekme
            SqlCommand komut = new SqlCommand("select
            _____DoktorAd ,DoktorSoyad ,Doktorid_from
            _____Tbl_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 + " " +

```

FrmDoktorDetay_Load içinde;

```
LblTC.Text = DoktorTCsi;

//Doktor Ad Soyad Çekme
SqlCommand komut = new SqlCommand("select
_____DoktorAd ,DoktorSoyad ,Doktorid_from
_____Tbl_Doktorlar_where_DoktorTC=@d1" ,
    bgl.baglanti());
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_Randevular.Randevuid ,
_____Tbl_Hastalar.HastaAd ,
_____Tbl_Hastalar.HastaSoyad ,
_____Tbl_Randevular.RandevuTarih ,
```

```

            Tbl_Randevular.RandevuSaat ,
            Tbl_Randevular.RandevuDurum ,
            Tbl_Randevular.Sikayet_from
            Tbl_Randevular , Tbl_Hastalar_where
            Doktorid=' ' + dr[2].ToString() + " '
            and_RandevuDurum=' True ' _AND
            Tbl_Randevular.Hastaid_=
            Tbl_Hastalar.Hastaid" , bgl.baglanti());
            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].Value.ToString();
txtRandevuid.Text = dataGridView1.
Rows[secilen].Cells[0].Value.ToString();

```

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 ,Doktorid_from
_____Tbl_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_ Tbl_Randevular.Randevuid ,
_____Tbl_Hastalar.HastaAd ,
_____Tbl_Hastalar.HastaSoyad ,

```

```

            Tbl_Randevular.RandevuTarih ,
            Tbl_Randevular.RandevuSaat ,
            Tbl_Randevular.RandevuDurum ,
            Tbl_Randevular.Sikayet_from
            Tbl_Randevular , Tbl_Hastalar_where
            Doktorid=' ' + dr[2].ToString() + ' '
            and_RandevuDurum='True' _AND
            Tbl_Randevular.Hastaid_=
            Tbl_Hastalar.Hastaid", 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 olarak 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_Randevuid=@p1", bgl.baglanti());
komut2.Parameters.AddWithValue("@p1",

```

```

txtRandevuid . Text );
komut2 . ExecuteNonQuery ();
bgl . baglanti (). Close ();
MessageBox . Show ( " Randevu _ Tamamlandı . " , " Bilgi " ,
    MessageBoxButtons . OK ,
    MessageBoxIcon . Information );
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 ,Duyuru_from_Tbl_Duyurular",
        bgl.baglanti());
        da.Fill(dt);
        dataGridView1.DataSource = dt;
    }
}

```

using System.Data.SqlClient;

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

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

komutuyla Sql bağlantısını yaptım.

FrmDuyurular_Load içinde;

```

DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select
        _____Duyuruid ,Duyuru_from_Tbl_Duyurular",
        bgl.baglanti());
        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 , SekreterSoyad_from_Tbl_Sekreterler
    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_Randevular.Randevuid ,
    Tbl_Doktorlar.DoktorAd ,
    Tbl_Doktorlar.DoktorSoyad ,
    Tbl_Branslar.BransAd ,
    Tbl_Randevular.RandevuTarih ,
    Tbl_Randevular.RandevuSaat ,
    Tbl_Randevular.RandevuDurum ,
    Tbl_Randevular.Sikayet_from_Tbl_Randevular ,
    Tbl_Doktorlar , Tbl_Branslar WHERE
    Tbl_Randevular.Doktorid = Tbl_Doktorlar .
    Doktorid AND Tbl_Randevular.Bransid =

```

```

        Tbl_Branslar.Bransid", _bgl.baglanti());
        da.Fill(dt);
        dataGridView1.DataSource = dt;

        // Branşı ComboBox'a aktarma
        SqlCommand komut2 = new SqlCommand("select
        BransAd_from_Tbl_Branslar", 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_Randevular.Randevuid,
        Tbl_Doktorlar.DoktorAd,
        Tbl_Doktorlar.DoktorSoyad,
        Tbl_Branslar.BransAd,
        Tbl_Randevular.RandevuTarih,
        Tbl_Randevular.RandevuSaat,
        Tbl_Randevular.RandevuDurum,
        Tbl_Randevular.Sikayet_from_Tbl_Randevular,
        Tbl_Doktorlar, Tbl_Branslar WHERE
        Tbl_Randevular.Doktorid = Tbl_Doktorlar
        .Doktorid AND Tbl_Randevular.Bransid =
        Tbl_Branslar.Bransid", bgl.baglanti());
    }

```

```

        da.Fill(dt);
        dataGridView1.DataSource = dt;
    }

    private void BtnKaydet_Click(object sender,
    EventArgs e)
    {
        SqlCommand komutkaydet = new SqlCommand("insert
        into
        Tbl_Randevular(RandevuTarih,RandevuSaat,
        Bransid,Doktorid,RandevuDurum) values (@p1,@p2,
        (SELECT Bransid FROM Tbl_Branslar WHERE
        BransAd='"+ CmbBrans.Text + " '), (SELECT
        Doktorid FROM Tbl_Doktorlar WHERE DoktorAd+'
        '+DoktorSoyad='" + CmbDoktor.Text + " '),
        'False ')", bgl.baglanti());
        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_oluřturuldu.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloyuGuncelle();
    }

```

```

private void CmbBrans_SelectedIndexChanged(object sender, EventArgs e)
{
    CmbDoktor.Items.Clear();
    SqlCommand komut = new SqlCommand("select
    _____DoktorAd ,DoktorSoyad_ from _Tbl_Doktorlar_ where
    _____Bransid_=(SELECT_ Bransid_ FROM_ Tbl_Branslar
    _____WHERE_ BransAd='"+ CmbBrans.Text + " ')",
    bgl.baglanti());
    SqlDataReader dr = komut.ExecuteReader();
    while (dr.Read())
    {
        CmbDoktor.Items.Add(dr[0] + "_" + dr[1]);
    }
    bgl.baglanti().Close();
}

private void BtnDuyuruOlustur_Click(object sender,
EventArgs e)
{
    SqlCommand komut = new SqlCommand("insert_into
    _____Tbl_Duyurular(Duyuru)_values(@d1)",
    bgl.baglanti());
    komut.Parameters.AddWithValue("@d1",
    RchDuyuru.Text);
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Duyuru_oluřturuldu.", "Bilgi",
    MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}

```

```
    }

    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_Randevular set RandevuTarih=@p1,
            RandevuSaat=@p2, Bransid=(SELECT Bransid FROM
            Tbl_Branslar WHERE BransAd=@p3), Doktorid=
            (SELECT Doktorid FROM Tbl_Doktorlar WHERE
            DoktorAd=@p4 and DoktorSoyad=@p5) where
            Randevuid=@p6", bgl.baglanti());

            SqlCommand komut2 = new SqlCommand("update
            Tbl_Randevular set RandevuTarih=@p1,
            RandevuSaat=@p2 where Randevuid=@p6",
            bgl.baglanti());
            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", MessageBoxButtons.OK,
            MessageBoxIcon.Information);

```

```
        TabloyuGuncelle ();
    }

    String DoktorAd , DoktorSoyad ;
    private void dataGridView1_CellClick(object sender ,
        DataGridViewCellEventArgs e)
    {
        int secilen = dataGridView1.SelectedCells[0].
        RowIndex ;
        Txtid.Text = dataGridView1.Rows[secilen].
        Cells[0].Value.ToString ();
        MskTarih.Text = dataGridView1.Rows[secilen].
        Cells[4].Value.ToString ();
        MskSaat.Text = dataGridView1.Rows[secilen].
        Cells[5].Value.ToString ();
        CmbBrans.Text = dataGridView1.Rows[secilen].
        Cells[3].Value.ToString ();
        CmbDoktor.Text = dataGridView1.Rows[secilen].
        Cells[1].Value.ToString () + "□" +
            dataGridView1.Rows[secilen].
        Cells[2].Value.ToString ();
        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_Randevular_WHERE_Randevuid='" + Txtid.Text
        + "'", bgl.baglanti());
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_silindi.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        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 , SekreterSoyad_from_Tbl_Sekreterler
_____ 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();
```

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_Randevular.Randevuid ,_Tbl_Doktorlar.
_____ DoktorAd ,_Tbl_Doktorlar.DoktorSoyad ,
```

```

.....Tbl_Branslar.BransAd, _Tbl_Randevular.
.....RandevuTarih, _Tbl_Randevular.RandevuSaat,
.....Tbl_Randevular.RandevuDurum, _Tbl_Randevular.
.....Sikayet_from _Tbl_Randevular, _Tbl_Doktorlar,
.....Tbl_Branslar WHERE _Tbl_Randevular.Doktorid_=
.....Tbl_Doktorlar.Doktorid AND _Tbl_Randevular.
.....Bransid=_Tbl_Branslar.Bransid",
.....bgl.baglanti());
.....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.

```

// Branş1 Combobox'a aktarma
.....SqlCommand_komut2=_new SqlCommand(" select
.....BransAd_from _Tbl_Branslar", bgl.baglanti());
.....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.Randevuid, _____Tbl_Doktorlar.
        _____DoktorAd, _____Tbl_Doktorlar.DoktorSoyad,
        _____Tbl_Branslar.BransAd, _____Tbl_Randevular.
        _____RandevuTarih, _____Tbl_Randevular.RandevuSaat,
        _____Tbl_Randevular.RandevuDurum, _____Tbl_Randevular.
        _____Sikayet_ from _____Tbl_Randevular, _____Tbl_Doktorlar,
        _____Tbl_Branslar WHERE _____Tbl_Randevular.Doktorid_ =
        _____Tbl_Doktorlar.Doktorid_ AND _____Tbl_Randevular.
        _____Bransid_ = _____Tbl_Branslar.Bransid",
        bgl.baglanti());

        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(RandevuTarih, RandevuSaat, Bransid,
_____Doktorid, RandevuDurum)_____values (@p1, @p2, (SELECT_____Bransid
_____FROM_____Tbl_Branslar_____WHERE_____BransAd=' " + CmbBrans.Text + " '),
_____ (SELECT_____Doktorid_____FROM_____Tbl_Doktorlar_____WHERE_____DoktorAd+'_' +
_____DoktorSoyad=' " + CmbDoktor.Text + " '), ' False ')",
        bgl.baglanti());

        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_oluřturuldu ." , " Bilgi " ,
        MessageBoxButtons . OK ,
        MessageBoxIcon . Information );
        TabloyuGuncelle ();

