

## Fonksiyon, Tetikleyici (trigger), Saklı Yordam (Stored procedure) Uygulamaları

Bir eğitim kurumunda öğrenciler farklı kurslara kayıt olabilmektedir. Her kursun belirli bir kontenjanı ve ücreti vardır. Bir öğrenci kursa kayıt edildiğinde sistem otomatik olarak kayıt ücretini hesaplamalı, kurs kontenjanını kontrol etmeli ve kayıt başarılı olursa kalan kontenjanı azaltmalıdır.

### Tablolar

Veritabanında aşağıdaki üç tablo bulunacaktır: öğrenci, kurs, kayıt

Veritabanı ve Tabloların Oluşturulması

```
CREATE DATABASE kurs_sistemi;

USE kurs_sistemi;

CREATE TABLE ogrenci (

    ogrenci_id INT AUTO_INCREMENT PRIMARY KEY,

    ad VARCHAR(50) NOT NULL,

    soyad VARCHAR(50) NOT NULL,

    bolum VARCHAR(100)

);

CREATE TABLE kurs (

    kurs_id INT AUTO_INCREMENT PRIMARY KEY,

    kurs_adi VARCHAR(100) NOT NULL,

    kontenjan INT NOT NULL,

    ucret DECIMAL(10,2) NOT NULL

);

CREATE TABLE kayıt (

    kayıt_id INT AUTO_INCREMENT PRIMARY KEY,

    ogrenci_id INT NOT NULL,

    kurs_id INT NOT NULL,
```

```
kayit_tarihi DATE NOT NULL,  
  
indirim_orani INT DEFAULT 0,  
  
odenen_ucret DECIMAL(10,2),  
  
FOREIGN KEY (ogrenci_id) REFERENCES ogrenci(ogrenci_id),  
  
FOREIGN KEY (kurs_id) REFERENCES kurs(kurs_id)  
);
```

## 2. Örnek Veriler

```
INSERT INTO ogrenci (ad, soyad, bolum) VALUES  
  
('Ayşe', 'Yılmaz', 'Yazılım Mühendisliği'),  
  
('Mehmet', 'Demir', 'Bilgisayar Mühendisliği'),  
  
('Zeynep', 'Kaya', 'Yazılım Mühendisliği');  
  
INSERT INTO kurs (kurs_adi, kontenjan, ucret) VALUES  
  
('SQL Temelleri', 2, 1000.00),  
  
('Python Programlama', 1, 1500.00),  
  
('Web Tasarımı', 3, 1200.00);
```

## Uygulama Görevleri

### Görev 1: Function Yazınız: ucret\_hesapla

Bir kursun ücretini ve indirim oranını alarak öğrencinin ödeyeceği son ücreti hesaplayan bir function yazınız.

**Parametreler:** p\_kurs\_id, p\_indirim\_orani

**Beklenen işlem:** Ödenecek ücret = kurs ücreti - indirim tutarı

### Görev 2: Trigger Yazınız

kayit tablosuna yeni kayıt eklenmeden önce çalışan bir trigger yazınız.

Trigger řu işlemleri yapmalıdır:

1. Kurs kontenjanını kontrol etmelidir.
2. Kontenjan 0 ise kayıt işlemini engellemelidir.
3. Öğrencinin ödeyeceđi ücreti `ucet_hesapla` function'ı ile hesaplamalıdır.
4. Hesaplanan ücreti `odenen_ucret` alanına otomatik yazmalıdır.

### **Görev 3: Trigger Yazınız**

kayıt tablosuna başarılı bir kayıt eklendikten sonra ilgili kursun kontenjanını 1 azaltan trigger yazınız.

### **Görev 4: Procedure Yazınız: öğrenci\_kursa\_kaydet**

Öğrenciyi kursa kaydetmek için bir procedure yazınız.

Parametreler: `p_ogrenci_id`, `p_kurs_id`, `p_indirim_orani`

Bu procedure, kayıt tablosuna yeni kayıt eklemelidir.

**Çağırılması :** `CALL öğrenci_kursa_kaydet(1, 1, 10);`