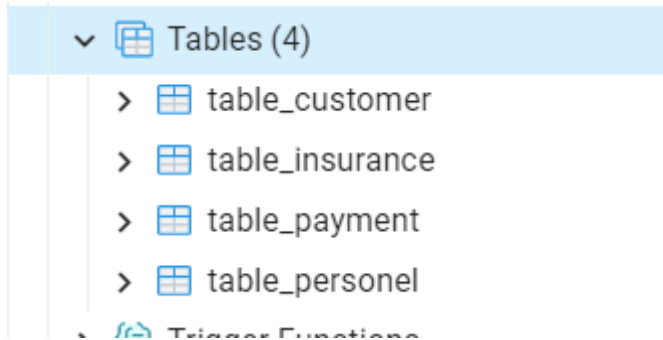
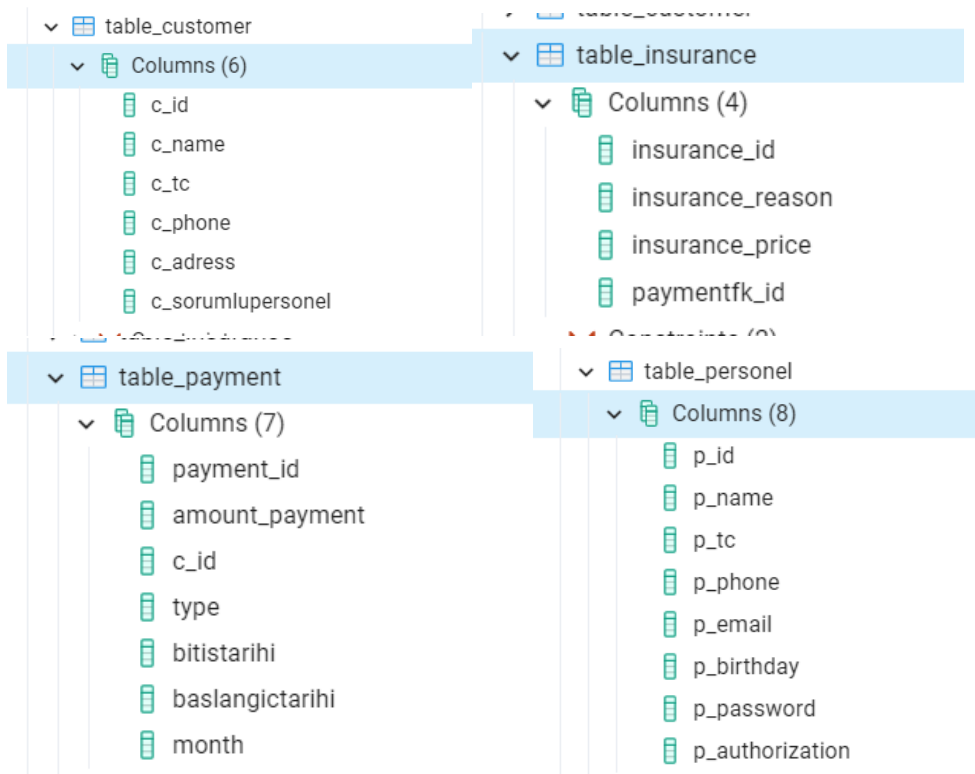


## Sigorta Veritabanı Veritabanı Sistemi

Veritabanını olarak **postgresql** kullandık. Veritabanımızda 4 adet tablomuz bulunmaktadır.



Tablolarımızdaki veriler;



**tablo\_customer** adlı tablodaki **primary key=c\_id**, **Foreign Key=c\_sorumlupersonel** olarak **tablo\_personel** den referans almaktadır.