```

komutlarında sql komutu yazdım. Sql komutta randevular tablosuna randevu tarihi, randevu saati, branř adına göre branř 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_ Tbl_ Doktorlar
        _____where_ Bransid_=(SELECT_ Bransid_ FROM
        _____Tbl_ Branslar_ WHERE_ BransAd=' "+ CmbBrans . Text
        + " ')", bgl . baglanti ());
        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(Duyuru)_values(@dl)", bgl . baglanti ());  
komut . Parameters . AddWithValue ("@dl" ,  
RchDuyuru . Text );  
komut . ExecuteNonQuery ();  
bgl . baglanti () . Close ();  
MessageBox . Show ("Duyuru_oluşturuldu .", " Bilgi",  
MessageBoxButtons . OK,  
MessageBoxIcon . Information );
```

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 rl = new FrmRandevuListesi ();  
rl . 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_RandevuTarih=@p1,_RandevuSaat=@p2_where_Randevuid=@p6",  
bgl . baglanti ( ) );  
  
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 " , MessageBoxButtons . OK ,  
MessageBoxIcon . Information );  
  
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.ToString();  
    MskTarih.Text = dataGridView1.Rows[secilen].  
        Cells[4].Value.ToString();  
    MskSaat.Text = dataGridView1.Rows[secilen].  
        Cells[5].Value.ToString();  
    CmbBrans.Text = dataGridView1.Rows[secilen].  
        Cells[3].Value.ToString();  
    CmbDoktor.Text = dataGridView1.Rows[secilen].  
        Cells[1].Value.ToString() + " " +  
        dataGridView1.Rows[secilen].  
        Cells[2].Value.ToString();
```

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_Randevuid='" + Txtid.Text + "'",
bgl.baglanti());

        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_silindi.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        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
                _____Tbl_Randevular .Randevuid ,_Tbl_Hastalar .
                _____HastaAd ,_Tbl_Hastalar .HastaSoyad ,
                _____Tbl_Branslar .BransAd ,_Tbl_Doktorlar .
                _____DoktorAd ,_Tbl_Doktorlar .DoktorSoyad ,
                _____Tbl_Randevular .RandevuTarih ,_Tbl_Randevular .
                _____RandevuSaat ,_Tbl_Randevular .RandevuDurum ,
                _____Tbl_Randevular .Sikayet____FROM
                _____Tbl_Randevular ,_Tbl_Branslar ,_Tbl_Doktorlar ,
                _____Tbl_Hastalar_WHERE _Tbl_Randevular .Bransid_ =
                _____Tbl_Branslar .Bransid_ AND _Tbl_Randevular .
                _____Doktorid_ = _Tbl_Doktorlar .Doktorid_ AND
                _____Tbl_Randevular .Hastaid_ =
                _____Tbl_Hastalar .Hastaid_" , bgl .baglanti ());
                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
        _____Tbl_Randevular.Randevuid,_____Tbl_Hastalar.
        _____HastaAd,_____Tbl_Hastalar.HastaSoyad,
        _____Tbl_Branslar.BransAd,_____Tbl_Doktorlar.DoktorAd,
        _____Tbl_Doktorlar.DoktorSoyad,_____Tbl_Randevular.
        _____RandevuTarih,_____Tbl_Randevular.RandevuSaat,
        _____Tbl_Randevular.RandevuDurum,
        _____Tbl_Randevular.Sikayet_____FROM
        _____Tbl_Randevular,_____Tbl_Branslar,_____Tbl_Doktorlar,
        _____Tbl_Hastalar_____WHERE_____Tbl_Randevular.Bransid_=
        _____Tbl_Branslar.Bransid_____AND_____Tbl_Randevular.
        _____Doktorid_=_____Tbl_Doktorlar.Doktorid_____AND
        _____Tbl_Randevular.Hastaid_=
        _____Tbl_Hastalar.Hastaid_____", bgl.baglanti());
        da.Fill(dt);

```

```
dataGridView1.DataSource = dt;
```

dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta 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 seçileni 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 FrmDoktorVeBranslar : Form
    {
        public FrmDoktorVeBranslar()
        {
            InitializeComponent();

            SqlBaglantisı bgl = new SqlBaglantisı();

            private void DoktorVeBranslar_Load(object sender,
            EventArgs e)
            {
                DataTable dt1 = new DataTable();
                SqlDataAdapter da1 = new SqlDataAdapter("select
                _____Doktorid ,DoktorAd ,DoktorSoyad ,BransAd_from
                _____Tbl_Doktorlar ,Tbl_Branslar_where
                _____Tbl_Branslar .Bransid=Tbl_Doktorlar .Bransid",
                bgl.baglanti());
                da1.Fill(dt1);
                dataGridView1.DataSource = dt1;

                DataTable dt2 = new DataTable();
                SqlDataAdapter da2 = new SqlDataAdapter("select
                _____Bransid ,BransAd_from_Tbl_Branslar",
                bgl.baglanti());
                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 ,DoktorAd ,DoktorSoyad ,BransAd_from  
        _____Tbl_Doktorlar ,_Tbl_Branslar_where  
        _____Tbl_Branslar .Bransid=Tbl_Doktorlar .Bransid",  
        bgl.baglanti());  
        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 ,BransAd_from_ Tbl_Branslar",  
        bgl.baglanti());  
        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_Tbl_Branslar", 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
        _____MusteriHizmetleriAd, _____MusteriHizmetleriSoyad
        _____from_Tbl_Musteri_Hizmetleri_where
        _____MusteriHizmetleriTC=@pl", 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
        _____Randevuid ,RandevuTarih ,RandevuSaat ,Sikayet
        _____from_Tbl_Randevular_where_Bransid_IN_(select
        _____Bransid_FROM_Tbl_Branslar_WHERE_BransAd_LIKE
        _____'" + CmbBrans.Text + "')_and_Doktorid_IN
        _____(select_Doktorid_FROM_Tbl_Doktorlar_WHERE
        _____DoktorAd+_','_+_DoktorSoyad_LIKE_'"+
        CmbDoktor.Text + "'_and
        _____RandevuDurum='False ')", bgl.baglanti());
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

    private void CmbBrans_SelectedIndexChanged_1(object
        sender , EventArgs e)
    {
        CmbDoktor.Items.Clear();
        SqlCommand komut3 = new SqlCommand("SELECT
        _____DoktorAd ,DoktorSoyad_FROM_Tbl_Doktorlar_WHERE
        _____Bransid_IN_(select_Bransid_from_tbl_Branslar
        _____where_BransAd_LIKE_@p1)", bgl.baglanti());
        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, EventArgs e)
{
    DataTable dt = new DataTable();
    SqlDataAdapter da = new SqlDataAdapter("select
    _____Randevuid ,RandevuTarih ,RandevuSaat ,Sikayet
    _____from_Tbl_Randevular_where_Bransid_IN_(select
    _____Bransid_FROM_Tbl_Branslar_WHERE_BransAd_LIKE
    _____'" + CmbBrans.Text + "')_and_Doktorid_IN
    _____(select_Doktorid_FROM_Tbl_Doktorlar_WHERE
    _____DoktorAd_+_ ' ' _+_DoktorSoyad_LIKE_'" +
    _____CmbDoktor.Text + "'_and
    _____RandevuDurum='False ')", bgl.baglanti());
    da.Fill(dt);
    dataGridView2.DataSource = dt;
}

private void BtnRandevuAl_Click(object sender, EventArgs e)
{
    SqlCommand komut = new SqlCommand("UPDATE
    _____Tbl_Randevular_SET_RandevuDurum='True ',Hastaid=
    _____(SELECT_Hastaid_FROM_Tbl_Hastalar_WHERE_HastaTC
    _____='"+ txtHastaTC.Text + "') ,Sikayet=@r2_WHERE
    _____Randevuid='" + Txtid.Text + "'_",
    bgl.baglanti());
    komut.Parameters.AddWithValue("@r2",
    RchSikayet.Text);
    komut.ExecuteNonQuery();
    bgl.baglanti().Close();
    MessageBox.Show("Randevu_oluřturuldu.",

```

```

        "Bilgi", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloGuncelle();
    }

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

    private void btnRandevularıGetir_Click(object
        sender, EventArgs e)
    {
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter("select
        _____Randevuid ,RandevuTarih ,RandevuSaat ,Sikayet
        _____from _Tbl_Randevular _where _Hastaid=(SELECT
        _____Hastaid _FROM _Tbl_Hastalar _WHERE _HastaTC='"+
            txtHastaTC.Text +"')_", bgl.baglanti());
        da.Fill(dt);
        dataGridView2.DataSource = dt;
    }

    private void btnRandevuSil_Click(object sender,
        EventArgs e)
    {
        SqlCommand komut = new SqlCommand("UPDATE

```

```

        Tbl_Randevular.SET_RandevuDurum='False'
        where_Randevuid=@p1", bgl.baglanti());
        komut.Parameters.AddWithValue("@p1",
        Txtid.Text);
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu_Silindi.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        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", bgl.baglanti());
        SqlDataReader dr2 = komut2.ExecuteReader();
        while (dr2.Read())
        {
            CmbBrans.Items.Add(dr2[0]);
        }
        bgl.baglanti().Close();

```

Sql komutta branş adlarını Branşlar 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
        _____MusteriHizmetleriAd ,_MusteriHizmetleriSoyad
        _____from _Tbl_Musteri_Hizmetleri_where
        _____MusteriHizmetleriTC=@p1", bgl.baglanti());
        komut.Parameters.AddWithValue("@p1",
        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 MusteriHizmetle-

riTC'yi parametre alıp getir dedim. dr1 isimli sql veri okuyucuyu komutu okutarak çalıştır dedim. 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
        _____Randevuid ,RandevuTarih ,RandevuSaat ,Sikayet
        _____from_Tbl_Randevular_where_ Bransid_IN_(select
        _____Bransid_FROM_Tbl_Branslar_WHERE_ BransAd_LIKE
        _____'" + CmbBrans.Text + "' )_and_ Doktorid_IN
        _____(select_ Doktorid_FROM_Tbl_Doktorlar_WHERE
        _____DoktorAd_+_'_'_+_DoktorSoyad_LIKE_'"+
        _____CmbDoktor.Text + "'_and_RandevuDurum='False' )
        _____", bgl.baglanti());
        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.

