

BİL312 VERİTABANI SİSTEMLERİ ARA SINAV PROJESİ

SENA GÜRKAN – 20120205021

1. En az 8 tane tablonun bulunduğu, tablo isimleri ve özniteliklerin Türkçe olarak tanımlandığı (Create Table ...) SQL cümleciklerinin hazırlanması gerekmektedir.

```
CREATE TABLE Kullanicilar (  
    id INT PRIMARY KEY,  
    ad VARCHAR(50),  
    soyad VARCHAR(50),  
    email VARCHAR(50),  
    telefon VARCHAR(20),  
    sifre VARCHAR(25),  
    olusturma_tarihi DATE,  
    son_giris_tarihi DATE  
);
```

```
CREATE TABLE Kitaplar (  
    id INT NOT NULL PRIMARY KEY,  
    ad VARCHAR(100),  
    yazar VARCHAR(100),  
    yayinevi VARCHAR(100),  
    sayfa_sayisi INT,  
    stok_sayisi INT,  
    fiyat DECIMAL(10,2),  
    kategori_id INT,  
    FOREIGN KEY (kategori_id) REFERENCES Kategoriler(id)  
);
```

```
CREATE TABLE Kategoriler (  
    id INT NOT NULL PRIMARY KEY,  
    ad VARCHAR(50),  
    aciklama VARCHAR(255)  
);
```

```
CREATE TABLE Yayinevleri (  
    id INT NOT NULL PRIMARY KEY,  
    ad VARCHAR(100),  
    ulke VARCHAR(50),  
);
```

```
CREATE TABLE Siparisler (  
  id INT NOT NULL PRIMARY KEY,  
  kullanıcı_id INT,  
  kitap_id INT,  
  tarih DATE,  
  adet INT,  
  FOREIGN KEY (kullanıcı_id) REFERENCES Kullanıcılar(id),  
  FOREIGN KEY (kitap_id) REFERENCES Kitaplar(id)  
);
```

```
CREATE TABLE Odeme (  
  id INT NOT NULL PRIMARY KEY,  
  kullanıcı_id INT,  
  siparis_id INT,  
  tutar DECIMAL(10,2),  
  tarih DATE,  
  FOREIGN KEY (kullanıcı_id) REFERENCES Kullanıcılar(id),  
  FOREIGN KEY (siparis_id) REFERENCES Siparisler(id)  
);
```

```
CREATE TABLE Yorumlar (  
  id INT NOT NULL PRIMARY KEY,  
  kullanıcı_id INT,  
  kitap_id INT,  
  yorum VARCHAR(255),  
  tarih DATE,  
  FOREIGN KEY (kullanıcı_id) REFERENCES Kullanıcılar(id),  
  FOREIGN KEY (kitap_id) REFERENCES Kitaplar(id)  
);
```

```
CREATE TABLE Favoriler (  
  id INT NOT NULL PRIMARY KEY,  
  kullanıcı_id INT,  
  kitap_id INT,  
  FOREIGN KEY (kullanıcı_id) REFERENCES Kullanıcılar(id),  
  FOREIGN KEY (kitap_id) REFERENCES Kitaplar(id)  
);
```

2. Seçmiş olduğunuz veri tabanı tablolarının indekslerini oluşturunuz (create index on).

```
CREATE INDEX kullanıcılar_email_index ON Kullanıcılar (email);
```

```

CREATE INDEX kullanicilar_soyad_index ON Kullanicilar (soyad);
CREATE INDEX kitaplar_ad_index ON Kitaplar (ad);
CREATE INDEX kitaplar_yazar_index ON Kitaplar (yazar);
CREATE INDEX kitaplar_kategori_index ON Kitaplar (kategori_id);
CREATE INDEX kategoriler_ad_index ON Kategoriler (ad);
CREATE INDEX siparisler_tarih_index ON Siparisler (tarih);
CREATE INDEX siparisler_kullanici_index ON Siparisler (kullanici_id);
CREATE INDEX siparisler_kitap_index ON Siparisler (kitap_id);
CREATE INDEX odeme_tarih_index ON Odeme (tarih);
CREATE INDEX odeme_kullanici_index ON Odeme (kullanici_id);
CREATE INDEX odeme_siparis_index ON Odeme (siparis_id);
CREATE INDEX yorumlar_tarih_index ON Yorumlar (tarih);
CREATE INDEX yorumlar_kullanici_index ON Yorumlar (kullanici_id);
CREATE INDEX yorumlar_kitap_index ON Yorumlar (kitap_id);
CREATE INDEX favoriler_kullanici_index ON Favoriler (kullanici_id);
CREATE INDEX favoriler_kitap_index ON Favoriler (kitap_id);

```

3. Her tabloya, önce tanım tablolarına sonra

detay(hareket) tablolarına olmak üzere veri girişini gerçekleştiriniz (insert into tablo1 values(...))

```

INSERT INTO Kullanicilar (ad, soyad, email, sifre) VALUES (Sena, Gürkan, 'senagurkann@gmail.com', '200105');
INSERT INTO Kullanicilar (ad, soyad, email, sifre) VALUES ('Nazlı', 'Gun', 'nazligunn@gmail.com', '191801');
INSERT INTO Kullanicilar (ad, soyad, email, sifre) VALUES ('Onur', 'Berat', 'oberat@gmail.com', '200501');

```

```

INSERT INTO Kitaplar (ad, yazar, fiyat, kategori_id) VALUES ('Lavinia', 'Ozdemir Asaf', 25.50, 1);
INSERT INTO Kitaplar (ad, yazar, fiyat, kategori_id) VALUES ('Aşk-ı Memnu', 'Halit Ziya Uşaklıgil', 12.99, 2);
INSERT INTO Kitaplar (ad, yazar, fiyat, kategori_id) VALUES ('Bilinmeyen Bir Kadının Mektubu', 'Stefan Zweig', 18.75, 4);
INSERT INTO Kitaplar (ad, yazar, fiyat, kategori_id) VALUES ('Suç ve Ceza', 'Fyodor Dostoyevski', 20.00, 3);

```

```

INSERT INTO Kategoriler (ad) VALUES ('Şiir');
INSERT INTO Kategoriler (ad) VALUES ('Roman');
INSERT INTO Kategoriler (ad) VALUES ('Polisiye');
INSERT INTO Kategoriler (ad) VALUES ('Klasik');

```

```

INSERT INTO Yayınevleri (ad) VALUES ('Kültür Yayınları');
INSERT INTO Yayınevleri (ad) VALUES ('İnkılap Kitabevi');

```

```
INSERT INTO Siparisler (tarih, kullanıcı_id, kitap_id, miktar, toplam_fiyat) VALUES ('2022-04-28', 1, 2, 1, 12.99);
INSERT INTO Siparisler (tarih, kullanıcı_id, kitap_id, miktar, toplam_fiyat) VALUES ('2022-04-29', 2, 3, 2, 37.50);
```

```
INSERT INTO Odeme (tarih, kullanıcı_id, siparis_id, odeme_turu, tutar) VALUES ('2022-04-29', 2, 2, 'Kredi Kartı', 37.50);
```

```
INSERT INTO Yorumlar (kullanıcı_id, kitap_id, yorum) VALUES (1, 1, 'Çok güzel bir roman. Kesinlikle tavsiye ederim.');
```

```
INSERT INTO Yorumlar (kullanıcı_id, kitap_id, yorum) VALUES (2, 2, 'Küçük Prens her yaştan insana hitap eden harika bir kitap.');
```

```
INSERT INTO Favoriler (kullanıcı_id, kitap_id)
VALUES (1, 2),
```

4. Hazırlamış olduğunuz veri tabanına göre, senaryolar oluşturunuz ve bir tablodan satırları filtreleyiniz. Filtreleme yaparken Select ve Where deyimlerini kullanmanız gereklidir.

Kullanıcı adı Sena olan kullanıcılar :

```
SELECT * FROM Kullanicilar WHERE ad = 'Sena';
```

Şiir kitaplarının listesi :

```
SELECT * FROM Kitaplar WHERE kategori_id = 1;
```

Yazarı Özdemir Asaf olan kitaplar :

```
SELECT * FROM Kitaplar WHERE yazar = 'Özdemir Asaf';
```

Stokta kalmayan kitaplar :

```
SELECT * FROM Kitaplar WHERE stok_sayisi = 0;
```

50 tlden fazla fiyatı olan kitaplar :

```
SELECT * FROM Kitaplar WHERE fiyat > 50;
```

Sena adlı kullanıcının siparişleri :

```
SELECT * FROM Siparisler WHERE kullanıcı_id = (SELECT id FROM Kullanicilar WHERE ad = 'Sena');
```

5. Birden çok tabloyu kullanabileceğiniz bir senaryo oluşturunuz. Filtreleme yaparken Select Where ve Join deyimlerini kullanmanız gereklidir.

Sipariş edilen Kitapları Listelemek için:

```
SELECT * FROM Siparisler  
Join Kitaplar on Kitaplar.id = Siparisler.id
```

Kitapların hangi yayinevine ait olduğunu görmek için

```
SELECT * FROM Yayinevleri  
Join Kitaplar on Kitaplar.yayinevi = Yayinevleri.id;
```

Yorum yapan kullanıcılar ve yorumlarını listeler

```
SELECT Kullanicilar.ad,Kullanicilar.soyad,Yorumlar.yorum FROM Kullanicilar  
inner Join Yorumlar  
on Yorumlar.kullanici_id =Kullanicilar.id ;
```

Favorilere Kitap ekleyen kullanıcılar :

```
SELECT Kullanicilar.ad,Kullanicilar.soyad FROM Kullanicilar  
Join Favoriler on Favoriler.kullanici_id = Kullanicilar.id
```

Sipariş edilen Özdemir Asaf kitapları:

```
SELECT Kitaplar.ad, Kitaplar.yazar  
FROM Kitaplar  
JOIN Siparisler ON Siparisler.kitap_id = kitaplar.id  
WHERE Kitaplar.yazar = 'Ozdemir Asaf';
```

Toplamsal fonksiyonları kullanabileceğiniz

(count, max, min, avg, sum gibi) senaryolar oluşturup, Select, Group by, Having, Order By deyimlerini kullanabileceğiniz senaryoların birden çok SQL deyimlerini yazınız (en az dört farklı senaryo için SQL ifadelerini oluşturunuz).

Kullanıcı sayısını bulma :

```
SELECT COUNT(*) AS kullanıcı_sayısı  
FROM Kullanicilar;
```

En pahalı kitabı bulma :

```
SELECT MAX(fiyat) AS en_yüksek_fiyat  
FROM Kitaplar;
```

Toplam kazancı/odemeyi hesaplar :

```
SELECT sum(tutar) as toplam_kazanc  
FROM Odeme;
```

Kitapları fiyatlarına göre sıralama :

```
SELECT Kitaplar.ad, AVG(fiyat) AS ortalama_fiyat  
FROM Kitaplar  
GROUP BY Kitaplar.ad  
ORDER BY ortalama_fiyat DESC;
```

Veri tabanınıza eklemiş olduğunuz satırların bazılarını değiştirmek için Update deyimini kullanan bir SQL cümlecği çalıştırınız.

```
UPDATE Yayınevleri  
SET ad = 'Can Yayınları'  
WHERE id = 3;
```

```
UPDATE Odeme  
SET tutar = 60  
WHERE id = 2;
```

Veri tabanınıza eklemiş olduğunuz satırların bazılarını silmek için Delete deyimini kullanan bir SQL cümlecği çalıştırınız.

```
DELETE FROM kitaplar  
WHERE id = 6;
```

```
DELETE FROM yorumlar  
WHERE id = 2;
```