

E GRUBU-2.ÖĞRETİM



DÜZCE ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ

PROJE ADI:ÖĞRENCİ YURTLARI İÇİN OTOMASYON SİSTEMİ

PROJEDE KULLANILAN YAZILIM DİLİ:C#

**PROJEDE KULLANILAN PROGRAMLAR:VİSUAL STUDIO 2017,MICROSOFT SQL
SERVER MANAGEMENT STUDIO 18**

GRUP ÜYELERİ

152120013 ASLINUR FURAT

152120014 HATİCE KAYA

162114037 BEYZA KILIÇ

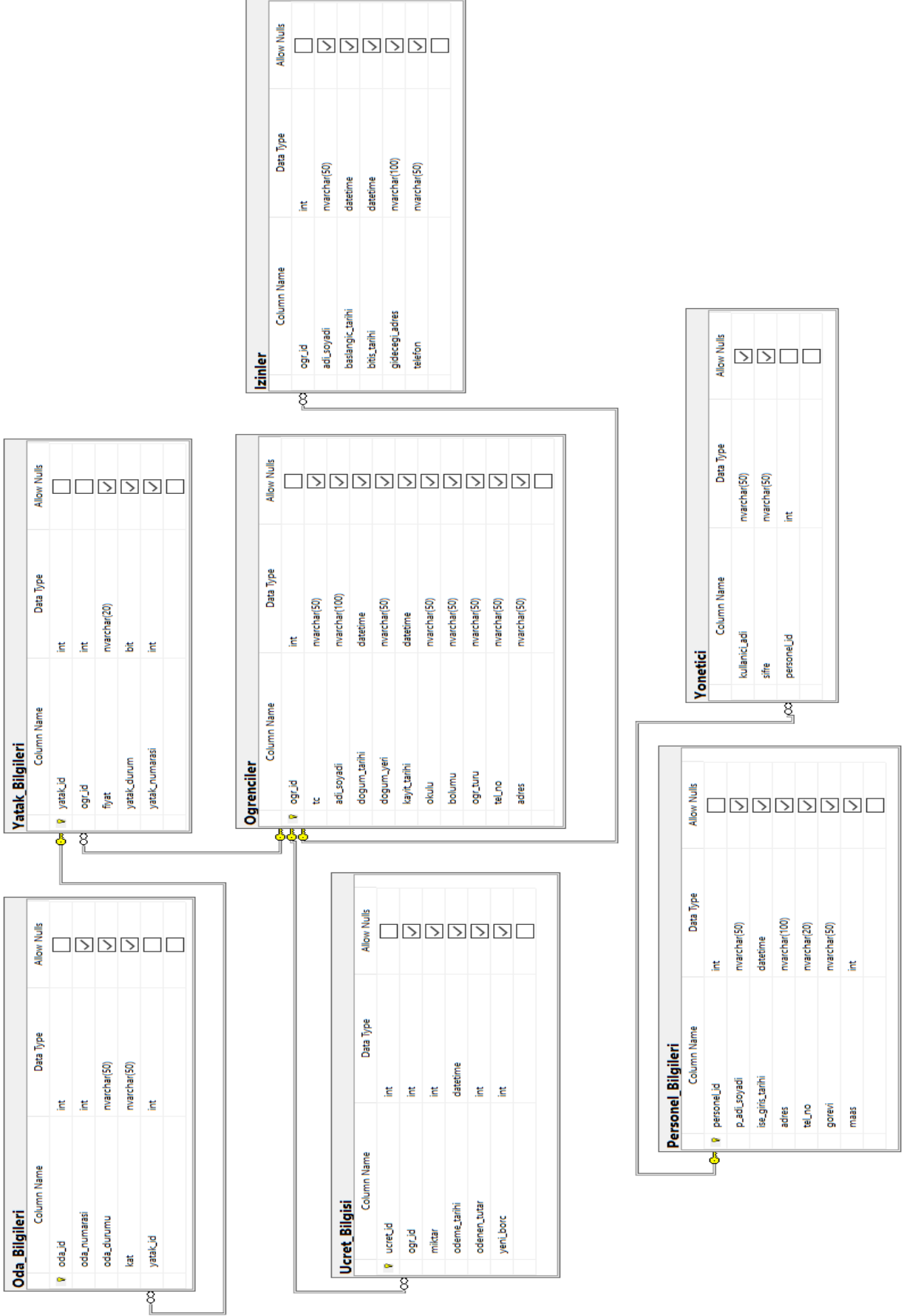
2019-2020 GÜZ YARIYILI

KABULLER

DÜZCE ÖZEL KIZ YURDU;

- Sistem özel bir yurda ait yapılabilecek tüm işlemler için tasarlanmıştır.
- Yurt sistemi 3 katlı, her katta 4 oda olacak şekilde ve her odada 3'er adet yatak olacak şekilde tasarlanmıştır.
- Yurda kayıt işlemi için öğrencinin genel bilgileri, kimlik bilgileri ve oda bilgileri kayıt esnasında Öğrenciler tablosunda yer alır.
- Kayıt olacak öğrencilerin bölümleri, öğrenim türleri belirtilecektir.
- Odaya yerleştirilecek öğrenciler için öncelikle boş olmayan katlardan biri seçilir, katlardaki odalardan dolu olmadığı belirtilen odalardan dolu olmayan yatak öğrenci için seçilebilir.
- Öğrenciler için izin durumu olacaktır ve belirtilen tarihler arasında gideceği adres ve veli telefonu girilerek öğrenci izin kaydı oluşturabilecektir.
- Öğrenciler için ödeme işlemleri en başta kayıt olurken girilen yatak ücreti üzerinden girilen tarihlerde ödenecek miktar kadar kalan ücret üzerinden gerçekleştirilecektir.
- Yurt personelleri için personel kaydı tutulacaktır.
- Yurtta bulunan her yönetici de bir personel olduğundan yönetici kaydı personel tablosunda olmak zorundadır.
- Eğer bir yönetici kaydı oluşturulacaksa öncelikle o yönetici personel olarak eklenmelidir.
- Her yönetici sisteme eklediği kullanıcı adı ve şifresi ile giriş yapabilecektir.