CmbBrans_SelectedIndexChanged_1 içinde;

```
CmbDoktor.Items.Clear();
        SqlCommand komut3 = new SqlCommand("SELECT
        _____DoktorAd ,DoktorSoyad_FROM_Tbl_Doktorlar
        _____WHERE_ Bransid_IN_(select_ Bransid_from
        _____tbl_Branslar_where_ BransAd_LIKE_@p1)",
        _____bgl.baglanti());
```

```

        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
        _____Randevuid ,RandevuTarih ,RandevuSaat ,Sikayet
        _____from _Tbl_Randevular _where _Bransid _IN_( select
        _____Bransid _FROM _Tbl_Branslar _WHERE _BransAd _LIKE
        _____'" + CmbBrans.Text + "' )_and _Doktorid _IN
        _____( select _Doktorid _FROM _Tbl_Doktorlar _WHERE
        _____DoktorAd _+_'_'_+ _DoktorSoyad _LIKE_' " +
        CmbDoktor.Text + "'_and _RandevuDurum=' False ' )
        _____", bgl.baglanti());
        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 doldur dedim. Veri kafesi görünümü 2'nin veri kaynağına dt isimli veri tablosunu aktardım.

BtnRandevuAl_Click içinde;

```
SqlCommand komut = new SqlCommand("UPDATE_Tbl_Randevular
SET_RandevuDurum=' True ', HastaId=(SELECT_HastaId_FROM
Tbl_Hastalar_WHERE_HastaTC_='"+ txtHastaTC.Text +"' ),
Sikayet=@r2_WHERE_Randevuid='" + Txtid.Text + "'_",
bgl.baglanti());

komut.Parameters.AddWithValue("@r2",
RchSikayet.Text);
komut.ExecuteNonQuery();
bgl.baglanti().Close();
MessageBox.Show("Randevu_oluşturuldu.", "Bilgi",
MessageBoxButtons.OK,
MessageBoxIcon.Information);
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.ToString();
```

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
        _____Randevid ,RandevuTarih ,RandevuSaat ,Sikayet
        _____from_Tbl_Randevular_where_Hastaid=(SELECT
        _____Hastaid_FROM_Tbl_Hastalar_WHERE_HastaTC='"+
        txtHastaTC.Text +"' )_", bgl.baglanti());
        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_RandevuDurum='False ',_Hastaid=null_WHERE_Randevid='" +
    Txtid.Text + "'_", bgl.baglanti());
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Randevu iptal edildi.",
        "Bilgi", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloGuncelle();

```

Randevuyu iptal et butonunda randevular tablosundan randevu durumunu pasif ve tablolar id'lerle birbirine bağlı olduğundan Hastaid'sini null olarak Randevid'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 12: 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 ,YoneticiSoyad_from_Tbl_Yoneticiler
    where_YoneticiTC=@p1", bgl.baglanti());
    komut.Parameters.AddWithValue("@p1",
    YoneticiTC);
    SqlDataReader dr = komut.ExecuteReader();
    while (dr.Read())
    {
        LblAdSoyad.Text = dr[0] + "_" + dr[1];
    }
    bgl.baglanti().Close();
}

private void LnkBilgiDuzenle_LinkClicked(object
sender, LinkLabelLinkClickedEventArgs 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 linkYoneticiciEkle_LinkClicked(object
sender, LinkLabelLinkClickedEventArgs e)
{
    FrmYoneticiciEkle fr = new FrmYoneticiciEkle();
```

```

        fr.Show();
    }

    private void linkSekreterEkle_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
    {
        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 , YoneticiSoyad_from_Tbl_Yoneticiler
.....where_YoneticiTC=@p1" , bgl.baglanti ());
        komut.Parameters.AddWithValue( "@p1" ,
        YoneticiTC );
        SqlDataReader dr = komut.ExecuteReader ();
        while ( dr.Read ())
        {
            LblAdSoyad.Text = dr[0] + "_" + dr[1];
        }
        bgl.baglanti ().Close ();

```

komutlarında YoneticiTC'yi LblTC'nin yazısına aktardım. Sql komutta yönetici ad ve soyadı Yöneticiler tablosundan YoneticiTC'ye göre YoneticiTC'yi parametre olarak sql bağlantısını sağlayarak getir dedim. Sql veri okuyucuyu komutu okutarak çalıştır dedim. Okurken ad ve soyadı arada boşluk bırakarak ilgili etikete yaz dedim ve bağlantıyı kapattım.

LnkBilgiDuzenle_LinkClicked içinde;

```

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

```

komutlarında FrmYoneticiBilgiDuzenle'nin TCno'suna ilgili etiketteki yönetici TC'yi aktardım ve FrmYoneticiBilgiDuzenle formunu açtım.

BtnDoktorPaneli_Click içinde;

```

FrmDoktorPaneli drp = new FrmDoktorPaneli ();
        drp.Show ();

```

komutunda FrmDoktorPaneli formunu açtım.

BtnBransPaneli_Click içinde;

```
FrmBransPaneli frb = new FrmBransPaneli ();  
    frb . Show ();
```

komutunda FrmBransPaneli formunu açtım.

BtnRandevuListe_Click içinde;

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

komutunda FrmRandevuListesi formunu açtım.

button1_Click içine;

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

komutunda FrmDuyurular formunu açtım.

linkYoneticiEkle_LinkClicked içinde;

```
FrmYoneticiEkle fr = new FrmYoneticiEkle ();  
    fr . Show ();
```

komutunda FrmYoneticiEkle formunu açtım.

linkSekreterEkle_LinkClicked içinde;

```
FrmSekreterEkle fr = new FrmSekreterEkle ();  
    fr . Show ();
```

komutunda FrmSekreterEkle formunu açtım.

button2_Click içinde;

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

komutunda uygulamadan çık dedim.

Dosya 13: Yönetici 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 FrmYoneticiciBilgiDuzenle : Form
    {
        public FrmYoneticiciBilgiDuzenle()
        {
            InitializeComponent();
        }
        public string TCno;

        SqlBaglantisi bgl = new SqlBaglantisi();

        private void BtnBilgiGuncelle_Click(object sender,
            EventArgs e)
        {
            if (FrmHastaKayit.TCKontrol(MskTC.Text)) {
                SqlCommand komut2 = new SqlCommand("update
                _____Tbl_Yoneticiler_set_YoneticiciAd=@p1,
                _____YoneticiciSoyad=@p2,_____YoneticiciSifre=@p3_where
```



```

        YoneticiTC=@p4", bgl.baglanti());
        komut2.Parameters.AddWithValue("@p1",
        TxtAd.Text);
        komut2.Parameters.AddWithValue("@p2",
        TxtSoyad.Text);
        komut2.Parameters.AddWithValue("@p3",
        TxtSifre.Text);
        komut2.Parameters.AddWithValue("@p4",
        MskTC.Text);
        komut2.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Bilgileriniz_güncellendi."
        , "Bilgi", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
    }
}

private void FrmYoneticiBilgiDuzenle_Load(object
sender, EventArgs e)
{
    MskTC.Text = TCno;
    SqlCommand komut = new SqlCommand("select_*
        from_Tbl_Yoneticiler_where_YoneticiTC=@p1",
        bgl.baglanti());
    komut.Parameters.AddWithValue("@p1", TCno);
    SqlDataReader dr = komut.ExecuteReader();
    while (dr.Read())
    {
        TxtAd.Text = dr[2].ToString();
        TxtSoyad.Text = dr[3].ToString();
        TxtSifre.Text = dr[4].ToString();
    }
}

```

```

        }
        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 TCno;
```

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

BtnBilgiGuncelle_Click içinde;

```

if (FrmHastaKayit.TCKontrol(MskTC.Text)) {
    SqlCommand komut2 = new SqlCommand("update
    _____Tbl_Yoneticiler_set_YoneticAd=@p1,
    _____YoneticSoyad=@p2, _YoneticSifre=@p3_where
    _____YoneticTC=@p4", bgl.baglanti());
    komut2.Parameters.AddWithValue("@p1",
    TxtAd.Text);
    komut2.Parameters.AddWithValue("@p2",
    TxtSoyad.Text);
    komut2.Parameters.AddWithValue("@p3",
    TxtSifre.Text);
    komut2.Parameters.AddWithValue("@p4",
    MskTC.Text);
    komut2.ExecuteNonQuery();
}

```

```

        bgl.baglanti().Close();
        MessageBox.Show("Bilgileriniz_güncellendi.",
            "Bilgi", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
    }

```

komutlarında TC kontrolü yapıldıktan sonra sql komutta Yöneticiler tablosunda yönetici adı ve soyadı, yönetici şifreyi ve yönetici TC'sini YoneticiTC'ye göre sql bağlantısını sağlayarak ve TxtAd'ın yazısını, TxtSoyad'ın yazısını, TxtSifre'nin yazısını ve MskTC'nin yazısını parametre olarak güncelledim. Komutu çalıştırdım ve bağlantıyı kapattım. Sonra "Bilgileriniz güncellendi." şeklinde bilgi mesajı verdirdim.

FrmYoneticiBilgiDuzenle_Load içinde;

```

MskTC.Text = TCno;

        SqlCommand komut = new SqlCommand("select_*
        _____from_Tbl_Yoneticiler_where_YoneticiTC=@p1",
        bgl.baglanti());
        komut.Parameters.AddWithValue("@p1", TCno);
        SqlDataReader dr = komut.ExecuteReader();
        while (dr.Read())
        {
            TxtAd.Text = dr[2].ToString();
            TxtSoyad.Text = dr[3].ToString();
            TxtSifre.Text = dr[4].ToString();
        }
        bgl.baglanti().Close();

```

komutlarında TCno'yu MskTC'nin yazısına aktardım. Sql komutta Yöneticiler tablosundan YoneticiTC'ye göre TCno'yu parametre olarak ve sql bağlantısını sağlayarak getir verileri dedim. Sql veri okuyucuyu komutu okutarak çalıştır dedim. Okurken adı TxtAd'nin yazısına, soyadı TxtSoyad'nin yazısına ve şifreyi TxtSifre'nin yazısına aktardım ve bağlantıyı kapattım.

