

**TÜRK**  
**KIZILAY** BİLGİ SİSTEMİ

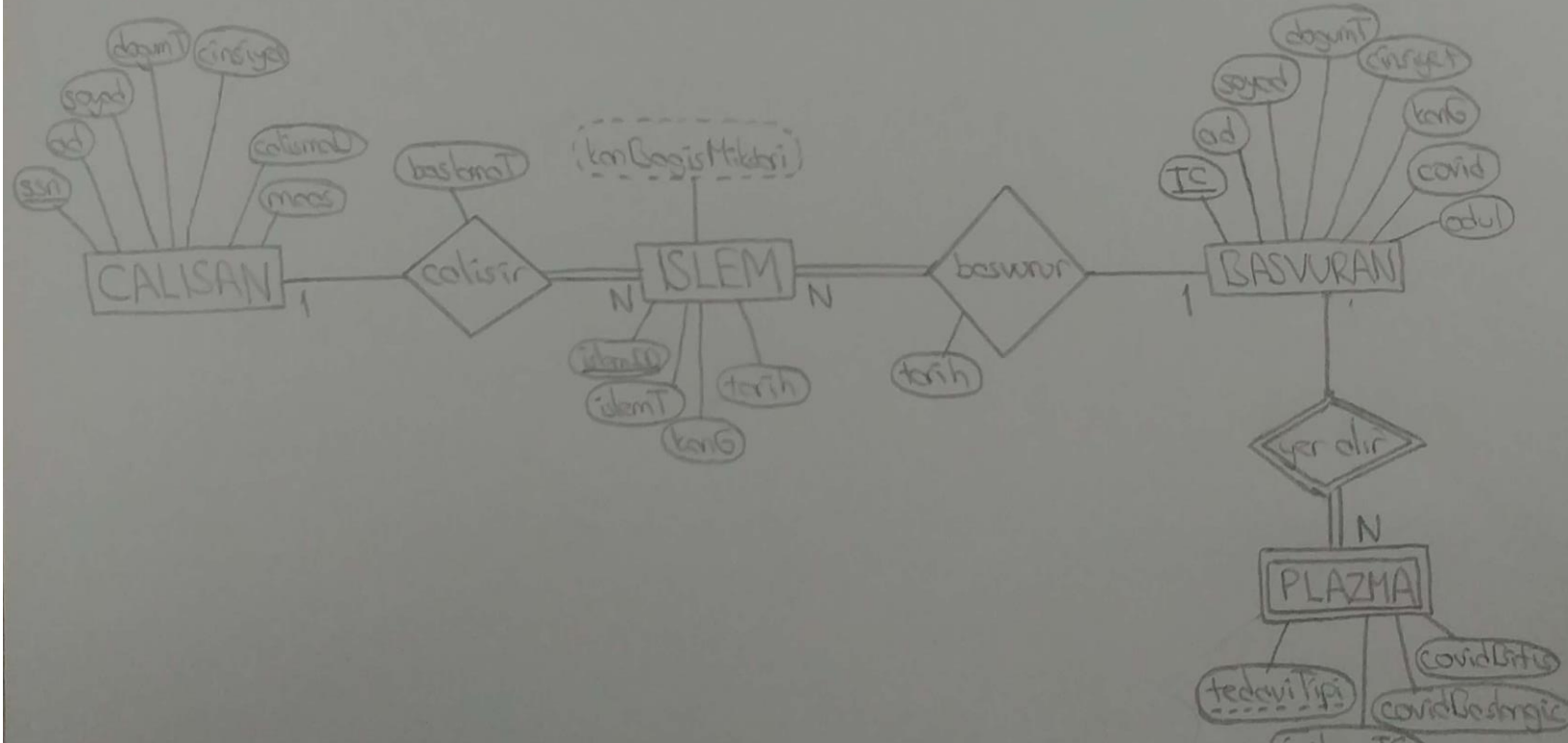
CALISAN (ssn, ad, soyad, dogumT, cinsiyet, baslamaT, calismaD, maaş)

\* Foreign Key

BASVURAN (TC, ad, soyad, dogumT, cinsiyet, kanG, tarih, covid)