YURT İÇİN OTOMASYON SİSTEMİ ER DİYAGRAMI



VERİTABANI İÇİN SCRIPT KODLARI

```
USE [master]
GO
/***** Object:  Database [YurtOtomasyonDB]    Script Date: 16.12.2019 18:29:40
*****/
CREATE DATABASE [YurtOtomasyonDB]
    CONTAINMENT = NONE
    ON PRIMARY
    ( NAME = N'YurtOtomasyonDB', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL14.SQLEXPRESS\MSSQL\DATA\YurtOtomasyonDB.mdf' , SIZE = 8192KB , MAXSIZE =
UNLIMITED, FILEGROWTH = 65536KB )
    LOG ON
    ( NAME = N'YurtOtomasyonDB_log', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL14.SQLEXPRESS\MSSQL\DATA\YurtOtomasyonDB_log.ldf' , SIZE = 8192KB ,
MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
GO
ALTER DATABASE [YurtOtomasyonDB] SET COMPATIBILITY_LEVEL = 140
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [YurtOtomasyonDB].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [YurtOtomasyonDB] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET ANSI_NULLS OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET ANSI_PADDING OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET ARITHABORT OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [YurtOtomasyonDB] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [YurtOtomasyonDB] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET DISABLE_BROKER
GO
ALTER DATABASE [YurtOtomasyonDB] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [YurtOtomasyonDB] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET HONOR_BROKER_PRIORITY OFF
```

```

GO
ALTER DATABASE [YurtOtomasyonDB] SET RECOVERY SIMPLE
GO
ALTER DATABASE [YurtOtomasyonDB] SET MULTI_USER
GO
ALTER DATABASE [YurtOtomasyonDB] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [YurtOtomasyonDB] SET DB_CHAINING OFF
GO
ALTER DATABASE [YurtOtomasyonDB] SET FILESTREAM( NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [YurtOtomasyonDB] SET TARGET_RECOVERY_TIME = 60 SECONDS
GO
ALTER DATABASE [YurtOtomasyonDB] SET DELAYED_DURABILITY = DISABLED
GO
ALTER DATABASE [YurtOtomasyonDB] SET QUERY_STORE = OFF
GO
USE [YurtOtomasyonDB]
GO
/***** Object: Table [dbo].[Izinler]    Script Date: 16.12.2019 18:29:40 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Izinler](
    [ogr_id] [int] NOT NULL,
    [adi_soyadi] [nvarchar](50) NULL,
    [baslangic_tarihi] [datetime] NULL,
    [bitis_tarihi] [datetime] NULL,
    [gidecegi_adres] [nvarchar](100) NULL,
    [telefon] [nvarchar](50) NULL
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Oda_Bilgileri]    Script Date: 16.12.2019 18:29:40 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Oda_Bilgileri](
    [oda_id] [int] IDENTITY(1,1) NOT NULL,
    [oda_numarasi] [int] NULL,
    [oda_durumu] [nvarchar](50) NULL,
    [kat] [nvarchar](50) NULL,
    [yatak_id] [int] NOT NULL,
    CONSTRAINT [PK_Oda_Bilgileri] PRIMARY KEY CLUSTERED
(
    [oda_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Ogrenciler]    Script Date: 16.12.2019 18:29:40 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Ogrenciler](
    [ogr_id] [int] IDENTITY(1,1) NOT NULL,
    [tc] [nvarchar](50) NULL,
    [adi_soyadi] [nvarchar](100) NULL,
    [dogum_tarihi] [datetime] NULL,
    [dogum_yeri] [nvarchar](50) NULL,
    [kayit_tarihi] [datetime] NULL,
    [okulu] [nvarchar](50) NULL,
    [bolumu] [nvarchar](50) NULL,
    [ogr_turu] [nvarchar](50) NULL,
    [tel_no] [nvarchar](50) NULL,

```

```

        [adres] [nvarchar](50) NULL,
    CONSTRAINT [PK_Ogrenciler] PRIMARY KEY CLUSTERED
(
    [ogr_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Personel_Bilgileri]    Script Date: 16.12.2019 18:29:40
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Personel_Bilgileri](
    [personel_id] [int] IDENTITY(1,1) NOT NULL,
    [p_adi_soyadi] [nvarchar](50) NULL,
    [ise_giris_tarihi] [datetime] NULL,
    [adres] [nvarchar](100) NULL,
    [tel_no] [nvarchar](20) NULL,
    [gorevi] [nvarchar](50) NULL,
    [maas] [int] NULL,
    CONSTRAINT [PK_Personel_Bilgileri] PRIMARY KEY CLUSTERED
(
    [personel_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Ucret_Bilgisi]    Script Date: 16.12.2019 18:29:40
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Ucret_Bilgisi](
    [ucret_id] [int] IDENTITY(1,1) NOT NULL,
    [ogr_id] [int] NULL,
    [miktar] [int] NULL,
    [odeme_tarihi] [datetime] NULL,
    [odenen_tutar] [int] NULL,
    [yeni_borc] [int] NULL,
    CONSTRAINT [PK_Ucret_Bilgisi] PRIMARY KEY CLUSTERED
(
    [ucret_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Yatak_Bilgileri]    Script Date: 16.12.2019 18:29:40
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Yatak_Bilgileri](
    [yatak_id] [int] IDENTITY(1,1) NOT NULL,
    [ogr_id] [int] NOT NULL,
    [fiyat] [nvarchar](20) NULL,
    [yatak_durum] [bit] NULL,
    [yatak_numarasi] [int] NULL,
    CONSTRAINT [PK_Yatak_Bilgileri] PRIMARY KEY CLUSTERED
(
    [yatak_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

```

```
/****** Object: Table [dbo].[Yonetici]    Script Date: 16.12.2019 18:29:40 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Yonetici](
    [kullanici_adi] [nvarchar](50) NULL,
    [sifre] [nvarchar](50) NULL,
    [personel_id] [int] NOT NULL
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[Izinler] WITH CHECK ADD CONSTRAINT [FK_Izinler_Ogrenciler]
FOREIGN KEY([ogr_id])
REFERENCES [dbo].[Ogrenciler] ([ogr_id])
GO
ALTER TABLE [dbo].[Izinler] CHECK CONSTRAINT [FK_Izinler_Ogrenciler]
GO
ALTER TABLE [dbo].[Oda_Bilgileri] WITH CHECK ADD CONSTRAINT
[FK_Oda_Bilgileri_Yatak_Bilgileri] FOREIGN KEY([yatak_id])
REFERENCES [dbo].[Yatak_Bilgileri] ([yatak_id])
GO
ALTER TABLE [dbo].[Oda_Bilgileri] CHECK CONSTRAINT [FK_Oda_Bilgileri_Yatak_Bilgileri]
GO
ALTER TABLE [dbo].[Ucret_Bilgisi] WITH CHECK ADD CONSTRAINT
[FK_Ucret_Bilgisi_Ogrenciler] FOREIGN KEY([ogr_id])
REFERENCES [dbo].[Ogrenciler] ([ogr_id])
GO
ALTER TABLE [dbo].[Ucret_Bilgisi] CHECK CONSTRAINT [FK_Ucret_Bilgisi_Ogrenciler]
GO
ALTER TABLE [dbo].[Yatak_Bilgileri] WITH CHECK ADD CONSTRAINT
[FK_Yatak_Bilgileri_Ogrenciler1] FOREIGN KEY([ogr_id])
REFERENCES [dbo].[Ogrenciler] ([ogr_id])
GO
ALTER TABLE [dbo].[Yatak_Bilgileri] CHECK CONSTRAINT [FK_Yatak_Bilgileri_Ogrenciler1]
GO
ALTER TABLE [dbo].[Yonetici] WITH CHECK ADD CONSTRAINT
[FK_Yönetici_Personel_Bilgileri] FOREIGN KEY([personel_id])
REFERENCES [dbo].[Personel_Bilgileri] ([personel_id])
GO
ALTER TABLE [dbo].[Yonetici] CHECK CONSTRAINT [FK_Yönetici_Personel_Bilgileri]
GO
USE [master]
GO
ALTER DATABASE [YurtOtomasyonDB] SET READ_WRITE
GO
```