Dosya 14: Yönetici Ekle 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 FrmYoneticiEkle : Form
    {
        public FrmYoneticiEkle()
        {
            InitializeComponent();

            SqlBaglantisı bgl = new SqlBaglantisı();
            private void BtnYoneticiEkle_Click(object sender,
            EventArgs e)
            {
                if (FrmHastaKayıt.TCKontrol(MskTC.Text)) {
                    SqlCommand komut2 = new SqlCommand("insert
                    into
                    Tbl_Yoneticiler(YoneticiTC,YoneticiAd,
                    YoneticiSoyad,YoneticiSifre)_values
                    (@p1,@p2,@p3,@p4)", bgl.baglanti());
```

```

        komut2.Parameters.AddWithValue("@p2",
        TxtAd.Text);
        komut2.Parameters.AddWithValue("@p3",
        TxtSoyad.Text);
        komut2.Parameters.AddWithValue("@p4",
        TxtSifre.Text);
        komut2.Parameters.AddWithValue("@p1",
        MskTC.Text);
        komut2.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Yönetici_Eklendi",
        "Bilgi", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloGuncelle();
    }
}

private void TabloGuncelle()
{
    DataTable dt2 = new DataTable();
    SqlDataAdapter da2 = new SqlDataAdapter("select
    _____Yoneticiid ,YoneticAd ,YoneticSoyad ,YoneticTC
    _____,YoneticSifre_from_Tbl_Yoneticiler",
    bgl.baglanti());
    da2.Fill(dt2);
    dataGridView1.DataSource = dt2;
}

private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
{

```

```

        int secilen = dataGridView1.SelectedCells[0].
RowIndex;
        TxtYoneticiid.Text = dataGridView1.
Rows[secilen].Cells[0].Value.ToString();
        TxtAd.Text = dataGridView1.Rows[secilen].
Cells[1].Value.ToString();
        TxtSoyad.Text = dataGridView1.Rows[secilen].
Cells[2].Value.ToString();
        MskTC.Text = dataGridView1.Rows[secilen].
Cells[3].Value.ToString();
        TxtSifre.Text = dataGridView1.Rows[secilen].
Cells[4].Value.ToString();
    }

    private void btnYoneticisi_Click(object sender,
EventArgs e)
    {
        SqlCommand komut2 = new SqlCommand("delete_from
        Tbl_Yoneticiler_where_Yoneticiid=@p1",
        bgl.baglanti());
        komut2.Parameters.AddWithValue("@p1",
        TxtYoneticiid.Text);
        komut2.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Yönetici_Silindi.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloGuncelle();
    }

    private void FrmYoneticisiEkle_Load(object sender,

```

```

        EventArgs e)
        {
            DataTable dt2 = new DataTable();
            SqlDataAdapter da2 = new SqlDataAdapter("select
            _____Yoneticiid ,YoneticiAd ,YoneticiSoyad ,YoneticiTC
            _____,YoneticiSifre_from_Tbl_Yoneticiler",
            bgl.baglanti());
            da2.Fill(dt2);
            dataGridView1.DataSource = dt2;
        }

        private void button1_Click(object sender,
        EventArgs e)
        {
            SqlCommand komut2 = new SqlCommand("update
            _____Tbl_Yoneticiler_set_YoneticiAd=@p1,
            _____YoneticiSoyad=@p2,_YoneticiTC=@p3,
            _____YoneticiSifre=@p4_where_Yoneticiid=@p5",
            bgl.baglanti());
            komut2.Parameters.AddWithValue("@p1",
            TxtAd.Text);
            komut2.Parameters.AddWithValue("@p2",
            TxtSoyad.Text);
            komut2.Parameters.AddWithValue("@p3",
            MskTC.Text);
            komut2.Parameters.AddWithValue("@p4",
            TxtSifre.Text);
            komut2.Parameters.AddWithValue("@p5",
            TxtYoneticiid.Text);

            komut2.ExecuteNonQuery();

```

```

        bgl.baglanti().Close();
        MessageBox.Show("Yönetici_Güncellendi.", "Bilgi",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        TabloGuncelle();

    }

}

}

using System.Data.SqlClient;

```

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

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

komutuyla Sql bağlantısını yaptım.

TabloGuncelle fonksiyonu içinde;

```

DataTable dt2 = new DataTable();

        SqlDataAdapter da2 = new SqlDataAdapter("select
        _____Yoneticiid ,YoneticiAd ,YoneticiSoyad ,YoneticiTC
        _____,YoneticiSifre_from_Tbl_Yonet iciler",
            bgl.baglanti());
        da2.Fill(dt2);
        dataGridView1.DataSource = dt2;

```

komutlarında dt2 isimli veri tablosu oluşturdum. da2 isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta yönetici id, yönetici ad ve soyad, Yönetici TC ve yönetici şifreyi Yöneticiler tablosundan sql bağlantısını sağlayarak getir dedim. da2 isimli sql veri uyarlayıcıya dt2 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt2 isimli veri tablosunu aktardım.

BtnYoneticiEkle_Click içinde;


```

if ( FrmHastaKayit . TCKontrol ( MskTC . Text ) ) {
    SqlCommand komut2 = new SqlCommand ( " insert
    _____ into
    _____ Tbl_Yoneticiler ( YoneticiTC , YoneticiAd ,
    _____ YoneticiSoyad , YoneticiSifre ) _ values
    _____ ( @p1 , _ @p2 , _ @p3 , _ @p4 ) " , bgl . baglanti ( ) ) ;
    komut2 . Parameters . AddWithValue ( " @p2 " ,
    TxtAd . Text ) ;
    komut2 . Parameters . AddWithValue ( " @p3 " ,
    TxtSoyad . Text ) ;
    komut2 . Parameters . AddWithValue ( " @p4 " ,
    TxtSifre . Text ) ;
    komut2 . Parameters . AddWithValue ( " @p1 " ,
    MskTC . Text ) ;
    komut2 . ExecuteNonQuery ( ) ;
    bgl . baglanti ( ) . Close ( ) ;
    MessageBox . Show ( " Yönetici _ Eklendi " ,
    " Bilgi " , MessageBoxButtons . OK ,
    MessageBoxIcon . Information ) ;
    TabloGuncelle ( ) ;
}

```

komutlarında TC kontrolü yapıp sql komutu yazdım. Sql komutta Yöneticiler tablosuna yönetici TC'yi yönetici adı, yönetici soyadı ve yönetici şifreyi TxtAd'nin yazısını, TxtSoyad'nin yazısını, TxtSifre'nin yazısını ve MskTC'nin yazısını parametre olarak sql bağlantısını yapıp ekle dedim. Komutu çalıştırdım ve bağlantıyı kapattım. Sonra "Yönetici Eklendi" şeklinde bilgi mesajı verip yönetici tablosunu güncelledim.

dataGridView1_CellClick içinde;

```

int secilen = dataGridView1 . SelectedCells [ 0 ] . RowIndex ;
    TxtYoneticiid . Text = dataGridView1 .

```

```

Rows[ secilen ]. Cells [0]. Value . ToString ();
TxtAd . Text = dataGridView1 . Rows[ secilen ].
Cells [1]. Value . ToString ();
TxtSoyad . Text = dataGridView1 . Rows[ secilen ].
Cells [2]. Value . ToString ();
MskTC . Text = dataGridView1 . Rows[ secilen ].
Cells [3]. Value . ToString ();
TxtSifre . Text = dataGridView1 . Rows[ secilen ].
Cells [4]. Value . ToString ();

```

komutlarında yönetici id'yi TxtYoneticiid'nin yazısına, yönetici adı TxtAd'nin yazısına, yönetici soyadı TxtSoyad'nin yazısına, yönetici TC'yi MskTC'nin yazısına ve yönetici şifreyi TxtSifre'nin yazısına yönetici tablosunda bir kez tıklanılan satırın satır indeksini seçilen değişkenine alarak aktardım.

btnYoneticisi_Click içinde;

```

SqlCommand komut2 = new SqlCommand("delete_from
Tbl_Yoneticiler_where_Yoneticiid=@p1", bgl.baglanti());
komut2.Parameters.AddWithValue("@p1",
TxtYoneticiid.Text);
komut2.ExecuteNonQuery();
bgl.baglanti().Close();
MessageBox.Show("Yönetici_Silindi.", "Bilgi",
MessageBoxButtons.OK,
MessageBoxIcon.Information);
TabloGuncelle();

```

komutlarında Yöneticiler tablosundan Yoneticiid'ye göre TxtYoneticiid'nin yazısını parametre olarak ve sql bağlantısını sağlayarak sil dedim. Komutu çalıştırdım ve bağlantıyı kapattım. Sonra "Yönetici Silindi." şeklinde bilgi mesajı verdirdim ve yönetici tablosunu güncelledim.

FrmYoneticiEkle_Load içinde;

```
DataTable dt2 = new DataTable();  
        SqlDataAdapter da2 = new SqlDataAdapter(" select  
        _____ Yoneticiid , YoneticiAd , YoneticiSoyad , YoneticiTC  
        _____ , YoneticiSifre _from_ Tbl_Yonet iciler " ,  
        bgl . baglanti ());  
        da2 . Fill ( dt2 );  
        dataGridView1 . DataSource = dt2 ;
```

komutlarında dt2 isimli veri tablosu oluşturdum. da2 isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta yönetici id'yi, yönetici adı ve soyadı, Yönetici TC'yi ve yönetici şifreyi Yöneticiler tablosundan sql bağlantısını sağlayarak getir dedim. da2 isimli sql veri uyarlayıcıya dt2 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt2 isimli veri tablosunu aktardım.

button1_Click içinde;

