

Bilgisayar Mühendisliği Programı  
Veritabanı Yönetimi Dersi 2021/1 Dönemi  
Dönem Projesi

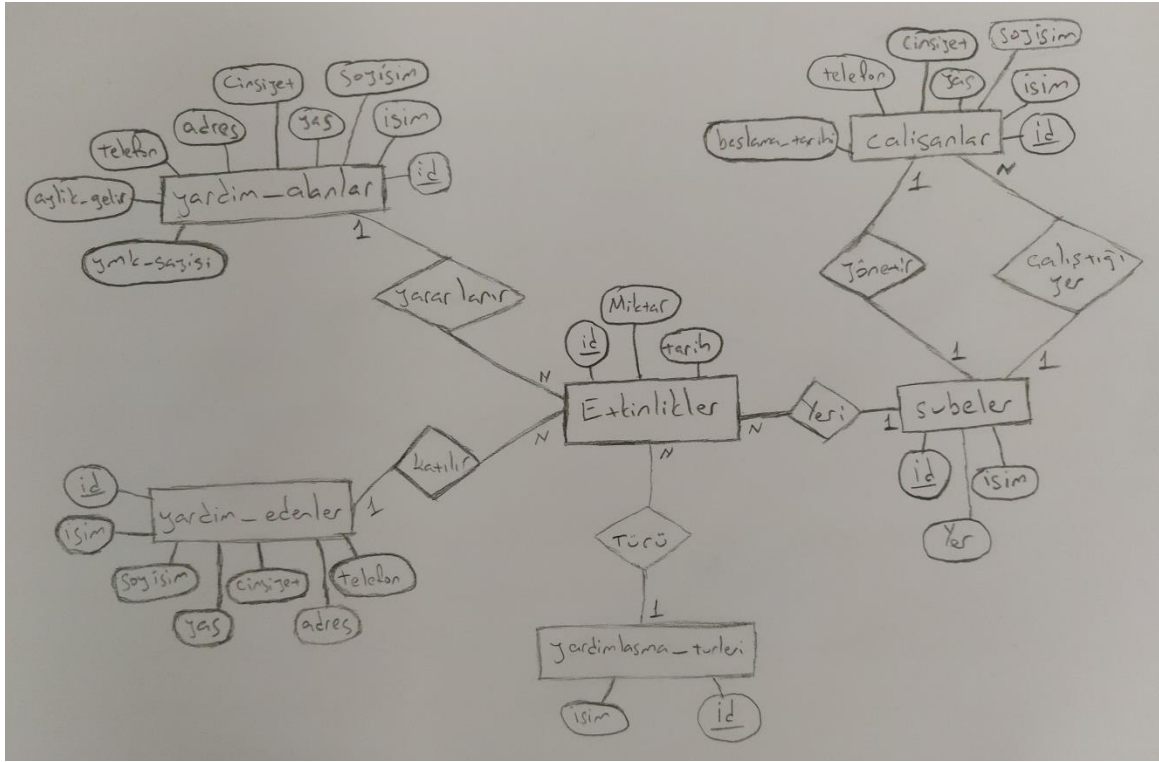
**Yardımlaşma Derneği Veritabanı**

Hazırlayanlar  
Hüseyin Enes Yılmaz, 18011097  
Ömer Talha Baysan, 18011103  
Tuğrul Hasan Karabulut, 17058055

14 Ocak 2022









Dr. Öğr. Üyesi Mustafa Utku KALAY  
Veritabanı Yönetimi

## Tasarım (ER Diyagramı)



## Tablolar

-Çalışanlar tablosu:

	 id [PK] integer	 isim character varying (100)	 soyisim character varying (100)	 yas smallint	 cinsiyet character (1)	 telefon character varying (10)	 sube_id integer	 baslama_tarihi date
1	1	Mukaddes	Suluova	57	K	5351712832	1	2018-01-30
2	2	Gülşen	Atasayar	35	E	5351232222	4	2019-01-22
3	3	Nail	Aşkın	32	E	5381814222	2	2018-04-12
4	4	Çağrı	Sözer	27	E	5341516272	3	2017-03-20
5	5	Yaşar Gökhan	Belder	55	E	5451452256	2	2021-05-03
6	6	Orgül Derya	Gözügül	55	K	5397168242	6	2020-05-30
7	7	İzzettin	Yaşit	22	E	5354517242	8	2020-10-11
8	8	Cem Yaşar	Sarı	33	E	5332415222	4	2020-09-15
9	9	Hilal	Numan	30	K	5345673672	5	2015-12-05
10	10	Ceyhun	Anadol	45	E	5381312582	3	2014-07-10
11	11	Fatma Betül	Morgül	62	K	5331412582	9	2013-07-14
12	12	Cebrail	Tekeli	68	E	5341312582	6	2012-04-11
13	13	Serdar Bora	Süveran	55	E	5451312582	7	2014-09-12
14	14	Onur Kadir	Kuşçu	65	E	5451312582	10	2014-02-22
15	15	Benay	Akkuş	48	K	5451312582	8	2013-01-23

-Etkinlikler tablosu:

	id [PK] integer	sube_id integer	yardimlasma_tur_id integer	yardim_eden_id integer	yardim_alan_id integer	miktar numeric (10,2)	tarih date
1	1	1	1	2	3	500.00	2021-10-13
2	2	1	2	4	2	1.00	2021-08-15
3	3	2	3	6	3	2.00	2021-11-15
4	4	2	4	8	1	3.00	2021-08-15
5	5	3	5	1	4	3.00	2021-10-15
6	6	3	6	3	5	2.00	2021-12-15
7	7	4	7	5	6	1.00	2021-05-23
8	8	4	8	7	7	1.00	2021-07-14
9	9	5	9	10	1	1.00	2021-12-15
10	10	5	10	1	3	5.00	2021-04-21
11	11	6	1	1	6	1520.00	2021-04-15
12	12	6	2	3	8	1.00	2021-08-18
13	13	7	3	3	1	2.00	2021-04-11
14	14	7	4	2	2	4.00	2021-01-15
15	15	8	5	3	4	3.00	2021-04-13

-Şubeler tablosu:

	id [PK] integer	isim character varying (50)	yer character varying (50)	yonetici_id integer
1	1	Çınar Şubesi	İstanbul	1
2	2	Davutpaşa Şubesi	İstanbul	5
3	3	Menemen Şubesi	İzmir	10
4	4	Esentepe Şubesi	İzmir	2
5	5	Çankaya Şubesi	Ankara	16
6	6	Beyoğlu Şubesi	İstanbul	6
7	7	Buca Şubesi	İzmir	18
8	8	Polatlı Şubesi	Ankara	7
9	9	Suadiye Şubesi	İstanbul	11
10	10	Merkez Şubesi	İstanbul	14

### -Yardım alanlar tablosu:

	id [PK] integer	isim character varying (100)	soyisim character varying (100)	yas smallint	cinsiyet character (1)	adres character varying (200)	telefon character varying (10)	aylik_gelir integer	ymk_sayisi smallint
1	1	Yıldırım	Işık	19	E	Merkez Mah. Bağcılar/İstanbul	5354112222	1500	2
2	2	Özge	Yetiz	33	K	Merkez Mah. Karşıyaka/İzmir	5361232522	2000	4
3	3	Akın	Müftüoğlu	57	E	Suadiye Mah. Kadıköy/İstanbul	5381516222	1200	2
4	4	Çağla	Özaydın	66	K	Asmalı Mescit Mah. Beyoğlu/İstanbul	5341516272	750	2
5	5	Sancar	Demirtaştan	44	E	Yıldıztepe Mah. Altındağ/Ankara	5451412256	4000	4
6	6	Şamil	Özcıtal	54	E	Göktürk Mah. Çankaya/Ankara	5394168242	2500	2
7	7	Sedef	Akkutay	28	K	Esentepe Mah. Polatlı/Ankara	5314587242	1100	2
8	8	Muhlis	Yakar	37	E	Esentepe Mah. Aliağa/İzmir	5302515222	3600	5
9	9	Serhat	İlerigiden	41	E	Atatürk Mah. Menemen/İzmir	5345672222	1200	3
10	10	Elif Nur	Akbaş	42	K	Şirince Mah. Selçuk/İzmir	5331412582	1550	3

### -Yardım edenler tablosu:

	id [PK] integer	isim character varying (100)	soyisim character varying (100)	yas smallint	cinsiyet character (1)	adres character varying (200)	telefon character varying (10)
1	1	Ahmet	Kara	25	E	Çifte Havuzlar Mah. Esenler/İstanbul	5351112222
2	2	Mehmet	Uzun	45	E	Caferağa Mah. Kadıköy/İstanbul	5361232222
3	3	Elif	Kara	32	K	Çınar Mah. Bağcılar/İstanbul	5381516222
4	4	Beyza	Demir	19	K	Merkez Mah. Şişli/İstanbul	5341516272
5	5	Tuğrul Hasan	Karabulut	23	E	Akpınar Mah. Çankaya/Ankara	5451412256
6	6	Muhammed	Oğuz	55	E	Göktürk Mah. Çankaya/Ankara	5394168242
7	7	Hüseyin Enes	Yılmaz	22	E	İnönü Mah. Yenimahalle/Ankara	5314517242
8	8	Ömer Talha	Baysan	22	E	29 Ekim Mah. Buca/İzmir	5302415222
9	9	Tuğba	Demir	40	K	Birlik Mah. Bornova/İzmir	5345672222
10	10	Ayşe	Yıldız	60	K	Cumhuriyet Mah. Karşıyaka/İzmir	5331412582

### -Yardımlaşma türleri tablosu:

	id [PK] integer	isim character varying (50)
1	1	para
2	2	ayakkabı
3	3	mont
4	4	kazak
5	5	battaniye
6	6	kömür
7	7	kitap
8	8	erzak
9	9	barınma
10	10	pantolon

## Madde 2:

```
CREATE TABLE subeler (  
    id SERIAL PRIMARY KEY,  
    isim VARCHAR(50) UNIQUE NOT NULL,  
    yer VARCHAR(50) NOT NULL,  
    yonetici_id INT UNIQUE,  
    FOREIGN KEY (yonetici_id) REFERENCES calisanlar(id)  
);
```

## Madde 3:

```
CREATE TABLE etkinlikler (  
    id INT PRIMARY KEY, -- burayı serial yapılmadı, sequence kullanıldı  
    sube_id INT NOT NULL,  
    yardimlasma_tur_id INT NOT NULL,  
    yardim_eden_id INT NOT NULL,  
    yardim_alan_id INT NOT NULL,  
    miktar NUMERIC(10, 2) NOT NULL,  
    tarih DATE NOT NULL,  
    CHECK (miktar > 0),  
    FOREIGN KEY (sube_id) REFERENCES subeler(id), -- Silme kısıtı  
    FOREIGN KEY (yardimlasma_tur_id) REFERENCES yardimlasma_turleri(id), -- Silme kısıtı  
    FOREIGN KEY (yardim_eden_id) REFERENCES yardim_edenler(id), -- Silme kısıtı  
    FOREIGN KEY (yardim_alan_id) REFERENCES yardim_alanlar(id) -- Silme kısıtı  
);
```

```
CREATE TABLE calisanlar (  
    id SERIAL PRIMARY KEY,  
    isim VARCHAR(100) NOT NULL,  
    soyisim VARCHAR(100) NOT NULL,  
    yas SMALLINT NOT NULL,  
    cinsiyet CHAR(1) NOT NULL,  
    telefon VARCHAR(10) NOT NULL,  
    sube_id INT,  
    baslama_tarihi DATE NOT NULL,  
    CHECK (yas < 150), -- Sayı kısıtı  
    CHECK (cinsiyet in ('E', 'K'))  
);
```

## Madde 4:

--Etkinlikler tablosuna kayıt eklenir.

```
INSERT INTO etkinlikler (id, sube_id, yardim_eden_id, yardim_alan_id, yardimlasma_tur_id, miktar,  
tarih) VALUES(nextval('etkinlikler_seq'), @sube_id, @yardim_eden_id, @yardim_alan_id,  
@yardimlasma_tur_id, @miktar, @tarih::date) ;
```

--Id'si verilen yardım alan kişinin aylık geliri güncellenir.

```
UPDATE yardim_alanlar SET aylık_gelir=@aylık_gelir WHERE id=@yardim_alan_id
```

--Çalışanlar tablosundan id'si verilen çalışan silinir.  
DELETE FROM calisanlar WHERE id=@param;

### Madde 5:

--İsim, soyisim, cinsiyet sütunları ile filitrelendirilir.  
SELECT \* FROM yaram\_alanlar  
WHERE LOWER(isim) LIKE @param1  
AND LOWER(soyisim) LIKE @param2  
AND cinsiyet LIKE @param3;

### Madde 6:

-- yardım sayısı birden çok olan kişiler: view  
CREATE or REPLACE VIEW yaram\_sayisi AS  
SELECT yaram\_edenler.id, CONCAT(yaram\_edenler.isim, ' ', yaram\_edenler.soyisim), COUNT(\*)  
FROM yaram\_edenler  
INNER JOIN etkinlikler  
ON yaram\_edenler.id = etkinlikler.yaram\_eden\_id  
GROUP BY yaram\_edenler.id, CONCAT(yaram\_edenler.isim, ' ', yaram\_edenler.soyisim)  
having count(\*)>1

### Madde 7:

CREATE SEQUENCE etkinlikler\_seq  
INCREMENT 1  
START 1;  
  
INSERT INTO etkinlikler (id, sube\_id, yaram\_eden\_id, yaram\_alan\_id, yaramlasma\_tur\_id, miktar, tarih) VALUES(nextval('etkinlikler\_seq'), @sube\_id, @yaram\_eden\_id, @yaram\_alan\_id, @yaramlasma\_tur\_id, @miktar, @tarih::date) ;

### Madde 8:

-- fonksiyon 1 intersect - bu buna yardım ediyor mu?  
create or replace function kim\_kime\_yaram\_ediyor(y\_isim varchar, a\_isim varchar)  
returns boolean as \$\$  
declare sonuc int;  
begin  
select count(\*) into sonuc from (select a.isim, y.isim from yaram\_edenler y  
inner join etkinlikler e on y.id=e.yaram\_eden\_id  
inner join yaram\_alanlar a on a.id=e.yaram\_alan\_id  
where y.isim=y\_isim  
intersect  
select a.isim, y.isim from yaram\_edenler y  
inner join etkinlikler e on y.id=e.yaram\_eden\_id  
inner join yaram\_alanlar a on a.id=e.yaram\_alan\_id  
where a.isim=a\_isim) tablo;

```
        return sonuc;
end;
$$ language 'plpgsql';
--select kim_kime_yardim_ediyor ('Mehmet', 'Akın');
```

### Madde 9:

```
--Altıncı maddedeki arayüzden çağrılan viewde kullanılmaktadır
--Birden çok yardım verenler.
SELECT yardım_edenler.id, CONCAT(yardım_edenler.isim, ' ', yardım_edenler.soyisim), COUNT(*)
FROM yardım_edenler
INNER JOIN etkinlikler
ON yardım_edenler.id = etkinlikler.yardım_eden_id
GROUP BY yardım_edenler.id, CONCAT(yardım_edenler.isim, ' ', yardım_edenler.soyisim)
HAVING count(*)>1
```

### Madde 10:

```
-- fonksiyon 1 intersect - bu buna yardım ediyor mu?
create or replace function kim_kime_yardim_ediyor(y_isim varchar, a_isim varchar)
returns boolean as $$
declare
sonuc int;
begin
    select count(*) into sonuc from (select a.isim, y.isim from yardım_edenler y
    inner join etkinlikler e on y.id=e.yardım_eden_id
    inner join yardım_alanlar a on a.id=e.yardım_alan_id
    where lower(y.isim)=lower(y_isim)
    intersect
    select a.isim, y.isim from yardım_edenler y
    inner join etkinlikler e on y.id=e.yardım_eden_id
    inner join yardım_alanlar a on a.id=e.yardım_alan_id
    where lower(a.isim)=lower(a_isim)) tablo;
    return sonuc;
end;
$$ language 'plpgsql';
select kim_kime_yardim_ediyor('Mehmet', 'akın');
```

```
--fonksiyon 2 - record
CREATE TYPE yeni_tur AS (sube_ismi VARCHAR, calisan_sayisi INT);

CREATE OR REPLACE FUNCTION sube_calisan_sayisi (sube_ismi VARCHAR)
RETURNS yeni_tur
AS $$
DECLARE
    sube_tur yeni_tur;
BEGIN
    SELECT s.isim, COUNT(*) INTO sube_tur
```

```

FROM subeler s
INNER JOIN calisanlar c ON c.sube_id = s.id
WHERE s.isim = sube_ismi
GROUP BY s.id;

RETURN sube_tur;
END;
$$ language 'plpgsql';
--SELECT sube_calisan_sayisi('Davutpaşa Şubesi');
--fonksiyon 3 cursor
CREATE OR REPLACE FUNCTION tur_yardim_miktari (yardim_turu VARCHAR, OUT
yardim_miktari numeric)
RETURNS NUMERIC
AS $$
DECLARE
    yardım_cursor CURSOR FOR SELECT miktar
                                FROM etkinlikler e
                                INNER JOIN yardimlasma_turleri y
                                ON y.id = e.yardimlasma_tur_id
                                WHERE y.isim = yardım_turu;
BEGIN
    yardım_miktari := 0;
    FOR yardım IN yardım_cursor LOOP
        yardım_miktari := yardım_miktari + yardım.miktar;
    END LOOP;
END;
$$ language 'plpgsql';

--SELECT tur_yardim_miktari('erzak')

```

### Madde 11:

```

-- trigger 1 – bir kişinin en fazla dört yardım almasını sağlayan trigger
CREATE OR REPLACE FUNCTION etkinlik_trigger_fonk()
RETURNS TRIGGER
AS $$
DECLARE
    yardım_alma_sayisi INT;
BEGIN
    SELECT COUNT(*) INTO yardım_alma_sayisi
    FROM etkinlikler e
    WHERE e.yardim_alan_id = new.yardim_alan_id;

    IF (yardim_alma_sayisi >= 4) THEN
        RAISE EXCEPTION 'Bir kişi en fazla 4 sefer yardım alabilir.';
    RETURN null;
    ELSE
        return new;
    END IF;
END;

```



```

        END IF;
    END;
    $$ LANGUAGE 'plpgsql';

CREATE OR REPLACE TRIGGER etkinlik_trigger
BEFORE INSERT
ON etkinlikler
FOR EACH ROW EXECUTE PROCEDURE etkinlik_trigger_fonk();

--trigger 2 – Yöneticisi silinen şubenin yeni yöneticisini şubedeki en yaşlı kişi yapan trigger
CREATE OR REPLACE FUNCTION sube_yonetici_trigger_fonk()
RETURNS TRIGGER AS $$
DECLARE
    sube_id_var INT;
BEGIN
    sube_id_var := 0;

    SELECT s.id INTO sube_id_var
    FROM subeler s
    WHERE s.yonetici_id = old.id;

    IF (sube_id_var > 0) THEN
        UPDATE subeler
        SET yonetici_id = tablo.id
        FROM (SELECT c.id FROM subeler s
              INNER JOIN calisanlar c
              ON s.id = c.sube_id
              WHERE s.id = sube_id_var
              AND c.yas = (SELECT MAX(yas) FROM calisanlar c1 WHERE c1.sube_id
= sube_id_var AND c1.id != old.id)
              ) tablo
        WHERE subeler.id = sube_id_var;

        RAISE NOTICE 'Şube yöneticisi güncellendi.';
    RETURN OLD;
    END IF;
    RETURN OLD;
END;
$$ LANGUAGE 'plpgsql';

CREATE OR REPLACE TRIGGER sube_yonetici_trigger
BEFORE DELETE
ON calisanlar
FOR EACH ROW EXECUTE PROCEDURE sube_yonetici_trigger_fonk());

```