ISLEM (islemID, islemTipi, TC\*, kanG, kanAlindiTC\*, ssn\*, tarih)

PLAZMA (TC\*, tedaviTipi, iyilesenTC, covidBaslangic, covidBitis)



ER Şeması

```
1 select * from calisan
```

Data Output Explain Messages Notifications

	ad character varying (20)	soyad character varying (20)	ssn [PK] character (9)	dogumt date	cinsiyet character varying (10)	baslamat date	calismad character varying (10)	salary integer
1	Sevinç	Yener	618665035	1996-07-12	Kadın	2004-01-25	Aktif	5500
2	Kerem	Altun	200634994	1992-04-22	Erkek	2005-11-12	Aktif	4700
3	Ali	Güneş	377665198	1985-12-03	Erkek	2009-07-15	Aktif	6000
4	Pelin	Yıldırım	513025984	1995-10-27	Kadın	2010-12-20	Aktif	4900
5	Eda	Filiz	756321040	1991-08-06	Kadın	2003-04-29	Aktif	5250
6	Merve	Toptaş	892014723	1997-12-26	Kadın	2013-01-17	Aktif	3750
7	Fatih	Sancak	203039445	1990-05-01	Erkek	2008-04-25	Aktif	5000
8	Ada	Yılmaz	852213630	1999-08-10	Kadın	2015-09-13	Aktif	5500
9	Bade	Mutlu	974890165	1993-04-01	Kadın	2006-10-24	Aktif	5200
10	Volkan	Bilir	720354169	1977-10-01	Erkek	1997-09-13	Emekli	[null]

Çalışan Tablosu

Query Editor Query History

1 select \* from basvuran

Data Output Explain Messages Notifications

	ad character varying (25)	soyad character varying (20)	tc [PK] character (11)	dogumt date	cinsiyet character varying (10)	kang character varying (10)	tarih date	covid character varying (10)	odul character varying (20)
1	Belma	Avşar	24156203142	1990-04-22	Kadın	A+	2015-03...	Geçirmedi	[null]
2	Ata	Demirel	21045103621	1995-05-30	Erkek	B-	2018-02...	Geçirmedi	[null]
3	Tarık	Beyaz	22658010511	1996-07-12	Erkek	B-	2019-01...	Geçirmedi	[null]
4	Orhan	Gündoğdu	30231478563	1991-09-01	Erkek	0-	2019-04...	Geçirmedi	[null]
5	Aslı	Sarım	17563021452	1991-01-28	Kadın	A-	2020-01...	Geçirmedi	[null]
6	Evra	Parlak	19852013541	1999-03-29	Kadın	A-	2020-02...	Geçirdi	[null]
7	Zeynep	Dağtaş	31202014502	1999-12-20	Kadın	AB+	2020-08...	Geçirdi	[null]
8	Burak	Sağlam	12103247813	1997-11-02	Erkek	B+	2020-09...	Geçirdi	[null]
9	Salih	Veran	14203973190	1997-02-08	Erkek	AB-	2020-10...	Geçirdi	[null]
10	Mustafa	Birlik	31047913546	1990-09-12	Erkek	0+	2020-12...	Geçirdi	[null]
11	Ugur	Altındal	23515231930	1957-04-05	Erkek	A+	2020-07...	Geçirdi	[null]
12	Onur	Porsuk	23515231931	1986-04-05	Erkek	0+	2020-08...	Geçirdi	[null]
13	Aras	Kaplan	23515231932	2001-04-05	Erkek	0-	2020-03...	Geçirmedi	Bronz Madalya
14	Furkan	Çakmak	23566231932	1980-04-05	Erkek	B+	2021-01...	Geçirdi	Altın Madalya
15	İdris	Demir	23566231966	1993-04-05	Erkek	AB+	2021-01...	Geçirdi	Gümüş Madalya