```
SqlCommand komut2 = new SqlCommand(" update_ Tbl_Yonet iciler  
set_ YoneticiAd=@p1, _ YoneticiSoyad=@p2, _ YoneticiTC=@p3,  
YoneticiSifre=@p4_ where_ Yoneticiid=@p5" , bgl . baglanti ());  
        komut2 . Parameters . AddWithValue ( "@p1" ,  
        TxtAd . Text );  
        komut2 . Parameters . AddWithValue ( "@p2" ,  
        TxtSoyad . Text );  
        komut2 . Parameters . AddWithValue ( "@p3" ,  
        MskTC . Text );  
        komut2 . Parameters . AddWithValue ( "@p4" ,  
        TxtSifre . Text );  
        komut2 . Parameters . AddWithValue ( "@p5" ,  
        TxtYoneticiid . Text );  
  
        komut2 . ExecuteNonQuery ( );
```

```

        bgl.baglanti().Close();
        MessageBox.Show("Yönetici_Güncellendi.",
        "Bilgi", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloGuncelle();

```

komutlarında sql komut yazdım. Sql komutta Yöneticiler tablosunda yönetici adı ve soyadı, yönetici TC'yi ve yönetici şifreyi Yoneticiid'ye göre TxtAd'nin yazısını, TxtSoyad'nin yazısını, MskTC'nin yazısını, TxtSifre'nin yazısını ve TxtYoneticiid'nin yazısını parametre olarak ve sql bağlantısını yaparak güncelle dedim. Komutu çalıştırdım ve sql bağlantısını kapattım. Sonra "Yönetici Güncellendi." şeklinde bilgi mesajı verdirdim ve yönetici tablosunu güncelledim.

Dosya 15: Doktor Paneli 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 FrmDoktorPaneli : Form
    {
        public FrmDoktorPaneli()
        {

```

```

        InitializeComponent();
    }

    SqlBaglantisi bgl = new SqlBaglantisi();

    private void FrmDoktorPaneli_Load(object sender,
    EventArgs e)
    {
        // Doktorları Datagridview2'ye aktarma
        DataTable dt1 = new DataTable();
        SqlDataAdapter da1 = new SqlDataAdapter(" select
        Tbl_Doktorlar.Doktorid , Tbl_Doktorlar.DoktorAd
        , Tbl_Doktorlar.DoktorSoyad , Tbl_Branslar .
        BransAd , Tbl_Doktorlar.DoktorTC , Tbl_Doktorlar
        .DoktorSifre from Tbl_Doktorlar , Tbl_Branslar
        WHERE Tbl_Doktorlar.Bransid = Tbl_Branslar .
        Bransid " , bgl.baglanti());
        da1.Fill(dt1);
        dataGridView1.DataSource = dt1;

        // Branşları Combobox'a aktarma
        SqlCommand komut1 = new SqlCommand(" select
        BransAd from Tbl_Branslar " , bgl.baglanti());
        SqlDataReader dr1 = komut1.ExecuteReader();
        while (dr1.Read())
        {
            CmbBrans.Items.Add(dr1[0]);
        }
        bgl.baglanti().Close();
    }

```

```

private void TabloGuncelle()
{
    //Tablo güncelleme
    DataTable dt1 = new DataTable();
    SqlDataAdapter da1 = new SqlDataAdapter("select
    _____Tbl_Doktorlar.Doktorid ,_Tbl_Doktorlar.DoktorAd
    _____, _Tbl_Doktorlar.DoktorSoyad ,_Tbl_Branslar .
    _____BransAd ,_Tbl_Doktorlar.DoktorTC ,_Tbl_Doktorlar
    _____.DoktorSifre_from _Tbl_Doktorlar ,_Tbl_Branslar
    _____WHERE _Tbl_Doktorlar.Bransid=_Tbl_Branslar .
    _____Bransid", bgl.baglanti());
    da1.Fill(dt1);
    dataGridView1.DataSource = dt1;
}

private void BtnEkle_Click(object sender,
EventArgs e)
{
    if (FrmHastaKayit.TCKontrol(MskTC.Text)) {
        SqlCommand komut = new SqlCommand("insert
        _____into
        _____Tbl_Doktorlar(DoktorAd ,DoktorSoyad ,Bransid
        _____,DoktorTC ,DoktorSifre)_values (@d1,@d2,
        _____(SELECT _Bransid_FROM _Tbl_Branslar_WHERE
        _____BransAd=' " + CmbBrans.Text + " '),
        _____@d4,@d5)", bgl.baglanti());
        komut.Parameters.AddWithValue("@d1",
        TxtAd.Text);
        komut.Parameters.AddWithValue("@d2",
        TxtSoyad.Text);
        komut.Parameters.AddWithValue("@d4",

```

```

        MskTC.Text);
        komut.Parameters.AddWithValue("@d5",
        TxtSifre.Text);
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Doktor_eklendi.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        TabloGuncelle();
    }
}

private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    int secilen = dataGridView1.SelectedCells[0].
    RowIndex;
    TxtDoktorid.Text = dataGridView1.Rows[secilen].
    Cells[0].Value.ToString();
    TxtAd.Text = dataGridView1.Rows[secilen].
    Cells[1].Value.ToString();
    TxtSoyad.Text = dataGridView1.Rows[secilen].
    Cells[2].Value.ToString();
    CmbBrans.Text = dataGridView1.Rows[secilen].
    Cells[3].Value.ToString();
    MskTC.Text = dataGridView1.Rows[secilen].
    Cells[4].Value.ToString();
    TxtSifre.Text = dataGridView1.Rows[secilen].
    Cells[5].Value.ToString();
}

```

```

        private void BtnSil_Click(object sender ,
        EventArgs e)
        {
            SqlCommand komut2 = new SqlCommand("DELETE_FROM
            _____Tbl_Randevular_WHERE_Doktorid='" +
            TxtDoktorid.Text + "'", bgl.baglanti());
            komut2.ExecuteNonQuery();
            bgl.baglanti().Close();

            SqlCommand komut = new SqlCommand("DELETE_FROM
            _____Tbl_Doktorlar_WHERE_Doktorid='"+
            TxtDoktorid.Text + "'", bgl.baglanti());
            komut.ExecuteNonQuery();
            bgl.baglanti().Close();
            MessageBox.Show("Doktor_kaydı_silindi.",
            "Bilgi", MessageBoxButtons.OK,
            MessageBoxIcon.Asterisk);
            TabloGuncelle();
        }

        private void BtnGuncelle_Click(object sender ,
        EventArgs e)
        {
            if (FrmHastaKayit.TCKontrol(MskTC.Text))
            {
                SqlCommand komut = new SqlCommand("update
                _____Tbl_Doktorlar_set_DoktorAd=@d2,
                _____DoktorSoyad=@d3,_Bransid=(SELECT_Bransid
                _____FROM_Tbl_Branslar_WHERE_BransAd=@d4),
                _____DoktorTC=@d5,_DoktorSifre=@d6_where
                _____Doktorid=@d1", bgl.baglanti());
            }
        }
    
```



```

        komut.Parameters.AddWithValue("@d1",
        TxtDoktorid.Text);
        komut.Parameters.AddWithValue("@d2",
        TxtAd.Text);
        komut.Parameters.AddWithValue("@d3",
        TxtSoyad.Text);
        komut.Parameters.AddWithValue("@d4",
        CmbBrans.Text);
        komut.Parameters.AddWithValue("@d5",
        MskTC.Text);
        komut.Parameters.AddWithValue("@d6",
        TxtSifre.Text);

        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        MessageBox.Show("Doktor_bilgisi
        güncellendi.", "Bilgi",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);

        TabloGuncelle();
    }
}
}
}

```

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

komutuyla Sql komutlarının çalışması için SqlConnection kütüphanesini ekledim.

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

komutuyla Sql bağlantısını yaptım.

FrmDoktorPaneli_Load içinde;

```
// Doktorları DataGridView1'e aktarma
DataTable dt1 = new DataTable();
SqlDataAdapter da1 = new SqlDataAdapter(" select
Tbl_Doktorlar.Doktorid , Tbl_Doktorlar.DoktorAd
, Tbl_Doktorlar.DoktorSoyad , Tbl_Branslar.
BransAd , Tbl_Doktorlar.DoktorTC ,
Tbl_Doktorlar.DoktorSifre from Tbl_Doktorlar ,
Tbl_Branslar WHERE Tbl_Doktorlar.Bransid =
Tbl_Branslar.Bransid " , bgl.baglanti());
da1.Fill(dt1);
dataGridView1.DataSource = dt1;
```

komutlarında dt1 isimli veri tablosu oluşturdum. da1 isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta doktor id'yi, doktor adı ve soyadı, branş adı, doktor TC'yi ve doktor şifreyi Doktorlar ve Branşlar tablosundan sql bağlantısını sağlayarak getir dedim. da1 isimli sql veri uyarlayıcıya dt1 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt1 isimli veri tablosunu aktardım.

```
// Branşları Combobox'a aktarma
SqlCommand komut1 = new SqlCommand(" select
BransAd from Tbl_Branslar " , bgl.baglanti());
SqlDataReader dr1 = komut1.ExecuteReader();
while (dr1.Read())
{
CmbBrans.Items.Add(dr1[0]);
}
bgl.baglanti().Close();
```

komutlarında branş adı Branşlar tablosundan sql bağlantısını sağlayarak getir dedim. Sql veri okuyucuyu komutu da okutarak çalıştır dedim. Okurken branşları branş combobox'ına ekledim ve bağlantıyı kapattım.

TabloGuncelle fonksiyonu içinde;

```
// Tablo güncelleme
```

```
        DataTable dt1 = new DataTable();
        SqlDataAdapter da1 = new SqlDataAdapter("select
        _____Tbl_Doktorlar.Doktorid ,Tbl_Doktorlar.DoktorAd
        _____,Tbl_Doktorlar.DoktorSoyad ,Tbl_Branslar.
        _____BransAd ,Tbl_Doktorlar.DoktorTC ,Tbl_Doktorlar
        _____.DoktorSifre_from_Tbl_Doktorlar ,Tbl_Branslar
        _____WHERE_Tbl_Doktorlar.Bransid=Tbl_Branslar.
        _____Bransid", bgl.baglanti());
        da1.Fill(dt1);
        dataGridView1.DataSource = dt1;
```

komutlarında dt1 isimli veri tablosu oluşturdum. da1 isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta doktor id'yi, doktor adı ve soyadı, brans adı, doktor TC'yi ve doktor şifreyi Doktorlar ve Branslar tablosundan sql bağlantısını sağlayarak getir dedim. da1 isimli sql veri uyarlayıcıya dt1 isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt1 isimli veri tablosunu aktardım.