1)Sisteme giriş butonu için gerekli kodlar

```
//SİSTEME GİRMEK İÇİN BİR SQL SORGUSU OLUŞTURDUK
DataTable dt = new DataTable();
String sql = "Select * from Yonetici where kullanıcı_adi = '" +
txt_kullanici_adi.Text + "' and sifre='" + txt_sifre.Text + "' ";
//BAĞLANTIMIZI AÇTIK
baglanti.Open();
SqlDataAdapter da = new SqlDataAdapter(sql, baglanti);
da.Fill(dt);
baglanti.Close();
//YÖNETİCİ TABLOSUNDA OLAN KULLANICI ADI VE ŞİFRESİ VARSA SİSTEME GİRİŞ
YAPTIRIYORUZ
if (dt.Rows.Count != 0)
{
    Anasayfa a = new Anasayfa();
    a.Show();
    this.Hide();
}
else
    MessageBox.Show("KULLANICI ADINIZI VE PAROLANIZI YANLIŞ
GİRDİNİZ..!!!");
```

2)Sistemi kapatmak için gerekli kodlar

```
//OTOMASYONU DİREK KAPATMA BUTONU
private void btn_cıkıs_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

3)Girilen şifrenin * ile ya da direkt açık bir şekilde yazılması için gerekli kodlar

```
private void chc_sifre_CheckedChanged(object sender, EventArgs e)
{
    if (chc_sifre.Checked)
    {
        txt_sifre.PasswordChar = '\0';
    }
    else
    {
        txt_sifre.PasswordChar = '*';
    }
}
```




Ana menüden erişilecek sekmeleri açmak için gerekli kodlar

//KAYIT İŞLEMLERİ SAYFASINI AÇMA İŞLEMİNİ YAPIYORUZ

```
private void toolStrip_kayıtislemleri_Click(object sender, EventArgs e)
{
    Kayit_Islemleri a = new Kayit_Islemleri();
    a.Show();
    this.Hide();
}
```

//ÖĞRENCİ İŞLEMLERİ SAYFASINI AÇMA İŞLEMİNİ YAPIYORUZ

```
private void toolStrip_ogrenci_islemleri_Click(object sender, EventArgs e)
{
    Ogrenci_Islemleri a = new Ogrenci_Islemleri();
    a.Show();
    this.Hide();
}
```

//ODALAR SAYFASINI AÇMA İŞLEMİNİ YAPIYORUZ

```
private void toolStrip_odalar_Click(object sender, EventArgs e)
{
    Odalar a = new Odalar();
    a.Show();
    this.Hide();
}
```

//PERSONEL SAYFASINI AÇMA İŞLEMİNİ YAPIYORUZ

```
private void toolStrip_personel_Click(object sender, EventArgs e)
{
    Personel a = new Personel();
    a.Show();
    this.Hide();
}
```

4)Kayıt işlemleri tıklandığında açılan form ve kodları;

Genel Bilgiler

Oda Bilgileri

Tc Kimlik No

Okulu

Adı Soyadı

Bölümü

Doğum Tarihi

16 Aralık 2019 Pazar

Öğrenim Türü

Doğum Yeri

Telefon

Kayıt Tarihi

16 Aralık 2019 Pazar

Adres

İb

Kaydı Tamamla

Kaydı Temizle

Anasayfaya Dön

9

10

11

ogr_id	tc	adi_soyadi	dogum_tarihi	dogum_yeri	kayit_tarihi	okulu	bolumu
1	11111111111	Fatma Yıldız	4.04.2012 00:17	İstanbul	4.09.2019 00:17	Düzce	Bilgisayar M
2	22222222222	Bügra Öztürk	16.01.2007 00:17	Ankara	12.04.2018 00:17	Düzce	Elektrik-Elek

Kayıt işlemlerindeki kaydı tamamla,kayıt temizle ve anasayfaya dön butonları kodları;

9) `private void btn_kayıt_tamamla_Click(object sender, EventArgs e)`

```
{
    // SQL SERVER'A BAĞLANTIMIZI OLUŞTURDUK
    SqlConnection baglanti = new SqlConnection("server=BEYZA\\SQLEXPRESS;
Initial Catalog=YurtOtomasyonDB;Integrated Security=SSPI");
    string sql = "Select ogr_id from Ogrenciler where tc='" + txt_tc.Text +
    """;

    baglanti.Open();
    komut = new SqlCommand(sql, baglanti);
    SqlDataAdapter da2 = new SqlDataAdapter(komut);
    DataTable dt2 = new DataTable();
    //ÖĞRENCİ İD'SİNİ ÇEKTIĞİMİZ SORGUYU TABLOYA DOLDURDUK
    da2.Fill(dt2);
    baglanti.Close();

    if (dt2.Rows.Count == 0)
    {
        // ÖĞRENCİLER TABLOSUNA VERİ EKLEMEK İÇİN BİR SORU YAZDIK
        string sorgu = "INSERT INTO
Ogrenciler(tc,adi_soyadi,dogum_tarihi,dogum_yeri,kayit_tarihi,okulu,bolumu,ogr_turu,te
l_no,adres) VALUES
(@tc,@adi_soyadi,@dogum_tarihi,@dogum_yeri,@kayit_tarihi,@okulu,@bolumu,@ogr_turu,@tel
_no,@adres)";

        //SORGU İÇİN SQL KOMUT OLUŞTURDUK
        komut = new SqlCommand(sorgu, baglanti);
        //YUKARIDAKİ SORGUMUZUN İÇİNDEKİ VERİLERİ AŞAĞIDA DEĞER ATADIK
        komut.Parameters.AddWithValue("@tc", txt_tc.Text);
        komut.Parameters.AddWithValue("@adi_soyadi",
txt_ogrenci_adsoyad.Text);
        komut.Parameters.AddWithValue("@dogum_tarihi",
dtp_dogum_tarihi.Value);
        komut.Parameters.AddWithValue("@dogum_yeri", txt_dogum_yeri.Text);
        komut.Parameters.AddWithValue("@kayit_tarihi",
dtp_kayit_tarihi.Value);
        komut.Parameters.AddWithValue("@okulu", txt_okulu.Text);
        komut.Parameters.AddWithValue("@bolumu", txt_bolumu.Text);
        komut.Parameters.AddWithValue("@ogr_turu", cmb_ogrturu.Text);
        komut.Parameters.AddWithValue("@tel_no", txt_telefon.Text);
        komut.Parameters.AddWithValue("@adres", txt_adres.Text);
        //VERİTABANIMIZA BAĞLANTI AÇTIK
```

```

        baglanti.Open();
        //ÖĞRENCİ EKLEME KOMUTUMUZU ÇALIŞTIRDIK
        komut.ExecuteNonQuery();
        baglanti.Close();

        //YATAK BİLGİLERİ TABLOSUNDA ÖĞRENCİ İD KULLANILDIĞI İÇİN ÖĞRENCİLER
        TABLOSUNDAN ÖĞRENCİ İD'SİNİ ÇEKTİK
        string sorgu1 = "Select ogr_id from Ogrenciler where tc='" +
txt_tc.Text + "'";
        baglanti.Open();
        komut = new SqlCommand(sorgu1, baglanti);
        SqlDataAdapter da = new SqlDataAdapter(komut);
        DataTable dt = new DataTable();
        //ÖĞRENCİ İD'SİNİ ÇEKTİĞİMİZ SORGUYU TABLOYA DOLDURDUK
        da.Fill(dt);
        baglanti.Close();
        //TABLODA BULUNAN ÖĞRENCİ İD SİNİ ÇEKTİK
        int ogrenciid = Convert.ToInt32(dt.Rows[0][0]);
        //YATAK TABLOSUNA VERİ EKLEMEK İÇİN SORU OLUŞTURDUK
        string sorgu2 = "INSERT INTO
Yatak_Bilgileri(ogr_id,fiyat,yatak_numarasi,yatak_durum) VALUES
(@ogr_id,@fiyat,@yatak_numarasi,@yatak_durum)";