Başvuran Tablosu

Data Output		Explain	Messages	Notifications				
	islemid [PK] smallint	islemtipi character varying (10)	islemtc character (11)	kangr character varying (10)	kanaldigitc character (11)	issn character (9)	tarih date	
1	1	Kan Verme	14203973190	AB-	[null]	756321040	2019-01...	
2	2	Kan Verme	14203973190	AB-	[null]	756321040	2019-01...	
3	3	Kan Verme	14203973190	AB-	[null]	756321040	2019-01...	
4	4	Kan Verme	14203973190	AB-	[null]	756321040	2019-01...	
5	5	Kan Verme	14203973190	AB-	[null]	756321040	2019-01...	
6	6	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
7	7	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
8	8	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
9	9	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
10	10	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
11	11	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
12	12	Kan Verme	31202014502	AB+	[null]	200634994	2020-08...	
13	13	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
14	14	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
15	15	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
16	16	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
17	17	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
18	18	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
19	19	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
20	20	Kan Verme	23515231931	0+	[null]	756321040	2020-01...	
21	21	Kan Verme	23515231932	0-	[null]	756321040	2019-09...	

İşlem Tablosu

Data Output Explain Messages Notifications

	ptc [PK] character (11)	tedavitipi [PK] character varying (30)	iyilesentc [PK] character (11)	covidbas date	covidbit date	
1	19852013541	Bağışıklık Sistemi	15426412041	2020-01-10	2020-01-21	
2	19852013541	Kan Plazma Seviye Arttırımı	15426412041	2020-01-10	2020-01-21	
3	31202014502	Kan Değerleri Testi	30231478563	2020-08-01	2020-08-10	
4	31202014502	İmmün Plazma COVID-19	22658010511	2020-08-01	2020-08-10	
5	31202014502	Bağışıklık Sistemi	21045103621	2020-08-01	2020-08-10	
6	12103247813	İmmün Plazma COVID-19	24156203142	2020-09-03	2020-09-08	
7	14203973190	İmmün Plazma COVID-19	15426412041	2020-10-05	2020-10-14	
8	14203973190	Kan Plazma Seviye Arttırımı	10023407901	2020-10-05	2020-10-14	
9	31047913546	Kan Plazma Seviye Arttırımı	10023407901	2020-11-24	2020-11-29	
10	31047913546	Dolaşım Sistemi	10023407901	2020-11-24	2020-11-29	

Plazma Tablosu



# Maddelerin sağlandığını gösteren kod blokları

```
CREATE TABLE ISLEM (  
  islemID      smallint,  
  islemTipi    varchar(10) not null,  
  islemTC      char(11),  
  kanGr        varchar(10) not null,  
  kanAldigiTC  char(11),  
  issn         char(9),  
  tarih        date not null,  
  primary key (islemID),  
  foreign key (islemTC) references BASVURAN(TC) ON DELETE NO ACTION ON UPDATE NO ACTION,  
  foreign key (kanAldigiTC) references BASVURAN(TC) ON DELETE NO ACTION ON UPDATE NO ACTION,  
  foreign key (issn) references CALISAN(ssn) ON DELETE NO ACTION ON UPDATE CASCADE  
);
```

Primary - Foreign Key ve  
silme kısıtlamaları

```
CREATE TABLE PLAZMA (  
  pTC          char(11),  
  tedaviTipi   varchar(30),  
  iyilesenTC    char(11),  
  covidBas      date not null,  
  covidBit      date not null,  
  primary key (pTC,tedaviTipi,iyilesenTC),  
  foreign key (pTC) references BASVURAN(TC) ON DELETE NO ACTION ON UPDATE NO ACTION,  
  CHECK ((covidBas)>('2020-01-01'))  
);
```

Sayı kısıtı

```

JButton btnNewButton_2 = new JButton("Insert");
btnNewButton_2.setFont(new Font("Dialog", Font.BOLD, 18));
btnNewButton_2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String temp="";
        temp="insert into islem values (";
        temp+="nextval('seq'),";
        if(textField_1.getText().length()!=0) {
            temp+=""+textField_1.getText()+"",";
        }
        if(textField_2.getText().length()!=0) {
            temp+=""+textField_2.getText()+"",";
        }
        if(textField_3.getText().length()!=0) {
            temp+=""+textField_3.getText()+"",";
        }
        if(textField_4.getText().length()!=0) {
            temp+=""+textField_4.getText()+"",";
        }
        else
            temp+="null",";
        if(textField_5.getText().length()!=0) {
            temp+=""+textField_5.getText()+"",";
        }
        if(textField_6.getText().length()!=0) {
            temp+=""+textField_6.getText()+"",";
        }
        temp+=")";
        java.sql.Statement s=null;
        try {
            s = conn.createStatement();
            s.executeQuery(temp);
            conn.setAutoCommit(false);
            conn.commit();

            s.close();