BtnEkle_Click içinde;

```
if (FrmHastaKayit.TCKontrol(MskTC.Text)) {
        SqlCommand komut = new SqlCommand("insert
        _____into
        _____Tbl_Doktorlar(DoktorAd ,DoktorSoyad ,Bransid
        _____,DoktorTC ,DoktorSifre)_values (@d1,@d2,
        _____(SELECT_Bransid_FROM_Tbl_Branslar_WHERE
        _____BransAd=' " + CmbBrans.Text + " '),
        _____@d4,@d5)", bgl.baglanti());
        komut.Parameters.AddWithValue("@d1",
        TxtAd.Text);
        komut.Parameters.AddWithValue("@d2",
```

```

        TxtSoyad . Text );
        komut . Parameters . AddWithValue ( "@d4" ,
        MskTC . Text );
        komut . Parameters . AddWithValue ( "@d5" ,
        TxtSifre . Text );
        komut . ExecuteNonQuery ();
        bgl . baglanti () . Close ();
        MessageBox . Show ( " Doktor_eklendi . " , " Bilgi " ,
        MessageBoxButtons . OK ,
        MessageBoxIcon . Information );
        TabloGuncelle ();
    }

```

komutlarında TC kontrolü yapıp sql komutu yazdım. Sql komutta Doktorlar tablosuna doktor adı, doktor soyadı, branş id'yi branş ada göre, doktor TC'yi ve doktor şifreyi TxtAd'nin yazısını, TxtSoyad'nin yazısını, MskTC'nin yazısını ve TxtSifre'nin yazısını parametre olarak sql bağlantısını yapıp ekle dedim. Komutu çalıştırdım ve bağlantıyı kapattım. Sonra "Doktor Eklendi" şeklinde bilgi mesajı verip doktor tablosunu güncelledim.

dataGridView1_CellClick içinde;

```

int  secilen  = dataGridView1 . SelectedCells [ 0 ] . RowIndex ;
        TxtDoktorid . Text  = dataGridView1 . Rows [ secilen ] .
        Cells [ 0 ] . Value . ToString ();
        TxtAd . Text  = dataGridView1 . Rows [ secilen ] .
        Cells [ 1 ] . Value . ToString ();
        TxtSoyad . Text  = dataGridView1 . Rows [ secilen ] .
        Cells [ 2 ] . Value . ToString ();
        CmbBrans . Text  = dataGridView1 . Rows [ secilen ] .
        Cells [ 3 ] . Value . ToString ();
        MskTC . Text  = dataGridView1 . Rows [ secilen ] .
        Cells [ 4 ] . Value . ToString ();

```

```
TxtSifre.Text = dataGridView1.Rows[secilen].  
Cells[5].Value.ToString();
```

komutlarında doktor id'yi TxtDoktorid'nin yazısına, doktor adı TxtAd'nin yazısına, doktor soyadı TxtSoyad'nin yazısına, doktor branşının adı CmbBrans'nin yazısına, doktor TC'yi MskTC'nin yazısına ve doktor şifreyi TxtSifre'nin yazısına doktor tablosunda bir kez tıklanılan satırın satır indeksini seçilen değişkenine alarak aktardım.

BtnSil_Click içinde;

```
SqlCommand komut2 = new SqlCommand("DELETE_FROM  
Tbl_Randevular_WHERE_Doktorid='" +  
TxtDoktorid.Text + "'", bgl.baglanti());  
komut2.ExecuteNonQuery();  
bgl.baglanti().Close();
```

komutunda Randevular tablosundan Doktorid'ye göre sql bağlantısını sağlayarak sildim ve komutu çalıştırdım. Sonra bağlantıyı kapattım. Doktoru olmayan doktorun randevusu olmaması için randevularıyla birlikte silmeliyim.

```
SqlCommand komut = new SqlCommand("DELETE_FROM  
Tbl_Doktorlar_WHERE_Doktorid='" + TxtDoktorid.Text + "'",  
bgl.baglanti());  
komut.ExecuteNonQuery();  
bgl.baglanti().Close();  
MessageBox.Show("Doktor_kaydı_silindi.",  
"Bilgi", MessageBoxButtons.OK,  
MessageBoxIcon.Asterisk);  
TabloGuncelle();
```

komutlarında Doktorlar tablosundan Doktorid'ye göre sql bağlantısını sağlayarak sildim ve komutu çalıştırdım. Sql bağlantısını kapattım ve "Doktor kaydı silindi." şeklinde bilgi mesajı verdirdim. Sonra doktor tablosunu güncelledim.

BtnGuncelle_Click içinde;

```
if ( FrmHastaKayit . TCKontrol ( MskTC . Text ) )
{
    SqlCommand komut = new SqlCommand ( "update
    .....Tbl_Doktorlar _set _DoktorAd=@d2,
    .....DoktorSoyad=@d3, _Bransid=(SELECT _Bransid
    .....FROM _Tbl_Branslar _WHERE _BransAd=@d4) ,
    .....DoktorTC=@d5, _DoktorSifre=@d6 _where
    .....Doktorid=@d1" , bgl . baglanti ( ) );
    komut . Parameters . AddWithValue ( "@d1" ,
    TxtDoktorid . Text );
    komut . Parameters . AddWithValue ( "@d2" ,
    TxtAd . Text );
    komut . Parameters . AddWithValue ( "@d3" ,
    TxtSoyad . Text );
    komut . Parameters . AddWithValue ( "@d4" ,
    CmbBrans . Text );
    komut . Parameters . AddWithValue ( "@d5" ,
    MskTC . Text );
    komut . Parameters . AddWithValue ( "@d6" ,
    TxtSifre . Text );

    komut . ExecuteNonQuery ( );
    bgl . baglanti ( ) . Close ( );
    MessageBox . Show ( " Doktor _bilgisi
    .....güncellendi . " , " Bilgi " ,
    MessageBoxButtons . OK ,
    MessageBoxIcon . Information );

    TabloGuncelle ( );
}
```

komutlarında TC kontrolü yapıp sql komutu yazdım. Sql komutta Doktorlar tablosunda doktor adı, doktor soyadı, branş id'yi branş ada göre getirip, doktor TC'yi ve doktor şifreyi Doktorid'ye göre TxtDoktorid'nin yazısını, TxtAd'ın yazısını, TxtSoyad'ın yazısını, CmbBrans'ın yazısını, MskTC'nin yazısını ve TxtSifre'nin yazısını parametre olarak sql bağlantısını yapıp güncelle dedim. Komutu çalıştırdım ve bağlantıyı kapattım. Sonra "Doktor bilgisi güncellendi." şeklinde bilgi mesajı verip doktor tablosunu güncelledim.

Dosya 16: Branş Paneli 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 FrmBransPaneli : Form
    {
        public FrmBransPaneli()
        {
            InitializeComponent();

            SqlBaglantisi bgl = new SqlBaglantisi();

            private void FrmBransPaneli_Load(object sender,
```

```

        EventArgs e)
        {
            DataTable dt = new DataTable();
            SqlDataAdapter da = new SqlDataAdapter("select
            _____Bransid , _BransAd _from _Tbl_Branslar",
            bgl.baglanti());
            da.Fill(dt);
            dataGridView1.DataSource = dt;
        }

        private void TabloGuncelle()
        {
            DataTable dt = new DataTable();
            SqlDataAdapter da = new SqlDataAdapter("select
            _____Bransid , _BransAd _from _Tbl_Branslar",
            bgl.baglanti());
            da.Fill(dt);
            dataGridView1.DataSource = dt;
        }

        private void BtnEkle_Click(object sender,
        EventArgs e)
        {
            SqlCommand komut = new SqlCommand("insert_into
            _____Tbl_Branslar(BransAd)_values(@b1)",
            bgl.baglanti());
            komut.Parameters.AddWithValue("@b1",
            TxtBrans.Text);
            komut.ExecuteNonQuery();
            bgl.baglanti().Close();
            TabloGuncelle();
        }
    
```

```

        MessageBox.Show("Branş_eklendi.", "Bilgi",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
    }

    private void dataGridView1_CellClick(object sender,
        DataGridViewCellEventArgs e)
    {
        int secilen = dataGridView1.SelectedCells[0].
            RowIndex;
        Txtid.Text = dataGridView1.Rows[secilen].
            Cells[0].Value.ToString();
        TxtBrans.Text = dataGridView1.Rows[secilen].
            Cells[1].Value.ToString();
    }

    private void BtnSil_Click(object sender,
        EventArgs e)
    {
        SqlCommand komut = new SqlCommand("delete_from
        Tbl_Branslar_where_Bransid=@b1",
        bgl.baglanti());
        komut.Parameters.AddWithValue("@b1",
        Txtid.Text);
        komut.ExecuteNonQuery();
        bgl.baglanti().Close();
        TabloGuncelle();
        MessageBox.Show("Branş_kaydı_silindi.",
            "Bilgi", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
    }

```

```

        private void BtnGuncelle_Click(object sender,
        EventArgs e)
        {
            SqlCommand komut = new SqlCommand("update
            Tbl_Branslar_set_BransAd=@b1_where
            Bransid=@b2", bgl.baglanti());
            komut.Parameters.AddWithValue("@b1",
            TxtBrans.Text);
            komut.Parameters.AddWithValue("@b2",
            Txttid.Text);
            komut.ExecuteNonQuery();
            bgl.baglanti().Close();
            TabloGuncelle();
            MessageBox.Show("Branş_kaydı_güncellendi.",
            "Bilgi", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        }

        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.