        komut = new SqlCommand(sorgu2, baglanti);
        komut.Parameters.AddWithValue("@ogr_id", ogrenciid + "");
        komut.Parameters.AddWithValue("@fiyat", txt_yatak_fiyat.Text);
        komut.Parameters.AddWithValue("@yatak_numarasi", cmb_yatakno.Text);
        komut.Parameters.AddWithValue("@yatak_durum", true);

        baglanti.Open();

        komut.ExecuteNonQuery();
        baglanti.Close();
        //ODA BİLGİLERİ TABLOSUNDA YATAK İD KULLANILDIĞI İÇİN YATAK
        TABLOSUNDAN YATAK İD'SİNİ ÇEKTİK
        string sorgu4 = "Select yatak_id from Yatak_Bilgileri where ogr_id='"
+ ogrenciid + "'";
        baglanti.Open();
        komut = new SqlCommand(sorgu4, baglanti);
        da = new SqlDataAdapter(komut);
        dt = new DataTable();
        da.Fill(dt);
        baglanti.Close();
        int yatakid = Convert.ToInt32(dt.Rows[0][0]);
        //ODA TABLOSUNA VERİ EKLEMEK İÇİN SORU OLUŞTURDUK
        string sorgu3 = "INSERT INTO Oda_Bilgileri(kat,oda_numarasi,yatak_id)
VALUES (@kat,@oda_numarasi,@yatak_id)";

        komut = new SqlCommand(sorgu3, baglanti);

        komut.Parameters.AddWithValue("@kat", cmb_kat.Text);
        komut.Parameters.AddWithValue("@oda_numarasi", cmb_odano.Text);
        komut.Parameters.AddWithValue("@yatak_id", yatakid + "");
        baglanti.Open();

        komut.ExecuteNonQuery();
        baglanti.Close();
        goster();

    }
    else
        MessageBox.Show("Bu Kimlik Numarasına sahip öğrenci
        bulunmaktadır.!!");

}

```

```

10) private void btn_kaydı_temizle_Click(object sender, EventArgs e)
    {
        //YENİ KAYIT OLUŞTURULACAĞI ZAMAN TEXTBOXDAKİ DEĞERLERİ TEMİZLİYORUZ
        txt_tc.Clear();
        txt_ogrenci_adsoyad.Clear();
        txt_dogum_yeri.Clear();
        txt_okulu.Clear();
        txt_bolumu.Clear();
        txt_telefon.Clear();
        txt_adres.Clear();
        txt_yatak_fiyat.Clear();
        txt_id.Clear();
    }

```

```

11) private void btn_anasayfa_Click(object sender, EventArgs e)
    {
        Anasayfa a = new Anasayfa();
        a.Show();
        this.Hide();
    }

```

5)Öğrenci bilgilerini düzenleme, öğrenci silme ve rapor etmek için yazdırma işlemlerini ve öğrencinin ücret ve izin bilgilerine erişme

The screenshot shows a web application for student management. At the top, there is a search bar labeled "Öğrenci Ara" and a button labeled "Anasayfaya Dön". Below the search bar is a table with columns: ogr_id, tc, adi_soyadi, dogum_tarihi, dogum_yeri, kayıt_tarihi, okulu, and b. The table contains two rows of student data. Below the table is a large grey rectangular area labeled "15". At the bottom of the interface, there are several buttons: "İzinler" (labeled 12), "Ücret Bilgisi" (labeled 16), "Öğrenci Düzenle" (labeled 15), "Öğrenci Sil" (labeled 13), and "Tüm Öğrencileri Yazdır" (labeled 14). The text "Seçilen Öğrencinin" is positioned above the "İzinler" and "Ücret Bilgisi" buttons.

ogr_id	tc	adi_soyadi	dogum_tarihi	dogum_yeri	kayıt_tarihi	okulu	b
1	11111111111	Fatma Yıldız	4.04.2012 00:17	İstanbul	4.09.2019 00:17	Düzce	B
2	22222222222	Büğra Öztürk	16.01.2007 00:17	Ankara	12.04.2018 00:17	Düzce	E

12)Öğrenci izin ekleme formu ve kodları

ID	<input type="text" value="1"/>	Telefon	<input type="text" value="() -"/>
Öğrenci TC	<input type="text" value="1111111111"/>	Gideceği Adres	<input type="text"/>
Adı Soyadı	<input type="text" value="Fatma Yıldız"/>		
Başlangıç Tarihi	<input type="text" value="16 Aralık 2019 Pazartesi"/>		
Bitiş Tarihi	<input type="text" value="16 Aralık 2019 Pazartesi"/>	<input type="button" value="İzin Ekle"/>	<input type="button" value="Geri"/>

	ogr_id	tc	adi_soyadi	baslangic_tarihi	bitis_tarihi	gidecegi_adres	telefon
▶	1	1111111111	Fatma Yıldız	16.12.2019 15:02	16.12.2019 15:02		() -
*							

```
private void btn_izin_ekle_Click_1(object sender, EventArgs e)
{
    //İZİN ALAN ÖĞRENCİYİ VERİTABANINA KAYIT EDEN SQL SORGUSUNU YAPIYORUZ
    string sorgu = "INSERT INTO
    Izinler(ogr_id,adi_soyadi,baslangic_tarihi,bitis_tarihi,gidecegi_adres,telefon)
    VALUES(@ogr_id,@adi_soyadi,@baslangic_tarihi,@bitis_tarihi,@gidecegi_adres,@telefon)";

    komut = new SqlCommand(sorgu, baglanti);
    komut.Parameters.AddWithValue("@ogr_id", txt_id.Text);
    komut.Parameters.AddWithValue("@adi_soyadi", txt_adi.Text);
    komut.Parameters.AddWithValue("@baslangic_tarihi",
    dtp_baslangic_tarihi.Value);
    komut.Parameters.AddWithValue("@bitis_tarihi", dtp_bitis_tarihi.Value);
    komut.Parameters.AddWithValue("@gidecegi_adres", txt_Adres.Text);
    komut.Parameters.AddWithValue("@telefon", txt_telefon.Text);

    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
    goster();
}
```

13)Öğrenci silme butonu kodları;

```
private void btn_ogrenci_sil_Click(object sender, EventArgs e)
{
    string sorgu4 = "DELETE FROM Izinler WHERE ogr_id=@ogr_id";
    komut4 = new SqlCommand(sorgu4, baglanti);
    komut4.Parameters.AddWithValue("@ogr_id",
    Convert.ToInt32(dataGridView1.CurrentRow.Cells[0].Value.ToString()));
    baglanti.Open();
    komut4.ExecuteNonQuery();
    baglanti.Close();
    //SEÇİLEN ÖĞRENCİYİ BAĞLI OLDUĞU EN UÇ TABLODAN SİLECEK ŞEKİLDE SQL
    SORGUMUZU YAZIYORUZ
    string sorgu3 = "DELETE FROM Oda_Bilgileri WHERE oda_id=@oda_id";
    komut3 = new SqlCommand(sorgu3, baglanti);
    komut3.Parameters.AddWithValue("@oda_id",
    Convert.ToInt32(dataGridView1.CurrentRow.Cells[16].Value.ToString()));
    baglanti.Open();
    komut3.ExecuteNonQuery();
}
```