```

```

JButton btnNewButton_1 = new JButton("Delete");
btnNewButton_1.setFont(new Font("Dialog", Font.BOLD, 18));
btnNewButton_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String temp="DELETE FROM islem ";
        boolean flag=false;
        String and=" AND ";
        String query="";
        if(textField_0.getText().length()!=0) {
            query+="islemid="+textField_0.getText()+"",;
            flag=true;
        }
        if(textField_1.getText().length()!=0) {
            if(flag==true)
                query+=and;
            query+="islemtipi="+textField_1.getText()+"",;
            flag=true;
        }
        if(textField_2.getText().length()!=0) {
            if(flag==true)
                query+=and;
            query+="islemtc="+textField_2.getText()+"",;
            flag=true;
        }
        if(textField_3.getText().length()!=0) {
            if(flag==true)
                query+=and;
            query+="kangr="+textField_3.getText()+"",;
            flag=true;
        }
        if(textField_4.getText().length()!=0) {
            if(flag==true)
                query+=and;

```

```

JButton btnNewButton_3 = new JButton("UPDATE");
btnNewButton_3.setFont(new Font("Dialog", Font.BOLD, 18));
btnNewButton_3.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String query = "UPDATE islem SET ";
        boolean flag = false;
        if(textField_7.getText().length()!=0) {
            query += (fields.get(0)+"="+textField_7.getText()+"",);
            flag=true;
        }
        if(textField_8.getText().length()!=0) {
            query += (fields.get(1)+"="+textField_8.getText()+"",);
            flag=true;
        }
        if(textField_9.getText().length()!=0) {
            query += (fields.get(2)+"="+textField_9.getText()+"",);
            flag=true;
        }
        if(textField_10.getText().length()!=0) {
            query += (fields.get(3)+"="+textField_10.getText()+"",);
            flag=true;
        }
        if(textField_11.getText().length()!=0) {
            query += (fields.get(4)+"="+textField_11.getText()+"",);
            flag=true;
        }
        if(textField_12.getText().length()!=0) {
            query += (fields.get(5)+"="+textField_12.getText()+"",);
            flag=true;
        }
        if(textField_13.getText().length()!=0) {

```

Insert, Delete ve Update işlemlerinin sağlandığı kod blokları



Arayüzden  
girilen tarih  
aralığına  
göre sorguyu  
döndürme

```
Object[] columnNames = {"İslem ID", "İslem Tipi", "İslem TC", "Kan Grubu", "Kan Aldigi T  
islem = new DefaultTableModel(columnNames, 0);  
JTable table = new JTable(islem);  
JScrollPane sp = new JScrollPane(table);  
sp.setBounds(28, 37, 850, 312);  
contentPane.add(sp);  
  
String query;  
query = "SELECT * FROM islem WHERE tarih > '" + textField.getText() + "'" + " and " + "t  
  
if(choice.getSelectedIndex()==0)  
    query += " and islemtipi='Kan Verme'";  
else  
    query += " and islemtipi='Kan Alma'";  
  
java.sql.Statement s;  
try {  
    s = conn.createStatement();  
  
    ResultSet r=s.executeQuery(query);  
  
    while(r.next()) {  
        String islemid = r.getString(1);  
        String islemtipi = r.getString(2);  
        String islemtc = r.getString(3);  
        String kangr = r.getString(4);  
        String kanaldigitc = r.getString(5);
```

```
CREATE VIEW COVIDLILER AS --Çağırma: SELECT * FROM COVIDLILER
SELECT TC, ad, soyad
FROM BASVURAN
WHERE covid='Geçirdi' AND
dogumT > (CURRENT_DATE - INTERVAL '65' YEAR );
```