FrmBransPaneli_Load içinde;

```
DataTable dt = new DataTable();  
        SqlDataAdapter da = new SqlDataAdapter("select  
        _____Bransid ,_BransAd_from_ Tbl_Branslar",  
        bgl.baglanti());  
        da.Fill(dt);  
        dataGridView1.DataSource = dt;
```

komutlarında dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta branş id'yi ve branş adı Branşlar tablosundan sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım.

TabloGuncelle fonksiyonu içinde;

```
DataTable dt = new DataTable();  
        SqlDataAdapter da = new SqlDataAdapter("select  
        _____Bransid ,_BransAd_from_ Tbl_Branslar",  
        bgl.baglanti());  
        da.Fill(dt);  
        dataGridView1.DataSource = dt;
```

komutlarında dt isimli veri tablosu oluşturdum. da isimli sql veri uyarlayıcıya sql komutu yazdım. Sql komutta branş id'yi ve branş adı Branşlar tablosundan sql bağlantısını sağlayarak getir dedim. da isimli sql veri uyarlayıcıya dt isimli veri tablosunu doldur dedim. Veri kafesi görünümü 1'in veri kaynağına dt isimli veri tablosunu aktardım.

BtnEkle_Click içinde;

```
SqlCommand komut = new SqlCommand("insert_into  
Tbl_Branslar(BransAd)_values(@b1)", bgl.baglanti());  
        komut.Parameters.AddWithValue("@b1",
```

```

        TxtBrans . Text );
        komut . ExecuteNonQuery ();
        bgl . baglanti (). Close ();
        TabloGuncelle ();
        MessageBox . Show ( " Branş _ eklendi . " , " Bilgi " ,
        MessageBoxButtons . OK ,
        MessageBoxIcon . Information );

```

komutlarında Branşlar tablosuna Branş adını TxtBrans'ın yazısını parametre olarak ve sql bağlantısını sağlayarak ekledim. Komutu çalıştırdım ve sql bağlantısını kapattım. Branşlar tablosunu güncelledim ve "Branş eklendi." şeklinde bilgi mesajı verdirdim.

dataGridView1_CellClick içinde;

```

int secilen = dataGridView1 . SelectedCells [ 0 ] . RowIndex ;
        Txtid . Text = dataGridView1 . Rows [ secilen ] .
        Cells [ 0 ] . Value . ToString ();
        TxtBrans . Text = dataGridView1 . Rows [ secilen ] .
        Cells [ 1 ] . Value . ToString ();

```

komutlarında branş id'yi Txtid'nin yazısına ve branş adı TxtBrans'ın yazısına branş tablosunda bir kez tıklanılan satırın satır indeksini seçilen değişkenine olarak aktardım.

BtnSil_Click içinde;

```

SqlCommand komut = new SqlCommand ( " delete _ from _ Tbl _ Branslar
_where _ Bransid=@b1 " , bgl . baglanti () );
        komut . Parameters . AddWithValue ( " @b1 " ,
        Txtid . Text );
        komut . ExecuteNonQuery ();
        bgl . baglanti (). Close ();
        TabloGuncelle ();
        MessageBox . Show ( " Branş _ kaydı _ silindi . " ,
        " Bilgi " , MessageBoxButtons . OK ,

```

```
MessageBoxIcon.Information);
```

komutlarında Branşlar tablosundan branşı Bransid'ye göre Txtid'nin yazısını parametre olarak ve sql bağlantısını sağlayarak sildim. Komutu çalıştırdım ve sql bağlantısını kapattım. Branşlar tablosunu güncelledim ve "Branş kaydı silindi." şeklinde bilgi mesajı verdirdim.

BtnGuncelle_Click içinde;

```
SqlCommand komut = new SqlCommand("update _Tbl_Branslar _set  
BransAd=@b1_where _Bransid=@b2", bgl.baglanti());
```

```
komut.Parameters.AddWithValue("@b1",  
TxtBrans.Text);
```

```
komut.Parameters.AddWithValue("@b2",  
Txtid.Text);
```

```
komut.ExecuteNonQuery();
```

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

```
TabloGuncelle();
```

```
MessageBox.Show("Branş_kaydı_güncellendi.",  
"Bilgi", MessageBoxButtons.OK,
```

```
MessageBoxIcon.Information);
```

komutlarında Branşlar tablosunda branş adını Bransid'ye göre TxtBrans'ın yazısını ve Txtid'nin yazısını parametre olarak ve sql bağlantısını sağlayarak güncelledim. Komutu çalıştırdım ve sql bağlantısını kapattım. Branşlar tablosunu güncelledim ve "Branş kaydı güncellendi." şeklinde bilgi mesajı verdirdim.

button1_Click içinde;

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

komutunda uygulamadan çık dedim.

Dosya 17: Program.cs dosyası:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Yonetim_Hastane
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.
            SetCompatibleTextRenderingDefault(false);
            Application.Run(new FrmGirisler());
        }
    }
}
```

komutlarında FrmGirisler formunu ilk açılan form yani parogramı açtığımızda veya çalıştırdığımızda ilk karşımıza gelen olarak ayarladım.

3 PROGRAM VERİ TABANI (SQL) ÇALIŞMASININ ANLATIMI

Dosya 1: Sql Veri Tabanı:

```
create database Hastane_Yonetim_ve_Randevu_Sistemi
use Hastane_Yonetim_ve_Randevu_Sistemi

create table Tbl_Yoneticiler(
Yoneticiid tinyint primary key identity(1,1) not null ,
Yoneticitc varchar(11),
Yoneticiad varchar(30),
Yoneticisoyad varchar(20),
Yoneticisifre varchar(10)
)

create table Tbl_Branslar(
Bransid tinyint primary key identity(1,1) not null ,
Yoneticiid tinyint foreign key references
Tbl_Yoneticiler(Yoneticiid),
BransAd varchar(50)
)

create table Tbl_Doktorlar(
Doktorid tinyint primary key identity(1,1) not null ,
Bransid tinyint foreign key references
Tbl_Branslar(Bransid),
Yoneticiid tinyint foreign key references
Tbl_Yoneticiler(Yoneticiid),
DoktorAd varchar(15),
DoktorSoyad varchar(15),
```

```
DoktorTC varchar(11),
DoktorSifre varchar(10),
)

create table Tbl_Sekreterler(
Sekreterid tinyint primary key identity(1,1) not null ,
SekreterAd varchar(30),
SekreterSoyad varchar(20),
SekreterTC varchar(11),
SekreterSifre varchar(10),
)

create table Tbl_Duyurular(
Duyuruid smallint primary key identity(1,1) not null ,
Yoneticiid tinyint foreign key references
Tbl_Yoneticiler(Yoneticiid),
Doktorid tinyint foreign key references
Tbl_Doktorlar(Doktorid),
Sekreterid tinyint foreign key references
Tbl_Sekreterler(Sekreterid),
Duyuru varchar(255)
)

create table Tbl_Hastalar(
Hastaid smallint primary key identity(1,1) not null ,
HastaAd varchar(30),
HastaSoyad varchar(20),
HastaTC varchar(11),
HastaTelefon varchar(15),
HastaSifre varchar(10),
HastaCinsiyet varchar(5)
```



```
)

create table Tbl_Musteri_Hizmetleri(
MusteriHizmetleriid smallint primary key identity(1,1)
not null ,
MusteriHizmetleriAd varchar(30),
MusteriHizmetleriSoyad varchar(20),
MusteriHizmetleriTC varchar(11),
MusteriHizmetleriSifre varchar(10)
)
```

```
create table Tbl_Randevular(
Randevuid int primary key identity(1,1) not null ,
Bransid tinyint foreign key references
Tbl_Branslar(Bransid),
Doktorid tinyint foreign key references
Tbl_Doktorlar(Doktorid),
Hastaid smallint foreign key references
Tbl_Hastalar(Hastaid),
MusteriHizmetleriid smallint foreign key references
Tbl_Musteri_Hizmetleri(MusteriHizmetleriid),
RandevuTarih varchar(10),
RandevuSaat varchar(5),
RandevuDurum bit ,
Sikayet varchar(255)
)
```

Bu dosyadaki komutlarla sql veri tabanını oluşturup kullanıyorum ve başka dosyalarda ve ilgili projede kullanılmak üzere bilgi depolamak, veri silmek, veri güncellemek için veri alt yapısını oluşturdum.

```
create database Hastane_Yonetim_ve_Randevu_Sistemi
```

Bu komutla Hastane_Yonetim_ve_Randevu_Sistemi veri tabanını oluşturdum.

```
use Hastane_Yonetim_ve_Randevu_Sistemi
```

Bu komutla Hastane_Yonetim_ve_Randevu_Sistemi veri tabanını kullan dedim yani verileri ekleme, verileri silme ve verileri güncelleme için yani veriler üzerinden işlem yapmak için kullanıma açtım.

```
create table Tbl_Yoneticiler(  
Yoneticiid tinyint primary key identity(1,1) not null ,  
Yoneticitc varchar(11),  
Yoneticiad varchar(30),  
Yoneticisoyad varchar(20),  
Yoneticisifre varchar(10)  
)
```

Bu komutlarda Yöneticiler tablosunu yönetici id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve birincil anahtar olarak, yönetici TC'yi 11 karakter, yönetici adı 30 karakter, yönetici soyadı 20 karakter ve yönetici şifreyi 10 karakter varchar - string - olarak tanımlayıp oluşturdum.

```
create table Tbl_Branslar(  
Bransid tinyint primary key identity(1,1) not null ,  
Yoneticiid tinyint foreign key references  
Tbl_Yoneticiler(Yoneticiid),  
BransAd varchar(50)  
)
```

Bu komutlarda Branşlar tablosunu branş id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve birincil anahtar olarak, yönetici id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Yöneticiler tablosunda yönetici id'yi referans alarak ve branş adı 50 karakter varchar - string - olarak tanımlayıp oluşturdum.