```

baglanti.Close();

string sorgu2 = "DELETE FROM Yatak_Bilgileri WHERE yatak_id=@yatak_id";
komut2 = new SqlCommand(sorgu2, baglanti);
komut2.Parameters.AddWithValue("@yatak_id",
Convert.ToInt32(dataGridView1.CurrentRow.Cells[15].Value.ToString()));
baglanti.Open();
komut2.ExecuteNonQuery();
baglanti.Close();

string sorgu = "DELETE FROM Ogrenciler WHERE ogr_id=@ogr_id";
komut = new SqlCommand(sorgu, baglanti);
komut.Parameters.AddWithValue("@ogr_id",
Convert.ToInt32(dataGridView1.CurrentRow.Cells[0].Value.ToString()));
baglanti.Open();
komut.ExecuteNonQuery();
baglanti.Close();
goster();
}

```

14)Öğrencileri Word dosyasına yazdırmak için kodlar;

```

static wordeaktar.Application worduygulama = new wordeaktar.Application();
wordeaktar.Document wordDocument =
worduygulama.Documents.Add(Type.Missing,Type.Missing, Type.Missing, Type.Missing)
public void WordAktar()
{
    //WORD E YAZDIR BUTONUNUN İŞLEMİNİ YAPAN FONKSİYON
    try
    {
        worduygulama.Visible = true;
        wordDocument.PageSetup.PaperSize = wordeaktar.WdPaperSize.wdPaperA4;

        worduygulama.Selection.Font.Bold =
(int)wordeaktar.WdConstants.wdFirst;

        for (int i = 0; i < dataGridView1.Rows.Count; i++)
        {
            for (int j = 0; j < dataGridView1.Columns.Count -1; j++)
            {
                worduygulama.Selection.TypeText(dataGridView1[j,
i].Value.ToString() + " ");
            }
            worduygulama.Selection.TypeParagraph();
        }
        string wyol = @"C:\Users\hp\Desktop\belge1" +
DateTime.Now.ToShortDateString() + ".docx";
        wordDocument.SaveAs(wyol, Type.Missing, Type.Missing, Type.Missing,
Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing,
Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing);
    }
    catch (Exception)
    {
        MessageBox.Show("İşlemleriniz Word Dosyasına Aktarıldı.!!");
    }
}

```

15)Öğrenci güncellemek için tıklandığında açılan form ve kodları;

Genel Bilgiler Oda Bilgileri

Tc Kimlik No Okulu

Adı Soyadı Bölümü

Doğum Tarihi 16 Aralık 2019 Pazar Öğrenim Türü

Doğum Yeri Telefon

Kayıt Tarihi 16 Aralık 2019 Pazar Adres

İd

Güncelle

Öğrenci İşlemlerine Dön

Anasayfaya Dön

ogr_id	tc	adi_soyadi	dogum_tarihi	dogum_yeri	kayit_tarihi	okulu	bolum
1	11111111111	Fatma Yıldız	4.04.2012 00:17	İstanbul	4.09.2019 00:17	Düzce	Bilgis
2	22222222222	Büya Öztürk	16.01.2007 00:17	Ankara	12.04.2018 00:17	Düzce	Elektr

17

18

17)string sorgu = "UPDATE Ogrenciler SET

tc=@tc,adi_soyadi=@adi_soyadi,dogum_tarihi=@dogum_tarihi,dogum_yeri=@dogum_yeri,kayit_tarihi=@kayit_tarihi,okulu=@okulu,bolumu=@bolumu,ogr_turu=@ogr_turu,tel_no=@tel_no,adres=@adres WHERE ogr_id=@ogr_id";

SqlCommand komut = new SqlCommand(sorgu, baglanti);

```
komut.Parameters.AddWithValue("@tc", txt_tc.Text);
komut.Parameters.AddWithValue("@adi_soyadi", txt_ogrenci_adsoyad.Text);
komut.Parameters.AddWithValue("@dogum_tarihi", dtp_dogum_tarihi.Value);
komut.Parameters.AddWithValue("@dogum_yeri", txt_dogum_yeri.Text);
komut.Parameters.AddWithValue("@kayit_tarihi", dtp_kayit_tarihi.Value);
komut.Parameters.AddWithValue("@okulu", txt_okulu.Text);
komut.Parameters.AddWithValue("@bolumu", txt_bolumu.Text);
komut.Parameters.AddWithValue("@ogr_turu", cmb_ogr_turu.Text);
komut.Parameters.AddWithValue("@tel_no", txt_telefon.Text);
komut.Parameters.AddWithValue("@adres", txt_adres.Text);
komut.Parameters.AddWithValue("@ogr_id", txt_id.Text);
```

```
baglanti.Open();
komut.ExecuteNonQuery();
baglanti.Close();
```

string sorgu2 = "UPDATE Yatak_Bilgileri SET fiyat=@fiyat,yatak_numarasi=@yatak_numarasi WHERE ogr_id=@ogr_id";
SqlCommand komut2 = new SqlCommand(sorgu2, baglanti);

```
komut2.Parameters.AddWithValue("@fiyat", txt_yatak_fiyat.Text);
komut2.Parameters.AddWithValue("@yatak_numarasi", cmb_yatakno.Text);
komut2.Parameters.AddWithValue("@ogr_id", txt_id.Text);
baglanti.Open();
komut2.ExecuteNonQuery();
baglanti.Close();
```

string sorgu3 = "UPDATE Oda_Bilgileri SET oda_numarasi=@oda_numarasi,kat=@kat WHERE yatak_id=@yatak_id";
SqlCommand komut3 = new SqlCommand(sorgu3, baglanti);
komut3.Parameters.AddWithValue("@oda_numarasi", cmb_odano.Text);
komut3.Parameters.AddWithValue("@kat", cmb_kat.Text);
komut3.Parameters.AddWithValue("@yatak_id", txt_yatak_id.Text);
baglanti.Open();
komut3.ExecuteNonQuery();
baglanti.Close();
goster();
cmb_odano.Enabled = false;
cmb_yatakno.Enabled = false;

}


```

18) private void btn_geri_Click(object sender, EventArgs e)
    {
        Ogrenci_Islemleri a = new Ogrenci_Islemleri();
        a.Show();
        this.Hide();
    }

```

16)Seçilen öğrencinin ücret bilgisine erişmek ve borç ödeme işlemi için;

	ucret_id	ogr_id	miktar	odeme_tarihi	odenen_tutar
▶	26	1	9500	16.12.2019 23:20	500
	27	1	9500	16.12.2019 23:20	400
	28	4	10000	16.12.2019 23:41	250
*					

19)Öğrenci işlemlerinde seçilen öğrencinin verileri aktarılarak ödeme yapma işlemi ve kodları;

```

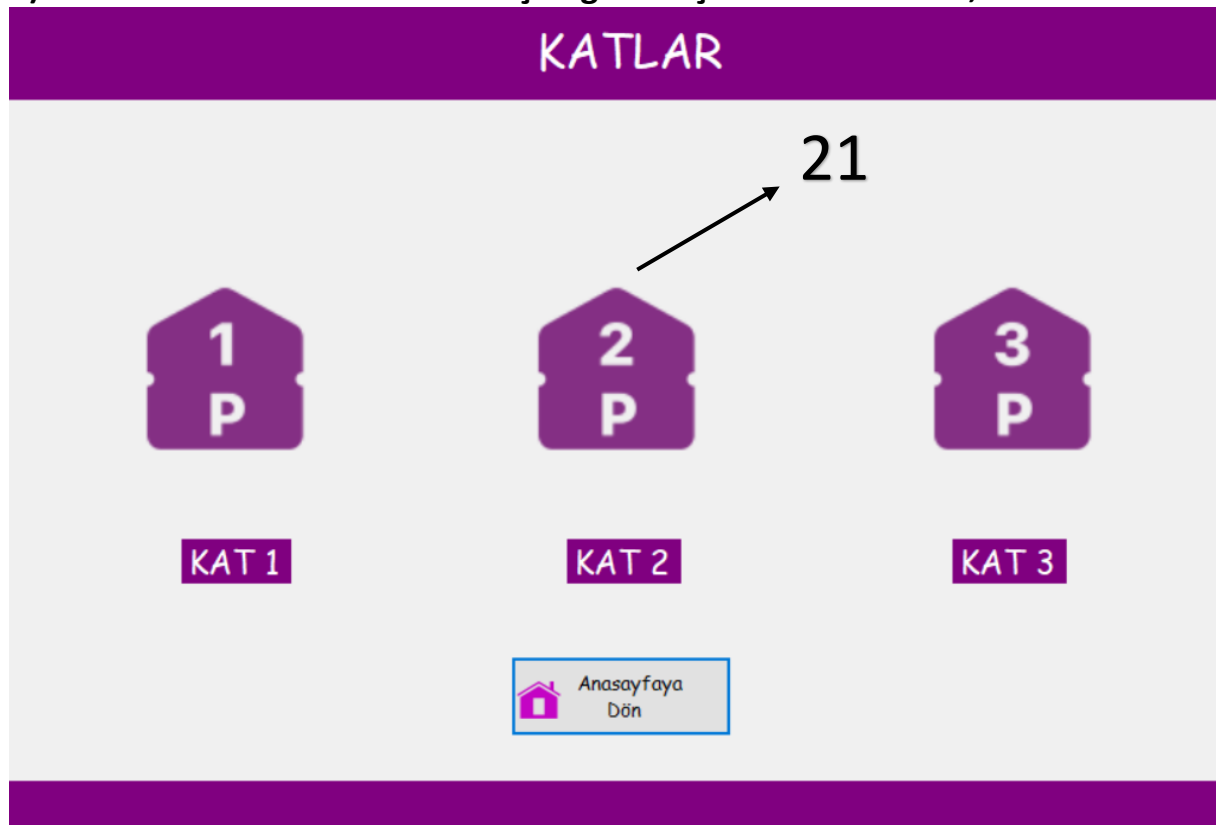
int toplam_tutar = 0;
DataTable dt = new DataTable();
string sorgu2 = "Select sum (odenen_tutar) from Ucret_Bilgisi where
ogr_id='" + txt_id.Text + "'";
baglanti.Open();
SqlDataAdapter da = new SqlDataAdapter(sorgu2, baglanti);
da.Fill(dt);
baglanti.Close();
try
{
    toplam_tutar = Convert.ToInt32(dt.Rows[0][0]);
}
catch (Exception)
{
}
int yeni_borc = Convert.ToInt32(txt_miktar.Text) - toplam_tutar -
Convert.ToInt32(txt_odenen_tutar.Text);
txt_yeni_borc.Text = yeni_borc.ToString();
string sorgu = "INSERT INTO
Ucret_Bilgisi(ogr_id,miktar,odeme_tarihi,odenen_tutar,yeni_borc) VALUES
(@ogr_id,@miktar,@odeme_tarihi,@odenen_tutar,@yeni_borc)";
komut = new SqlCommand(sorgu, baglanti);
komut.Parameters.AddWithValue("@ogr_id", txt_id.Text);
komut.Parameters.AddWithValue("@miktar", txt_miktar.Text);
komut.Parameters.AddWithValue("@odeme_tarihi", dtp_odeme_Tarih.Value);
komut.Parameters.AddWithValue("@odenen_tutar", txt_odenen_tutar.Text);
komut.Parameters.AddWithValue("@yeni_borc", txt_yeni_borc.Text);
baglanti.Open();
komut.ExecuteNonQuery();
baglanti.Close();
goster();
}

```


20)Tablodan ücret bilgilerini silmek için;

```
private void btn_silme_Click(object sender, EventArgs e)
{
    String sql = "delete Ucret_Bilgisi where ucret_id=@ucret_id ";
    komut = new SqlCommand(sql, baglanti);
    komut.Parameters.AddWithValue("@ucret_id",
dataGridView1.CurrentRow.Cells[0].Value.ToString());
    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
    goster();
}
```

6)Ana menü üzerinden Odalar seçildiğinde açılan katlar formu;



//Yurdun 3 katı ve her kata tıklandığında açılacak odalar için gerekli kodlar;

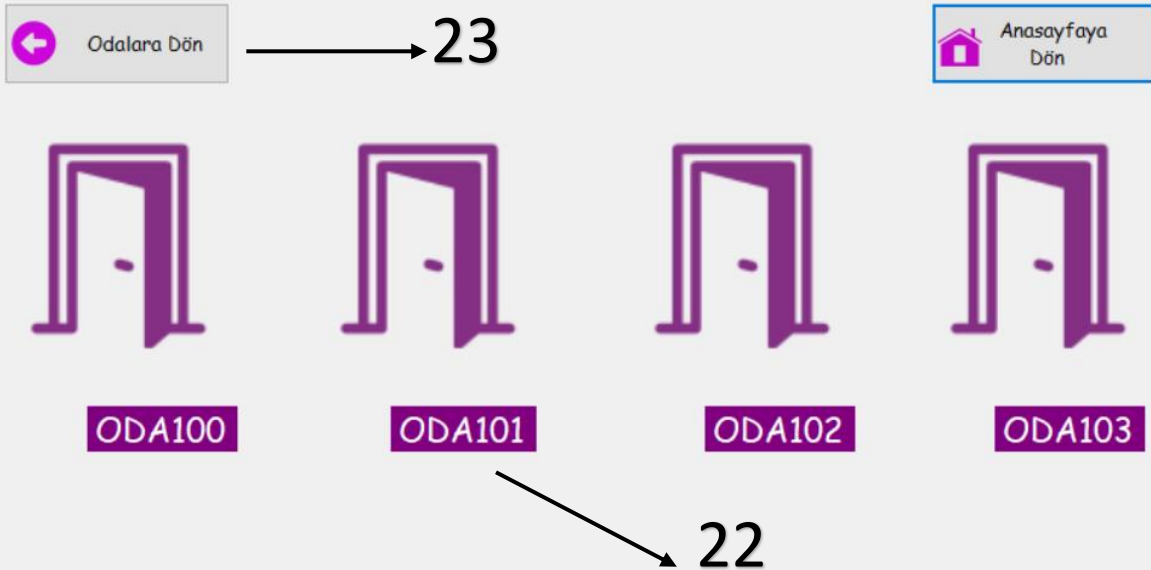
```
private void pictureBox1_Click(object sender, EventArgs e)
{
    Katlardaki_Odalar a = new Katlardaki_Odalar();
    a.bulundugu_kat = 1;
    a.Show();
    this.Hide();
}
private void pictureBox2_Click(object sender, EventArgs e)
{
    Katlardaki_Odalar a = new Katlardaki_Odalar();
    a.bulundugu_kat = 2;
    a.Show();
    this.Hide();
}
private void pictureBox3_Click(object sender, EventArgs e)
{
    Katlardaki_Odalar a = new Katlardaki_Odalar();
    a.bulundugu_kat = 3;
    a.Show();
    this.Hide();
}
```

```

21)//Kat için fotoğraf tıklandığında açılacak odalar formu için gerekli kodlar;
int odasayisi = 4;
int yataksayisi = 3;
int katsayisi = 3;
private void Odalar_Load(object sender, EventArgs e)
{
    int[] a = new int[katsayisi];
    for (int j = 1; j <= katsayisi; j++)
    {
        int doluodasayisi = 0;
        for (int i = 0; i < odasayisi; i++)
        {
            int odanumeralari = j * 100 + i;
            string sorgu = "Select yatak_id from Oda_Bilgileri where
oda_numarasi=" + odanumeralari + "" + " and kat='" + j + "' ";
            baglanti.Open();
            komut = new SqlCommand(sorgu, baglanti);
            SqlDataAdapter da3 = new SqlDataAdapter(komut);
            DataTable dt3 = new DataTable();
            da3.Fill(dt3);
            baglanti.Close();
            if (dt3.Rows.Count == yataksayisi)
            {
                doluodasayisi++;
            }
        }
        if (doluodasayisi == odasayisi)
        {
            a[j-1] = 1;
        }
        else
        {
            a[j-1] = 0;
        }
    }
    //EĞER TÜM KATLAR DOLDUĞUNDA KATLARIN DOLU OLDUĞUNU BELİRTMEK İÇİN ALT LABEL
    RENKLERİNİ BELİRLERİZ
    if (a[0] == 1)
    {
        lbl_kat1.Text = "KAT 1 DOLU";
        lbl_kat1.BackColor = Color.Red;
        lbl_kat1.ForeColor = Color.White;
    }
    else
    {
        lbl_kat1.Text = "KAT 1";
        lbl_kat1.BackColor = Color.Purple;
        lbl_kat1.ForeColor = Color.White;
    }
}

```

ODALAR



```
22) for (int j = 1; j <= katsayisi; j++)
{
    for (int i = 0; i < odasayisi; i++)
    {
        //GELEN KAT DEĞERİNE GÖRE ODA NUMARALARINI BULUYORUZ
        int odanumarlari = j * 100 + i;
        SqlConnection baglanti = new
SqlConnection("server=BEYZA\\SQLEXPRESS; Initial Catalog=YurtOtomasyonDB;Integrated
Security=SSPI");
        string sorgu = "Select yatak_id from Oda_Bilgileri where
oda_numarasi=" + odanumarlari + "" + " and kat='" + j + "' ";
        baglanti.Open();
        komut = new SqlCommand(sorgu, baglanti);
        SqlDataAdapter da = new SqlDataAdapter(komut);
        DataTable dt = new DataTable();
        da.Fill(dt);
        baglanti.Close();
        if (dt.Rows.Count == yataksayisi)
        {
            a[i] = 1;
        }
        else
        {
            a[i] = 0;
        }
    }
}
//ODALAR EĞER DOLARSA LABELİMİZE ODA NUMARASINA GÖRE DOLU YAZDIRIP RENGİNİ
DEĞİŞTİRİYORUZ
//ODALARIMIZ BOŞ İSE RENGİ MOR YAPIYORUZ
if (a[0] == 1)
{
    lbl_oda1.Text = "ODA"+((bulundugu_kat)*100)+ " DOLU";
    lbl_oda1.BackColor = Color.Red;
    lbl_oda1.ForeColor = Color.White;
}
else
{
    lbl_oda1.Text = "ODA" + ((bulundugu_kat ) * 100);
    lbl_oda1.BackColor = Color.Purple;
    lbl_oda1.ForeColor = Color.White;}
```

```

23) //ODALAR FORMUNDAN KATLAR FORMUNA GERİ DÖNÜŞ KODLARI
private void btn_odaya_dön_Click(object sender, EventArgs e)
{
    Odalar a = new Odalar();
    a.Show();
    this.Hide();
}

```

Yatak Öğrenci Bilgisi					
Öğrenci TC	KİŞİ TC	Öğrenci Bölümü	BÖLÜMÜ		
Öğrenci Adı Soyadı	ADI SOYADI	Öğrenci Öğrenim Türü	ÖĞRENİM TÜRÜ		
Öğrenci Kayıt Tarihi	KAYIT TARİHİ	Yatak Ücreti	YATAK ÜCRETİ		

24)Yatak fotoğrafı seçildiğinde o yatak eğer dolu ise bilgiler çekilir;

```

for (int j = 0; j < odasayisi; j++)
{
    //YATAK BİLGİLERİNİ ÇEKTİĞİMİZ VERİTABANI SORGUMUZU YAZIYORUZ
    string sorgu = "Select yatak_numarasi from Yatak_Bilgileri where
yatak_id in( Select yatak_id from Oda_Bilgileri where oda_numarasi=" + oda_numarasi +
" and kat=" + bulundu_kat + " )";
    baglanti.Open();
    komut = new SqlCommand(sorgu, baglanti);

    SqlDataAdapter da = new SqlDataAdapter(komut);
    DataTable dt = new DataTable();
    da.Fill(dt);
    baglanti.Close();
    for (int i = 0; i < dt.Rows.Count; i++)
    {
        a[Convert.ToInt32(dt.Rows[i][0]) - 1] = 1;
    }
}
//DOLU OLAN YATAĞIN RENGİNİ DEĞİŞTİRİYORUZ
if (a[0] == 1)
{
    lbl_yatak1.Text = "YATAK 1 DOLU";
    lbl_yatak1.BackColor = Color.Red;
    lbl_yatak1.ForeColor = Color.White;
}
else

```

```

{
    lbl_yatak1.Text = "YATAK 1";
    lbl_yatak1.BackColor = Color.Purple;
    lbl_yatak1.ForeColor = Color.White;
}

```

25)Yatak dolu ise o yataktaki öğrenci bilgileri listelenir;

```

PictureBox pc = (PictureBox)sender;
//ÜZERİNE GELDİĞİMİZ YATAK RESMİNİN HANGİ ÖĞRENCİYE AİT OLDUĞUNU BİLDİREN
PANELİMİZİ GÖSTERİYORUZ
string sorgu2 = "Select ogr_id from Yatak_Bilgileri where yatak_numarasi
='"+ Convert.ToInt32(pc.Tag) + "' and yatak_id in (Select yatak_id from Oda_Bilgileri
where oda_numarasi=" + oda_numarasi + " and kat='" + bulundu_kat + "')";
baglanti.Open();
komut = new SqlCommand(sorgu2, baglanti);
SqlDataAdapter da2 = new SqlDataAdapter(komut);
DataTable dt2 = new DataTable();
da2.Fill(dt2);
baglanti.Close();
try
{
    ogr_id = Convert.ToInt32(dt2.Rows[0][0]);
}
catch (Exception)
{
}
}
goster();
try
{
    //YATAĞIN ÖĞRENCİYE AİT OLDUĞU PANELİ DOLDURDUĞUMUZ BİLGİLERİ BURADAN
    lKisiTC.Text = dt.Rows[0][1].ToString();
    lAdiSoyadi.Text = dt.Rows[0][2].ToString();
    lKayitTarihi.Text = dt.Rows[0][5].ToString();
    lBolumu.Text = dt.Rows[0][7].ToString();
    lOgrenimTuru.Text = dt.Rows[0][8].ToString();
    lYatakUcreti.Text = dt.Rows[0][14].ToString() + " TL";
}
}

```

26)Yataklar formundan odalar formuna geri dönüş yapmak için gerekli kodlar;

```

private void btn_odaya_dön_Click(object sender, EventArgs e)
{
    Katlardaki_Odalar a = new Katlardaki_Odalar();
    a.Show();
    this.Hide();
}

```

7)Personel kaydı oluşturmak için açılan form;

Personel Kaydı

Personel Kodu	1	Telefonu	(533) 279-6332
Adı Soyadı	Hatice Kaya	Görev	Yönetici
İşe Giriş Tarihi	4 Haziran 2016 Cumartesi	Maaş	7500
Adres	Ankara		

Temizle

Yeni Personel Personel Sil Personel Güncelle Anasayfaya Dön

personel_id	p_adi_soyadi	ise_giris_tarihi	adres	tel_no	gorevi	maas
1	Hatice Kaya	4.06.2016	Ankara	5332796332	Yönetici	7500

27 28 29 30

27)Yeni personel kaydı için gerekli kodlar;

```
private void btn_personel_ekle_Click(object sender, EventArgs e)
{
    //PERSONEL BİLGİLERİ TABLOSUNA PERSONEL EKLEMEK İÇİN SQL SORGUMUZU
    YAZIYORUZ
    string sorgu = "INSERT INTO
Personel_Bilgileri(p_adi_soyadi,ise_giris_tarihi,adres,tel_no,gorevi,maas) VALUES
(@p_adi_soyadi,@ise_giris_tarihi,@adres,@tel_no,@gorevi,@maas)";

    komut = new SqlCommand(sorgu, baglanti);
    komut.Parameters.AddWithValue("@p_adi_soyadi", txt_p_adisoyadi.Text);
    komut.Parameters.AddWithValue("@ise_giris_tarihi",
dtp_p_isegiristarihi.Value);
    komut.Parameters.AddWithValue("@adres", txt_p_adres.Text);
    komut.Parameters.AddWithValue("@tel_no", txt_p_telefon.Text);
    komut.Parameters.AddWithValue("@gorevi", cmb_p_gorev.Text);
    komut.Parameters.AddWithValue("@maas", txt_p_maas.Text);
    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
    personelgoster();
}
```

28)Personel silme kodları;

```
private void btn_personel_sil_Click(object sender, EventArgs e)
{
    //PERSONEL BİLGİLERİ TABLOSUNDAN PERSONEL TABLOLARI SİLMEK İÇİN SQL
    SORGUMUZU YAZIYORUZ
    string sorgu2 = "DELETE FROM Yoneticici WHERE personel_id=@personel_id";
    SqlCommand komut2 = new SqlCommand(sorgu2,baglanti);
    komut2.Parameters.AddWithValue("@personel_id",
    Convert.ToInt32(dataGridView1.CurrentRow.Cells[0].Value.ToString()));
    baglanti.Open();
    komut2.ExecuteNonQuery();
    baglanti.Close();
    string sorgu = "DELETE FROM Personel_Bilgileri WHERE
    personel_id=@personel_id";
    komut = new SqlCommand(sorgu, baglanti);
    komut.Parameters.AddWithValue("@personel_id",
    Convert.ToInt32(txt_pkodu.Text));
    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
    personelgoster();
}
```

29)Seçilen personeli güncellemek için;

```
private void btn_personel_guncelle_Click(object sender, EventArgs e)
{
    //PERSONEL BİLGİLERİ TABLOSUNDAKİ BİLGİLERİ GÜNCELLEME İŞLEMİ YAPIYORUZ
    string sorgu = "UPDATE Personel_Bilgileri SET
    p_adi_soyadi=@p_adi_soyadi,ise_giris_tarihi=@ise_giris_tarihi,adres=@adres,tel_no=@tel_
    _no,gorevi=@gorevi,maas=@maas WHERE personel_id=@personel_id";
    komut = new SqlCommand(sorgu, baglanti);
    komut.Parameters.AddWithValue("@personel_id",
    Convert.ToInt32(txt_pkodu.Text));
    komut.Parameters.AddWithValue("@p_adi_soyadi", txt_p_adisoyadi.Text);
    komut.Parameters.AddWithValue("@ise_giris_tarihi",
    dtp_p_isegiristarihi.Value);
    komut.Parameters.AddWithValue("@adres", txt_p_adres.Text);
    komut.Parameters.AddWithValue("@tel_no", txt_p_telefon.Text);
    komut.Parameters.AddWithValue("@gorevi", cmb_p_gorev.Text);
    komut.Parameters.AddWithValue("@maas", txt_p_maas.Text);

    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
    personelgoster();
}
```

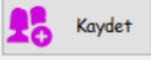
30)Yeni bir personel girişi yapmak için textbox içindeki verileri temizlemek için gerekli kodlar;


```
private void btn_temizle_Click(object sender, EventArgs e)
{
    //YENİ PERSONEL EKLEMEK İÇİN TEXTBOXLARI TEMİZLEME İŞLEMİ YAPIYORUZ
    txt_pkodu.Clear();
    txt_p_adisoyadi.Clear();
    txt_p_adres.Clear();
    txt_p_maas.Clear();
    txt_p_telefon.Clear();
}
```

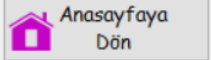
8)Yöneticinin sisteme giriř yapması için oluşturulan form;

	personel_id	p_adi_soyadi	ise_giris_tarihi
▶	1	Hatice Kaya	4.06.2016
*			

	kullanici_adi	sifre	personel_id
▶	hatice	1	1
*			

31 ←  Kaydet

 Sil

 Anasayfaya Dön

32 →

31)Yöneticinin otomasyona giriři için kullanıcı adı ve řifre oluşturarak kaydetmesi için kodlar;

```
private void btn_kaydet_Click(object sender, EventArgs e)
{
    try
    {
        //GİRİLEN PERSONELİN KULLANICI ADI VE PAROLASINI VERİTABANINA EKLEYEN
        SQL SORGUMUZU YAZIYORUZ
        int i =
        Convert.ToInt32(dataGridView1.CurrentRow.Cells[0].Value.ToString());

        string sorgu = "INSERT INTO Yonetici(personel_id,kullanici_adi,sifre)
VALUES (@personel_id,@kullanici_adi,@sifre)";
        SqlCommand komut = new SqlCommand(sorgu, baglanti);
        komut.Parameters.AddWithValue("@personel_id", i);
        komut.Parameters.AddWithValue("@kullanici_adi",
txt_kullanici_adi.Text);
        komut.Parameters.AddWithValue("@sifre", txt_parola.Text);

        baglanti.Open();
        komut.ExecuteNonQuery();
        baglanti.Close();
        goster();
    }
    catch (Exception)
    {
        MessageBox.Show("Lütfen Tablodan Bir Personel Seçiniz..!!");
    }
}
```

32)Yöneticinin kullanıcı adı ve řifresini silme işlemi için;

```
private void btn_sil_Click(object sender, EventArgs e)
{
    //GİRİLEN PERSONELİN İD SİNE GÖRE VERİTABANINDAN SİLEN SQL SORGUMUZU
    YAZIYORUZ
    String sql = "delete Yonetici where personel_id=@personel_id ";
    SqlCommand komut = new SqlCommand(sql, baglanti);

    komut.Parameters.AddWithValue("@personel_id",
dataGridView2.CurrentRow.Cells[2].Value.ToString());
    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
    goster();}
```