

ANKARA ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BLM2058 LAB2
2025 BAHAR

CHECKS Tablosu:

CHECK_ID	PAYEE	AMOUNT	REMARKS
1	Migros	270.12	Groceries
2	Petrol Ofisi	105.00	Gas (For Car)
3	TCDD	25.00	Train to Konya
4	Turkcell	528.00	Cellular Phone
5	Baskent Elektrik	113.78	Electricity Bill
6	Flo	175.00	Shoes
7	Migros	315.90	Groceries
8	Allianz Sigorta	365.43	Car Insurance

SELECT STATEMENT

SELECT * FROM CHECKS; sorgusu tablodaki bütün sütunları getirir.

Önceden belirlenmiş sırayı takip etmek zorunda değiliz. Aşağıdaki sorgu tablodan farklı bir sıra ile sütunları getirir.

SELECT PAYEE, REMARKS, AMOUNT, CHECK_ID FROM CHECKS;

Farklı sayıda sütun da seçebiliriz.

SELECT PAYEE, REMARKS FROM CHECKS;

Yukarıdaki sorgunun sonucunda tekrarlanan veri görüyoruz.

SELECT DISTINCT PAYEE, REMARKS FROM CHECKS; sorgusunu kullandığımızda tekrarlanan verinin sadece bir kopyasını görürüz.

CONDITIONS

Belirli özellikteki kayıtları almak istiyorsak bu özellikleri WHERE içinde tanımlayabiliriz.

```
SELECT AMOUNT, CHECK_ID FROM CHECKS WHERE PAYEE = 'TCDD';
```

Farklı sayıda özellik gerekiyorsa WHERE içine ek koşullar eklenebilir.

```
SELECT REMARKS FROM CHECKS WHERE PAYEE = 'TCDD' and AMOUNT>10;
```

OPERATORS

Toplama (+), çıkarma (-), bölme(/) , çarpma (*) ve mod (%) işlemlerinden bazılarının kullanımı aşağıdaki gibidir.

- ```
SELECT AMOUNT,AMOUNT+0.15 FROM CHECKS;
```
- ```
SELECT PAYEE, -AMOUNT AS NEWAMOUNT FROM CHECKS;
```
- ```
SELECT PAYEE, (AMOUNT/2) AS NEWAMOUNT FROM CHECKS;
```
- ```
SELECT CHECK_ID, (AMOUNT*0.9) AS NEWAMOUNT FROM CHECKS;
```

COMPARISON OPERATORS

Karşılaştırmalar için büyüktür (>), küçüktür (<), büyük veya eşittir (>=), küçük veya eşittir (<=), eşittir (=) ve eşit değildir (!=) operatörlerinden bazılarının kullanımı aşağıdaki gibidir.

- ```
SELECT *
FROM CHECKS
WHERE PAYEE='Migros';
```
- ```
SELECT *  
FROM CHECKS  
WHERE PAYEE !='Petrol Ofisi';
```

Eğer verinin belli kısımlarının belirli bir örüntüye uygun olmasını bekliyorsak LIKE kullanabiliriz.

- ```
SELECT *
FROM CHECKS
WHERE REMARKS LIKE '%EL%';
```

%EL EL ile biten REMARKS ları, EL%, EL ile başlayan REMARKS ları aramak için kullanılır.

### **LOGICAL OPERATORS**

- ```
SELECT PAYEE  
FROM CHECKS  
WHERE AMOUNT <=150  
AND  
CHECK_ID >3;
```

IN and BETWEEN

```
SELECT * FROM FRIENDS WHERE STATE='CA' OR STATE='CO' OR STATE='LA';
```

Çoklu OR yerine IN kullanılabilir.

```
SELECT * FROM FRIENDS WHERE STATE IN ('CA','CO','LA');
```

Belirli bir aralık için BETWEEN kullanılabilir.

```
SELECT * FROM CHECKS WHERE AMOUNT >25 AND AMOUNT<200;
```

Aşağıdaki sorgu AND yerine BETWEEN kullanmıştır.

```
SELECT * FROM CHECKS WHERE AMOUNT BETWEEN 25 AND 200;
```

ORDER BY

Belirli bir field kullanılarak sıralama yapmak için ORDER BY kullanılır.

```
SELECT * FROM CHECKS ORDER BY CHECK_ID DEC;
```

Yukarıdaki sorgunun sonucunda kayıtlar CHECK_ID leri azalan şekilde sıralanacaktır.

LAB EXERCISES

CHECKS tablosunu kullanarak aşağıdaki soruları cevaplayınız.

CHECK_ID	PAYEE	AMOUNT	REMARKS
1	Migros	270.12	Groceries
2	Petrol Ofisi	105.00	Gas (For Car)
3	TCDD	25.00	Train to Konya
4	Turkcell	528.00	Cellular Phone
5	Baskent Elektrik	113.78	Electricity Bill
6	Flo	175.00	Shoes
7	Migros	315.90	Groceries
8	Allianz Sigorta	365.43	Car Insurance

1. PAYEE leri M veya P ile başlayan bütün çeklerin CHECK_ID, PAYEE ve AMOUNT bilgilerini getiren sorguyu yazınız.

2. Car harcamaları ile ilgili tüm çeklerin CHECK_ID, PAYEE ve AMOUNT bilgilerini getiren sorguyu yazınız.

3. Çek AMOUNT değeri 200.00 TL nin üzerinde olan bütün PAYEE leri getiren sorguyu yazınız. Tekrarlanan cevabın sadece bir kopyası gösterilecektir.

4. Tüm çeklerin tüm bilgilerini çek AMOUNT değerine göre azalan sıralı olacak şekilde getiren sorguyu yazınız.

5. Shoes veya Groceries harcamaları ile ilgili tüm çeklerin CHECK_ID, PAYEE ve AMOUNT bilgilerini getiren sorguyu yazınız.