Kompleks Sorgular

Ad	Soyad	TC
12103247813	Burak	Sağlam
14203973190	Salih	Veran
17563021452	Aslı	Sarım
19852013541	Evra	Parlak
23515231930	Ugur	Altındal
23515231931	Onur	Porsuk
23515231932	Aras	Kaplan
23515233131	Ömer	Altındal
23566231932	Furkan	Çakmak
23566231966	İdris	Demir
31047913546	Mustafa	Birlik
31202014502	Zeynep	Dağtaş

Covid geçirmiş genç-yetişkin veya 2020 yılında Kızılay'a başvurmuş kişilerin ad, soyad ve kimlik numaralarını listeleyiniz.

Seç

Sorgu Getir

X

View  
Sorgusu

# Union

Kompleks Sorgular

Ad	Soyad	TC
12103247813	Burak	Sağlam
14203973190	Salih	Veran
17563021452	Aslı	Sarım
19852013541	Evra	Parlak
23515231930	Ugur	Altındal
23515231931	Onur	Porsuk
23515231932	Aras	Kaplan
23515233131	Omer	Altındal
23566231932	Furkan	Çakmak
23566231966	Idris	Demir
31047913546	Mustafa	Birlik
31202014502	Zeynep	Dağtaş

Covid geçirmiş genç-yetişkin veya 2020 yılında Kızılay'a başvurmuş kişilerin ad, soyad ve kimlik numaralarını listeleyiniz.

Seç

Sorgu Getir

X

# Sequence

```
CREATE SEQUENCE seq
start with 1
increment by 1
;
```

```
(nextval('seq'),'Kan Verme','23566231966','AB+',null,'852213630','2021-01-01');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Verme','23566231966','AB+',null,'852213630','2021-01-01');
```

```
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','19852013541','A-','17563021452','852213630','2020-02-16');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','19852013541','A-','17563021452','852213630','2020-02-16');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','19852013541','A-','17563021452','852213630','2020-02-16');
```

```
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','12103247813','B+','22658010511','852213630','2020-09-14');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','12103247813','B+','22658010511','852213630','2020-09-14');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','12103247813','B+','22658010511','852213630','2020-09-14');
```

```
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','31047913546','B-','22658010511','852213630','2019-01-25');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','31047913546','B-','22658010511','852213630','2019-01-25');
INSERT INTO ISLEM VALUES
(nextval('seq'),'Kan Alma','31047913546','B-','22658010511','852213630','2019-01-25');
```

```
String query="SELECT TC, ad, soyad FROM COVIDLILER UNION SELECT TC, ad, soyad FROM BASVURAN WHERE tarih>'2020-01-01' and tarih<'2021-01-01'";
```

# Aggregate fonksiyonları ve Having

```
String query;
```

```
query = "select b.* from basvuran b, islem i where b.tc=i.islemtc and islemtipi='Kan Verme' group by b.tc having count(*)>9 order by count(*) desc";
```

Kompleks Sorgular

— □ ×

Kan Grubu	Unite Sayisi
AB+	48

Kan ünite bulma fonksiyonu çıktısı  
ve kan ünite kontrol fonksiyonu

```
create or replace function kankontrol()
returns trigger as $$
DECLARE
unite integer;
begin
select unitebul(new.kangr) into unite;
IF new.islemtipi = 'Kan Alma' THEN
    IF unite>0 THEN
        return new;
    ELSE
        RAISE NOTICE ' % : Kan grubu stokta yok..',new.kangr;
        return NULL;
    END IF;
ELSE
return new;
END IF;
END
$$
language plpgsql;

create trigger kantrig
before insert on islem
for each row
execute procedure kankontrol();
```

**Kan kontrol fonksiyonu**

X Kan Grubundaki Ünite Sayısı (Boş Bırakılırsa Bütün Kan Grupları Gösterilecek)

X Kan Grubunu Giriniz

AB+

Seç

Sorgu Getir

X



## Fonksiyonda Cursor kullanımı

```
create function unitebul(kangrubu varchar(10) )
returns integer as $$
declare
toplaml integer;
islem_cur CURSOR FOR SELECT * from islem where kangr=kangrubu;
BEGIN
toplaml:=0;
for satir in islem_cur loop
if satir.islemtipi='Kan Verme' then
toplaml:=toplaml+1;
elsif satir.islemtipi='Kan Alma' then
toplaml := toplaml-1;
end if;
end loop;
return toplaml;
end;
$$
language plpgsql;
```

## Fonksiyonda Record kullanımı

```
create type yeni AS (ad VARCHAR(25),soyad varchar(20),tc char(11),sayac integer);
```

### Record

```
create or replace function odulkontrol()
returns trigger AS $$
DECLARE
bilgi yeni;
BEGIN

select ad,soyad,tc, count(islemtipi) into bilgi
from basvuran b, islem i
where b.tc=i.islemtc and islemtipi='Kan Verme' and islemtc=new.islemtc group by tc;

IF bilgi.sayac=10 THEN
update basvuran set odul='Bronz Madalya' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 10 kez bağış yaparak BRONZ MADALYA kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
ELSIF bilgi.sayac=25 THEN
update basvuran set odul='Gümüş Madalya' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 25 kez bağış yaparak GÜMÜŞ MADALYA kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
ELSIF bilgi.sayac=35 THEN
update basvuran set odul='Altın Madalya' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 35 kez bağış yaparak ALTIN MADALYA kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
ELSIF bilgi.sayac=45 THEN
update basvuran set odul='Plaket' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 45 kez bağış yaparak PLAKET kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
else
return new;
END IF;
END
$$
language plpgsql;
```

```
create or replace function odulkontrol()
returns trigger AS $$
DECLARE
bilgi yeni;
BEGIN

select ad,soyad,tc, count(islemtipi) into bilgi
from basvuran b, islem i
where b.tc=i.islemtc and islemtipi='Kan Verme' and islemtc=new.islemtc group by tc;
```

## Trigger fonksiyonu

```
IF bilgi.sayac=10 THEN
update basvuran set odul='Bronz Madalya' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 10 kez bağış yaparak BRONZ MADALYA kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
ELSIF bilgi.sayac=25 THEN
update basvuran set odul='Gümüş Madalya' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 25 kez bağış yaparak GÜMÜŞ MADALYA kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
ELSIF bilgi.sayac=35 THEN
update basvuran set odul='Altın Madalya' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 35 kez bağış yaparak ALTIN MADALYA kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
ELSIF bilgi.sayac=45 THEN
update basvuran set odul='Plaket' where tc=new.islemtc;
RAISE NOTICE ' Ad : % Soyad : % TC : % vatandaşımız 45 kez bağış yaparak PLAKET kazanmıştır.',bilgi.ad,bilgi.soyad,bilgi.tc;
return new;
else
return new;
END IF;
END
$$
language plpgsql;
```

```
create trigger odultrig
after insert on islem
for each row
execute procedure odulkontrol();
```

## Trigger

Ödül sistemi için madalya kontrol ve yaş kontrol triggerları

```
create or replace function yaskontrol()
returns trigger
as
$$
begin
IF extract(YEAR FROM age(current_date,NEW.dogumt))<18 THEN
RAISE NOTICE'18 yasından kucuk bireyler kan bagisinda bulunamaz';
RETURN NULL;
ELSIF extract(YEAR FROM age(current_date,NEW.dogumt))>65 THEN
RAISE NOTICE '65 yasından buyuk bireyler kan bagisinda bulunamaz';
RETURN NULL;
ELSE
RETURN NEW;
END IF;
END
$$
language plpgsql;
```

## Trigger fonksiyonu

```
create trigger yastrig
before insert on basvuran
for each row
execute procedure yaskontrol();
```

## Trigger



# Hazırlayanlar

## Grup 1

- Ömer Aras Kaplan – 17011039
- Ömer Buğrahan Çalışkan – 17011076
- Uğur Altındal – 17011043
- Onur Porsuk - 17011038