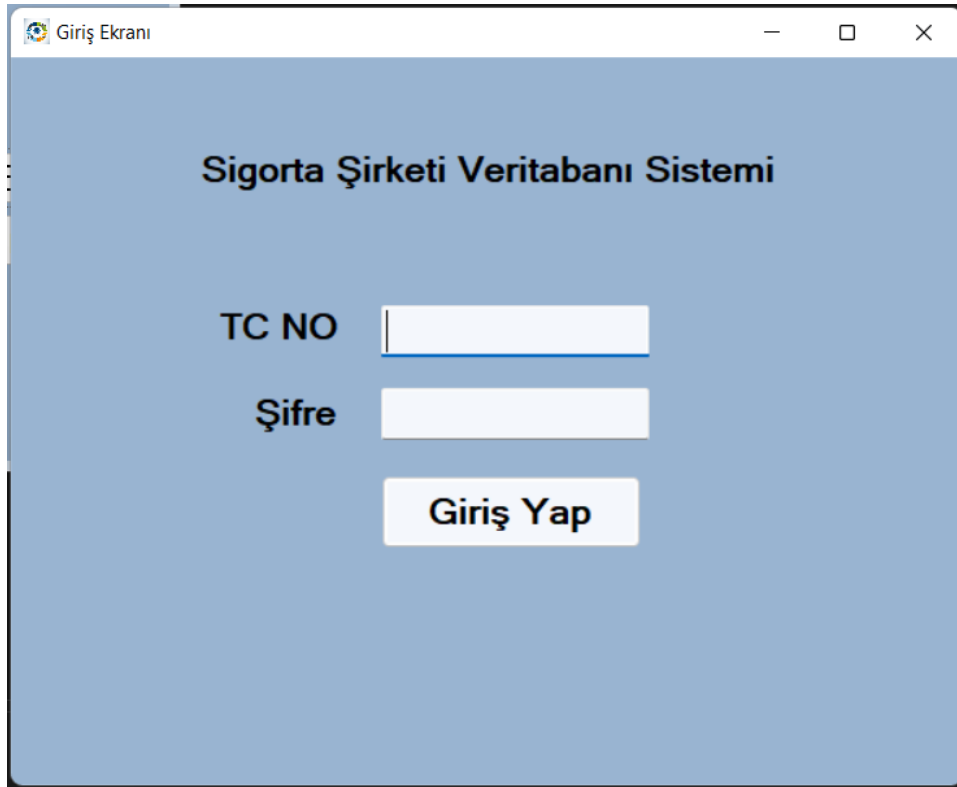
**tablo\_insurance** adlı tablodaki **primary key=insurance\_id**, **Foreign Key=paymentfk\_id** olarak **tablo\_payment** dan referans almaktadır.

**tablo\_payment** adlı tablodaki **primary key =payment\_id**, **Foreign Key=c\_id** olarak **tablo\_customer** dan referans almaktadır.

**tablo\_personel** adlı tablodaki **primary key , p\_id**

Bu primary keyler auto increment olarak 1'er 1'er veri eklendikçe artmaktadır.

Uygulama yapımında **Visual Studio** da **C#** Windows form uygulamasını kullandık.

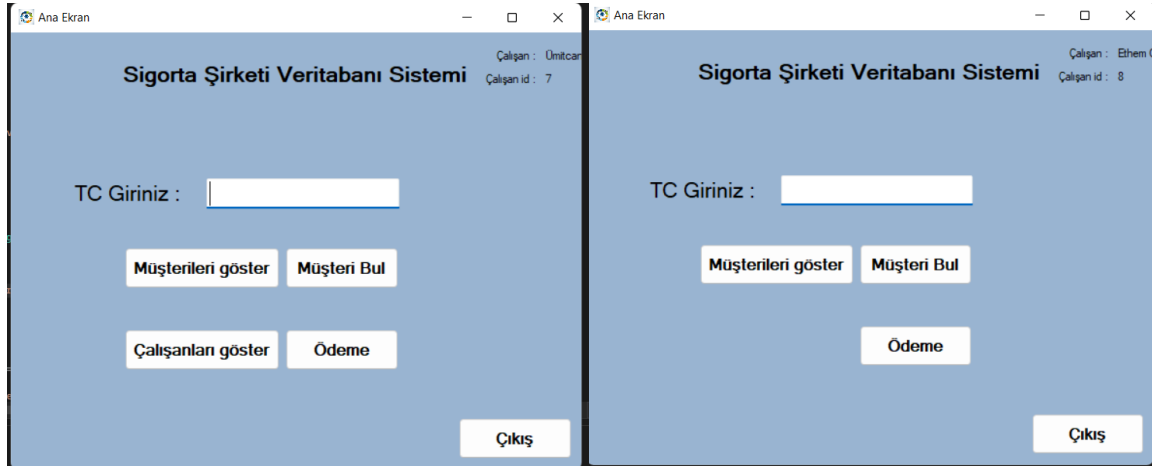


Giriş ekranımızda veritabanında kayıtlı olan **table\_personel** tablosundan Tc\_no ve şifreyi çekerek kontrol sağlıyoruz . Sorgumuz aşağıdaki gibidir.

```
"select * from public.table_personel Where p_tc='" + textBox1.Text + "' and p_password='" + textBox2.Text + "'"
```

Eğer TC ya da şifre boş girilirse veya yanlış girilirse uyarı vermektedir.

Eğer TC ve Şifre doğru ise kullanıcının yetki durumu kontrol edilir Yönetici ise admin sayfasına personel ise çalışan sayfasına yönlendirilir.



Çalışan sayfasında müşteri bilgilerini ekleme,silme,güncelleme ve müşteriye sigorta tarafından ödeme yapılabilirken yönetici sayfasında bunlara ek olarak çalışan ekleme,silme,güncelleme işlemleri yapılabilir ve “Çalışanları Göster” butonu yalnızca yönetici sayfasında görülmektedir.

Müşterileri Göster butonu ile geldiğimiz sayfada müşteri bilgilerini ekleme,silme,güncelleme TC noya göre arama ve yenilenen bilgileri kontrol etmek için yenile butonlarımız bulunmaktadır.Ekleme,silme,güncelleme işlemleri veritabanımızdaki tüm tabloları düzenlemektedir.Çünkü foreign keylerimizi tanımlarken cascade özelliğini aktif ettik.

Ekleme için kullandığımız sorgu;

**“insert into**

**table\_customer(c\_name,c\_tc,c\_phone,c\_adress,c\_sorumlupersonel)VALUES(@c\_name,@c\_tc,@c\_phone,@c\_adress,@c\_sorumlupersonel)”**

ve ekleme sayfasında sigorta başlatma işlemi için kullandığımız sorgu;

**“INSERT INTO**

**table\_payment(c\_id,month,type,amount\_payment,baslangictarihi,bitistarihi)VALUES(@c\_id,@month,@type,@amount,@bas,@bit)”**

Silmek için kullandığımız sorgu;

**DELETE FROM table\_customer WHERE c\_tc=@c\_tc**

Tablolarımız birbiriyle ilişkili olduğu için bu sorgu ile table\_customer dan sildiğimiz veriler aynı zamanda diğer tablolarımızdan da silinecektir.

Güncelleme işlemi için kullandığımız sorgu;

```
"UPDATE table_customer SET  
c_name=@c_name,c_tc=@c_tc,c_phone=@c_phone,c_adress=@c_address  
WHERE c_id=@c_id"
```

ve güncelleme sayfasında diğer verileri de güncellemek için kullandığımız sorgu;

```
"UPDATE table_payment SET  
month=@month,type=@type,amount_payment=@amount WHERE  
c_id=@c_id"
```

TC ye göre müşteri filtrelemek için kullandığımız sorgu;

```
"select  
c_name,c_tc,c_phone,c_adress,c_sorumlupersonel,baslangictarihi,bitistarihi,  
amount_payment from table_customer join table_payment on  
table_customer.c_id=table_payment.c_id WHERE c_tc ='" + txttc.Text+"'" "
```

Tüm müşterileri göstermek için kullandığımız sorgu;

```
"select  
c_name,c_tc,c_phone,c_adress,c_sorumlupersonel,baslangictarihi,bitistarihi,  
amount_payment from table_customer join table_payment on  
table_customer.c_id=table_payment.c_id "
```

Çalışanları Göster butonu ile geldiğimiz sayfada  
ekleme,silme,güncelleme,çalışanları tcye göre filtreleme ve tüm çalışanları  
gösterme işlemleri için sorgularımız;

```
"INSERT INTO  
table_personel(p_name,p_tc,p_phone,p_email,p_birthday,p_password)VALU  
ES(@p_name,@p_tc,@p_phone,@p_email,@p_birthday,@p_password)"
```

```
"DELETE FROM table_personel WHERE p_id=@p_id"
```

```
"UPDATE table_personel SET p_name=@p_name,  
p_tc=@p_tc,p_phone=@p_phone,  
p_email=@p_email,p_birthday=@p_birthday,p_password=@p_password  
WHERE p_id=@p_id"
```

**"SELECT\*FROM table\_personel WHERE p\_tc='"+textBoxtc.Text+""**

**"SELECT \* FROM table\_personel"**

Ödeme butonu ile geldiğimiz sayfada sigorta şirketinin müşteriye sigorta sebebine göre ödeme yapması sağlanır ve şu sorgu ile veritabanını kayıt işlemi gerçekleştirilir;

**"select c\_name from table\_customer join table\_payment on  
table\_customer.c\_id=table\_payment.c\_id join table\_insurance on  
table\_payment.payment\_id=table\_insurance.paymentfk\_id where  
c\_tc=@c\_tc"**

kontrol etmek için kullandığımız sorgu;

**"select  
c\_name,c\_tc,insurance\_reason,insurance\_price,baslangictarihi,bitistarihi  
from table\_customer join table\_payment on  
table\_customer.c\_id=table\_payment.c\_id join table\_insurance on  
table\_payment.payment\_id=table\_insurance.paymentfk\_id"**