```
create table Tbl_Doktorlar(  

```

```

Doktorid tinyint primary key identity(1,1) not null ,
Bransid tinyint foreign key references
Tbl_Branslar(Bransid) ,
Yoneticiid tinyint foreign key references
Tbl_Yoneticiler(Yoneticiid) ,
DoktorAd varchar(15) ,
DoktorSoyad varchar(15) ,
DoktorTC varchar(11) ,
DoktorSifre varchar(10) ,
)

```

Bu komutlarda Doktorlar tablosunu doktor id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve birincil anahtar olarak, branş id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Branşlar tablosunda branş id'yi referans alarak, yönetici id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Yöneticiler tablosunda yönetici id'yi referans alarak ve doktor adı 15 karakter, doktor soyadı 15 karakter, doktor TC'si 11 karakter, doktor şifresi 10 karakter varchar - string - olarak tanımlayıp oluşturdum.

```

create table Tbl_Sekreterler(
Sekreterid tinyint primary key identity(1,1) not null ,
SekreterAd varchar(30) ,
SekreterSoyad varchar(20) ,
SekreterTC varchar(11) ,
SekreterSifre varchar(10) ,
)

```

Bu komutlarda Sekreterler tablosunu sekreter id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve birincil anahtar olarak ve sekreter adı 30 karakter, sekreter soyadı 20 karakter, sekreter TC'si 11 karakter, sekreter şifresi 10 karakter varchar - string - olarak tanımlayıp oluşturdum.

```

create table Tbl_Duyurular(
Duyuruid smallint primary key identity(1,1) not null ,
Yoneticiid tinyint foreign key references
Tbl_Yoneticiler(Yoneticiid),
Doktorid tinyint foreign key references
Tbl_Doktorlar(Doktorid),
Sekreterid tinyint foreign key references
Tbl_Sekreterler(Sekreterid),
Duyuru varchar(255)
)

```

Bu komutlarda Duyurular tablosunu duyuru id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve birincil anahtar olarak, yönetici id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Yöneticiler tablosunda yönetici id'yi referans alarak, doktor id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Doktorlar tablosunda doktor id'yi referans alarak, sekreter id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Sekreterler tablosunda sekreter id'yi referans alarak ve duyuruyu 255 karakter varchar - string - olarak tanımlayıp oluşturdum.

```

create table Tbl_Hastalar(
Hastaid smallint primary key identity(1,1) not null ,
HastaAd varchar(30),
HastaSoyad varchar(20),
HastaTC varchar(11),
HastaTelefon varchar(15),
HastaSifre varchar(10),
HastaCinsiyet varchar(5)
)

```

Bu komutlarda Hastalar tablosunu hasta id'yi smallint - -32,768'den 32,767'ye kadar 2 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve

birincil anahtar olarak ve hasta adı 30 karakter, hasta soyadı 20 karakter, hasta TC'si 11 karakter, hasta telefonu 15 karakter, hasta şifresi 10 karakter, hasta cinsiyeti 5 karakter varchar - string - olarak tanımlayıp oluşturdum.

```
create table Tbl_Musteri_Hizmetleri(  
MusteriHizmetleriid smallint primary key identity(1,1)  
not null ,  
MusteriHizmetleriAd varchar(30),  
MusteriHizmetleriSoyad varchar(20),  
MusteriHizmetleriTC varchar(11),  
MusteriHizmetleriSifre varchar(10)  
)
```

Bu komutlarda Müşteri Hizmetleri tablosunu müşteri hizmetleri id'yi smallint - -32,768'den 32,767'ye kadar 2 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılmayacak şekilde ve birincil anahtar olarak ve müşteri hizmetleri adı 30 karakter, müşteri hizmetleri soyadı 20 karakter, müşteri hizmetleri TC'si 11 karakter, müşteri hizmetleri şifresi 10 karakter varchar - string - olarak tanımlayıp oluşturdum.

```
create table Tbl_Randevular(  
Randevuid int primary key identity(1,1) not null ,  
Bransid tinyint foreign key references  
Tbl_Branslar(Bransid),  
Doktorid tinyint foreign key references  
Tbl_Doktorlar(Doktorid),  
Hastaid smallint foreign key references  
Tbl_Hastalar(Hastaid),  
MusteriHizmetleriid smallint foreign key references  
Tbl_Musteri_Hizmetleri(MusteriHizmetleriid),  
RandevuTarih varchar(10),  
RandevuSaat varchar(5),  
RandevuDurum bit ,  
Sikayet varchar(255)
```

)

Bu komutlarda Randevular tablosunu randevu id'yi int - -2,147,483,648'den 2,147,483,647'ye kadar 4 bayt yer kaplayacak şekilde - otomatik birer birer artan boş bırakılamayacak şekilde ve birincil anahtar olarak, branş id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Branşlar tablosunda branş id'yi referans alarak, doktor id'yi tinyint - 0'dan 255 kadar 1 bayt yer kaplayacak şekilde - yabancı anahtar olarak Doktorlar tablosunda doktor id'yi referans alarak, hasta id'yi smallint - -32,768'den 32,767'ye kadar 2 bayt yer kaplayacak şekilde - yabancı anahtar olarak Hastalar tablosunda hasta id'yi referans alarak, müşteri hizmetleri id'yi smallint - -32,768'den 32,767'ye kadar 2 bayt yer kaplayacak şekilde - yabancı anahtar olarak Müşteri Hizmetleri tablosunda müşteri hizmetleri id'yi referans alarak, randevu durumu bit - 1 bayt olarak - ve randevu tarihi 10 karakter, randevu saati 5 karakter, şikayeti 255 karakter varchar - string - olarak tanımlayıp oluşturdum.

4 SONUÇLAR VE ÖNERİLER

Bu proje çalışmasında, hastane bazında yönetici tarafından yönetici, branş, doktor ve sekreter kaydını kaydetmesi, silmesi ve güncellemesi yapılması, hastanın kendini sisteme kaydetmesini sağlaması, hasta ve yöneticinin bilgilerini düzenlemesini sağlaması, sekreter tarafından pasif yani alınmamış randevu açılmasını sağlaması, güncellemesi ve silmesi, hasta ve müşteri hizmetleri tarafından hastanın şikayetini yazabilerek randevu alması ve randevu iptal etmesi, alınmış veya alınmamış randevuları ilgili koşullara göre ayrı ayrı listelemesi, doktor tarafından işi biten randevuları silmesi, sekreter tarafından duyuru oluşturmaları, doktor, sekreter ve yönetici tarafından duyurulara bakması ve sekreter tarafından alınmamış randevu oluştururken doktor ve branşlara bakması gerçekleştirilmiştir.

Proje kapsamında kullanılan formlar için C# ve veri tabanı kayıtları için Sql programlama dili kullanılmıştır. Kullanılan programlar olarak ise formlar için Visual Studio 2017 ve veri tabanı kayıtları için MS SQL SERVER 2014 kullanılmıştır. SQL Server Express veri tabanı kayıtları için, Sql Server Managemet Studio veri tabanı kayıtları düzenlemek için kullanılmıştır.

Veri tabanından veri çekme, veri ekleme, silme ve güncelleme, çoklu yani birden fazla tablodan veri çekme, veri ekleme, silme ve güncelleme tablo işlemleri, formlardaki tabloları ekleme, silme ve güncelleme işlemlerinden sonra otomatik olarak güncellemek, bütün ekleme, silme güncelleme, randevu alma ve randevuyu iptal etme gibi işlemlerinden sonra kullanıcıyı bilgilendirmek , tablolara ilgili verileri çağırmak, tablolardaki verileri bir kez tıklamak koşuluyla ilgili alanlara aktarmak, ilgili butona tıklandığında ilgili formlar arası geçiş yapmak, gözükmemesi gereken formları görünmez yapmak, işlemler bittiğinde programı sonlandırmak ve formlarda şekil, biçim, renk, büyüklük, yazı biçimi, yazı büyüklüğü, yazı rengi, formlardaki yerleşim, nesnelerin ve yazının neyin üzerinde hangi koşulla duracağı gibi görünümü tasarlama işlemleri yapmak programın çalışmasının içindedir.

Bu proje bir hastane sisteminde yönetim işlemleri ve randevu alma veya iptal etme işlemleri için kullanılabilir. Ancak bu projede hastane bazında işlem yapmaktadır. İnsanların

tüm hastanelerden randevu alabileceđi bir randevu sistemi yapılabilir.

5 EKLER

KAYNAKLAR

- [1] <https://docs.microsoft.com/en-us/sql/t-sql/data-types/bit-transact-sql?view=sql-server-2017>
- [2] <https://docs.microsoft.com/en-us/sql/t-sql/data-types/int-bigint-smallint-and-tinyint-transact-sql?view=sql-server-2017>
- [3] <http://selcukgural.com/c-ile-datagridview-uzerinde-veri-listeleme/>
- [4] <https://www.kodlamamerkezi.com/c-net/c-ile-sql-server-veritabanina-kayit-ekleme/>
- [5] <https://www.kodlamamerkezi.com/c-net/c-ile-sql-server-veritabanindan-kayit-silme/>
- [6] <https://www.kodlamamerkezi.com/c-net/c-ile-sql-server-veritabani-kayit-guncelleme-islemleri/>
- [7] <https://sanalkurs.net/c-datagridview-de-secilen-satirlari-textbox-da-gosterme-10584.html>

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Eyüp Furkan ARIKAN
Uyruğu : T.C.
Doğum Yeri ve Tarihi: Afyonkarahisar-05/06/1992
Adres : Ali Kaleli Mah. Menekşe Sok. Gül Evleri Sitesi
: C Blok Dış Kapı No: 2 İç Kapı No: 9
: Çay/Afyonkarahisar

Telefon : 0551 553 99 49
e-mail : eyupfurkanarikan03@gmail.com

EĞİTİM DURUMU

Lisans Öğrenimi : B.Ş.E.Ü. Bilgisayar Mühendisliği Bölümü
Bitirme Yılı :
Lise : Mustafa Hüsni Gemicici Anadolu Lisesi (Anadolu Öğretmen Lisesi)

İŞ DENEYİMLERİ

Yıl :
Kurum :
Stajlar :

İLGİ ALANLARI: Ruby ile Oyun, Php ile Web Tasarımı, C# ile Otomasyon ve SQL ile Veri Tabanları

YABANCI DİLLER:

İngilizce

BELİRTMEK İSTEDİĞİNİZ DİĞER ÖZELLİKLER: