

# BIL206 - Veritabanı Sistemleri II

## Proje Ödevi



### TAKIM

191216002 - Furkan Kaya  
191216008 - Ali Murat Tava  
191216018 - Tolgahan Şişman  
191216022 - Kaan İnce

### DANIŞMAN

Pınar Efe

5 Haziran 2021

# Dershane Otomasyon Sistemi

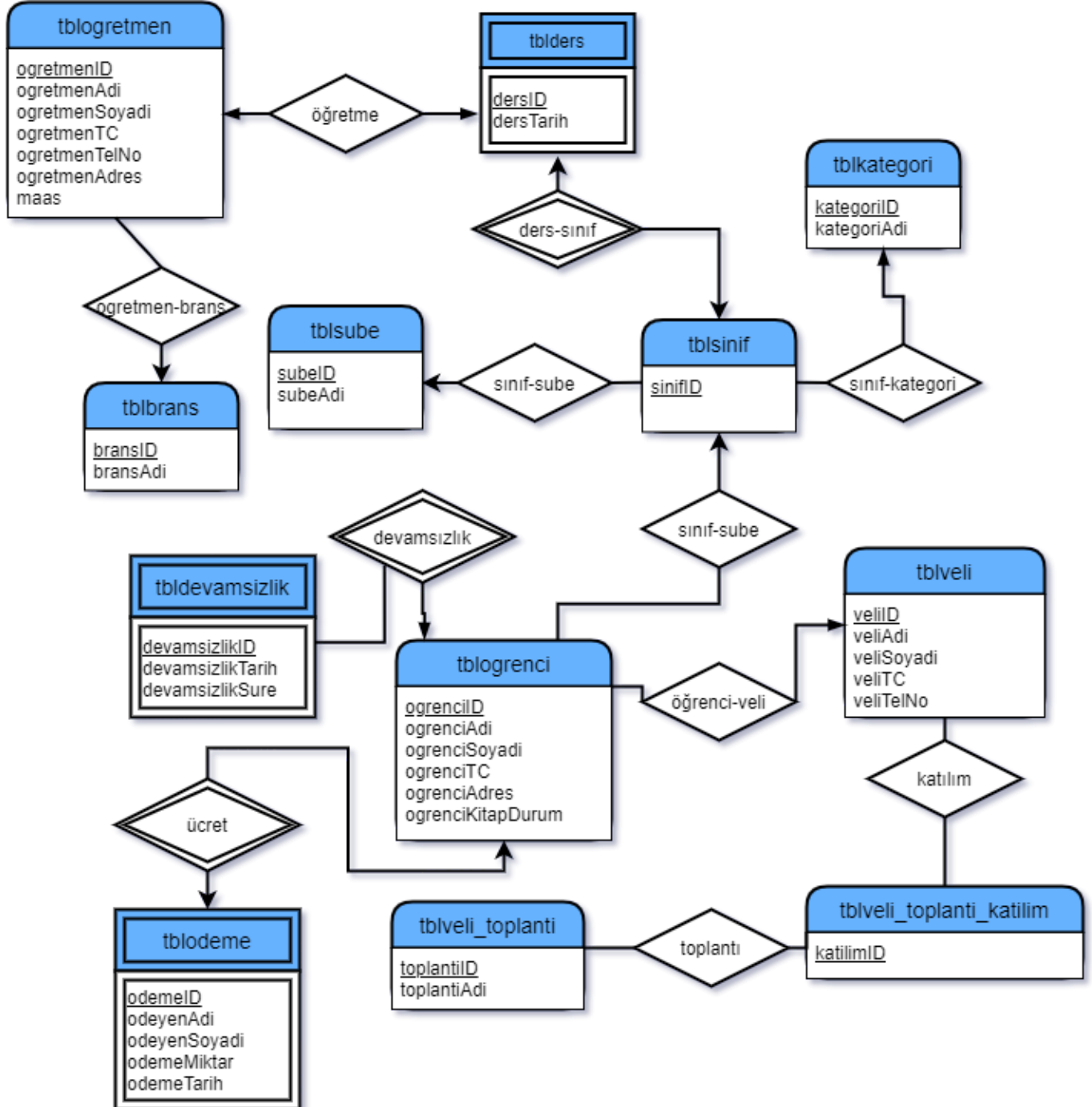
## Proje Konusu ve Açıklaması:

Dershane Otomasyon Sistemi, tüm süreç,durum ve bilgilerin takip edilip tutulmasını sağlayan bir veritabanı sistemidir. 12 tablodan oluşmaktadır. Bunlar;

<b>tblbrans</b>	Öğretmenlerin hangi sınıfta dersi olduğunu tutan tablodur.
<b>tblders</b>	Dershanede ki branşlarının tutulduğu tablodur.
<b>tbldevamsizlik</b>	Öğrencilerin yapmış olduğu devamsızlıkların tutulduğu tablodur.
<b>tblkategori</b>	Dershanede ki sınıfların kategorisinin (9A,9B,11B) tutulduğu tablodur.
<b>tblodeme</b>	Öğrenciler adına ödenmiş olan ücretlerin tutulduğu tablodur.
<b>tblogrenci</b>	Dershaneye kayıtlı öğrencilerin tutulduğu tablodur.
<b>tblogretmen</b>	Dershanede çalışan öğretmenlerin tutulduğu tablodur.
<b>tblsinif</b>	Dershanede ki sınıfların tutulduğu tablodur.
<b>tblsube</b>	Dershanede ki sınıfların şubelerinin(9A,9B,11B) tutulduğu tablodur.
<b>tblveli</b>	Dershaneye kayıtlı öğrencilerin velilerinin tutulduğu tablodur.
<b>tblveli_toplanti</b>	Dershanede şu ana kadar olan toplantılarının isimlerinin tutulduğu tablodur.
<b>tblveli_toplanti_katilim</b>	Velilerin katılım durumlarının tutulduğu tablodur.

## E-R Diyagramı:

Proje kapsamında dersane otomasyonunun veritabanı ihtiyaçları için geliştirilme yapılmıştır. Sistemin ihtiyaçları için tasarlanan veri tabanının E-R Diyagramını aşağıda görebilirsiniz:



# INSERT-DELETE-UPDATE

1. Veli tablosuna kayıt ekleyen SQL sorgusunu yazınız.

```
INSERT INTO tblveli (veliAdi, veliSoyadi, veliTC, veliTelNo) VALUES ('Kaan', 'Kalin', '45456565655', '0545645454');
```

2. ÖğrenciID değeri 9 olan öğrenci 05.06.2021 tarihinde 1 gün devamsızlık yaptı.

Öğrencinin devamsızlık bilgisini tabloya ekleyen SQL sorgusunu yazınız.

```
INSERT INTO tbldevamsizlik(ogrenciID,devamsizlikTarih,devamsizlikSure) VALUES(9,2021-06-05,1)
```

3. Öğretmen tablosundan TC'si 22447896124 olan öğretmeni silen SQL sorgusunu yazınız.

```
DELETE FROM tblogretmen WHERE ogretmenTC="22447896124"
```

4. Maaşı 6000 TL altında olan tüm öğretmenlerin maaşına %10 zam yapan SQL sorgusunu yazınız.

```
UPDATE tblogretmen  
SET Maas=Maas*1.1  
WHERE Maas<=6000
```

5. TC'si 22451785493 olan Deniz Ekici öğretmen yeni bir eve taşınmış. Öğretmenin adres bilgisini güncelleyen SQL sorgusunu yazınız.

```
UPDATE tblogretmen SET ogretmenAdres=("Ninehatun mahallesi 315.sokak No:15 Daire:3")  
WHERE ogretmenTC="22451785493"
```

## BASİT SORGULAR

1.Tanışma Toplantısına katılan velilerin adını soyadını getiren SQL sorgusunu yazınız.

```
SELECT tblveli.veliAdi, tblveli.veliSoyadi FROM tblveli_toplanti_katilim , tblveli_toplanti , tblveli  
WHERE tblveli.veliID = tblveli_toplanti_katilim.veliID AND tblveli_toplanti_katilim.toplantilD =  
tblveli_toplanti.toplantilD AND tblveli_toplanti.toplantiAdi = "Tanışma Toplantısı"
```

**2. Dershanedeki her bir öğrenci için devamsızlık toplamalarını getiren SQL sorgusunu yazınız.**

```
SELECT tbldevamsizlik.ogrenciID, tblogrenci.ogrenciTC, tblogrenci.ogrenciAdi,  
tblogrenci.ogrenciSoyadi, SUM(devamsizlikSure) AS DEVAMSIZLIK  
FROM tbldevamsizlik, tblogrenci  
WHERE tblogrenci.ogrenciID=tbldevamsizlik.ogrenciID  
GROUP BY ogrenciID
```

**3. Devamsızlığı 5 ve üzeri olan öğrencileri getiren SQL sorgusunu yazınız.**

```
SELECT tbldevamsizlik.ogrenciID, tblogrenci.ogrenciTC, tblogrenci.ogrenciAdi,  
tblogrenci.ogrenciSoyadi, SUM(devamsizlikSure) AS DEVAMSIZLIK  
FROM tbldevamsizlik, tblogrenci  
WHERE tblogrenci.ogrenciID=tbldevamsizlik.ogrenciID  
GROUP BY ogrenciID  
HAVING DEVAMSIZLIK >= 5
```

**4. Dershanedeki tüm öğrencilerin sınıflarını ve velilerini getiren SQL sorgusunu yazınız.**

```
SELECT tblkategori.kategoriAdi,tblsube.subeAdi,tblogrenci.ogrenciTC,  
tblogrenci.ogrenciAdi,tblogrenci.ogrenciSoyadi, tblveli.veliTC,tblveli.veliAdi,tblveli.veliSoyadi  
FROM tblogrenci,tblveli, tblkategori,tblsube,tblsinif  
WHERE tblogrenci.sinifID = tblsinif.sinifID  
AND tblsinif.subelD = tblsube.subelD  
AND tblsinif.kategoriD = tblkategori.kategoriD  
AND tblogrenci.veliID = tblveli.veliID
```

**5. Dershaneye ödenen ortalama miktarı getiren SQL sorgusunu yazınız.**

```
SELECT AVG(odemeMiktar) AS Dershaneye_odenene_ortalama_para FROM tblodeme
```

**6. 07.01.2020 tarihinden sonra dershaneye ödeme yapan öğrencileri ve ödedikleri miktarı getiren bir SQL sorgusu yazınız.**

```
SELECT tblogrenci.ogrenciTC ,tblogrenci.ogrenciAdi, tblogrenci.ogrenciSoyadi, tblodeme.odemeTarih  
,tblodeme.odemeMiktar  
FROM tblogrenci, tblodeme  
WHERE tblogrenci.ogrenciID=tblodeme.ogrenciID AND tblodeme.odemeTarih > "2020-01-07"
```

# KARMAŞIK SORGULAR

**1. Ortalamanın üzerinde maaş alan öğretmenleri getiren SQL sorgusu yazınız.**

```
SELECT tblogretmen.ogretmenAdi,tblogretmen.Maas
FROM tblogretmen
WHERE tblogretmen.Maas>(SELECT AVG(tblogretmen.Maas)
FROM tblogretmen)
```

**2. Ödemesi yapılmamış öğrencileri getiren SQL sorgusu yazınız.**

```
SELECT tblogrenci.ogrenciID, tblogrenci.ogrenciTC,tblogrenci.ogrenciAdi,tblogrenci.ogrenciSoyadi
FROM tblogrenci
WHERE tblogrenci.ogrenciID NOT IN
(SELECT tblogrenci.ogrenciID
FROM tblogrenci,tblogrenci
WHERE tblogrenci.ogrenciID = tblogrenci.ogrenciID )
GROUP BY tblogrenci.ogrenciID
```

**3. Matematik branşındaki ortalama maaştan düşük veya eşit maaş olan öğretmenleri getiren SQL sorgusu yazınız.**

```
SELECT * FROM tblogretmen, tblbrans
WHERE tblogretmen.bransID=tblbrans.bransID
AND tblbrans.bransAdi="Matematik"
AND tblogretmen.Maas <= (
SELECT AVG(tblogretmen.Maas) AS ORTMAAS
FROM tblogretmen, tblbrans
WHERE tblbrans.bransID=tblogretmen.bransID
AND tblbrans.bransAdi="Matematik")
```

**4. Matematik branşındaki ortalama maaştan düşük alan matematik branşı öğretmenlerinin maaşına %10 zam yapan SQL sorgusunu yazınız.**

```
UPDATE tblogretmen
SET Maas=Maas*1.1
WHERE ogretmenID IN
(SELECT ogretmenID FROM tblogretmen, tblbrans
WHERE tblogretmen.bransID=tblbrans.bransID)
```

```
AND tblbrans.bransAdi="Matematik"  
    AND Maas<=  
(SELECT AVG(Maas)  
FROM tblogretmen  
WHERE tblogretmen.bransID=tblbrans.bransID  
AND tblbrans.bransAdi="Matematik"))
```

**5. Öğretmenleri ve branşlarını join kullanarak gösteriniz. Maaşa göre küçükten büyüğe doğru sıralayan SQL sorgusunu yazınız.**

```
SELECT tblogretmen.ogretmenID, tblogretmen.ogretmenAdi, tblogretmen.ogretmenSoyadi,  
tblbrans.bransAdi, tblogretmen.Maas  
FROM tblogretmen INNER JOIN tblbrans  
ON tblogretmen.bransID=tblbrans.bransID  
ORDER BY tblogretmen.Maas ASC
```

**6. Öğretmenleri ve branşlarını getiren SQL sorgusunu yazınız.. Branşı 'NULL' olan öğretmenler de dahil.**

```
SELECT tblogretmen.ogretmenAdi, tblogretmen.ogretmenSoyadi, tblbrans.bransAdi  
FROM tblogretmen  
LEFT JOIN tblbrans  
ON tblogretmen.bransID=tblbrans.bransID
```

**7. Dershanede bazı branşlarda öğretmen mevcut değil. Öğretmeni olmayan branşları getiren SQL sorgusunu yazınız..**

```
SELECT tblbrans.bransID, tblbrans.bransAdi FROM tblbrans WHERE tblbrans.bransID NOT IN  
(SELECT tblogretmen.bransID  
FROM tblogretmen, tblbrans  
WHERE tblogretmen.bransID=tblbrans.bransID  
AND tblogretmen.bransID)
```

# TRIGGER

1. Bir öğretmen silindiğinde, öğretmenin vereceği dersleri de ders tablosundan silen bir trigger yazınız.

```
CREATE TRIGGER kayit_silme AFTER DELETE ON tblogretmen
FOR EACH ROW DELETE tblders
FROM tblders
WHERE tblders.ogretmenID=OLD.ogretmenID
```

# PROSEDÜR

1. TC ve MAAS değerlerini parametre olarak alarak, girilen TC numarasına ait öğretmenin maaş değerini güncelleyen bir prosedür yazınız.

```
DELIMITER $$
CREATE DEFINER='root'@'localhost' PROCEDURE `maas_guncelleme` (IN `TC` VARCHAR(11), IN
`MAAS` DOUBLE(15,2))
NO SQL
UPDATE tblogretmen
SET Maas = MAAS
WHERE tblogretmen.ogretmenTC=TC$$
DELIMITER ;
```

2. Öğrenci TC'si, devamsızlık tarihi ve devamsızlık uzunluğunu alıp devamsızlık tablosuna ekleyen bir prosedür yazınız.

```
DELIMITER $$
CREATE DEFINER='root'@'localhost' PROCEDURE `devamsizlik_ekle` (IN `TC` VARCHAR(11), IN
`GUN` DATE, IN `SURE` DOUBLE(2,1))
NO SQL
DETERMINISTIC
BEGIN
DECLARE D INT(11);
SET D = (SELECT ogrenciID FROM tblogrenci WHERE ogrenciTC=TC);
INSERT INTO tbldevamsizlik (ogrenciID, devamsizlikTarih, devamsizlikSure) VALUES (D, GUN, SURE);
END$$
DELIMITER ;
```



# FONKSİYON

1. Şu ana kadar dershaneye ödenen toplam parayı hesaplayan bir fonksiyon yazınız.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` FUNCTION `kasada_bulunan_para`() RETURNS
double(15,2) unsigned zerofill
NO SQL
DETERMINISTIC
BEGIN
DECLARE profit DOUBLE(15,2);
SET profit = (SELECT SUM(odemeMiktar) FROM tblodeme);
RETURN profit;
END$$
DELIMITER ;
```

2. Öğretmenin TC değerini parametre olarak alıp en yakın tarihteki dersini getiren bir fonksiyon yazınız.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` FUNCTION `en_yakin_ders`(`TC` VARCHAR(11))
RETURNS date
NO SQL
BEGIN
DECLARE d DATE;
SET d = (SELECT d.dersTarih
FROM tblogretmen o INNER JOIN tblders d ON o.ogretmenID=d.ogretmenID
WHERE o.ogretmenTC=TC AND d.dersTarih >= NOW()
ORDER BY d.dersTarih ASC LIMIT 1);
RETURN d;
END$$
DELIMITER
```