CREATE TABLE ogrenciler3

(

id int,

isim VARCHAR(50),

veli\_isim VARCHAR(50),

yazili\_notu int

);

INSERT INTO ogrenciler3 VALUES(123, 'Ali Can', 'Hasan',75);

INSERT INTO ogrenciler3 VALUES(124, 'Merve Gul', 'Ayse',85);

INSERT INTO ogrenciler3 VALUES(125, 'Kemal Yasa', 'Hasan',85);

INSERT INTO ogrenciler3 VALUES(126, 'Nesibe Yilmaz', 'Ayse',95);

INSERT INTO ogrenciler3 VALUES(127, 'Mustafa Bak', 'Can',99);

INSERT INTO ogrenciler3 VALUES(127, 'Mustafa Bak', 'Ali', 99);

select \* from ogrenciler3

-- İsmi Mustafa Bak ve Nesibe Yılmaz olan kayıtları silelim

DELETE FROM ogrenciler3 WHERE isim='Mustafa Bak' or isim = 'Nesibe Yilmaz';

-- Veli ismi Hasan datayi silelim

DELETE FROM ogrenciler3 WHERE veli\_isim='Hasan';

-- TRUNCATE --

-- Bir tablodaki tüm verileri geri alamayacağımız şekilde siler. Şartlı silme yapmaz

TRUNCATE ogrenciler3

-- ON DELETE CASCADE

DROP TABLE if exists adresler -- Eğer tablo varsa tabloyu siler

INSERT INTO talebe VALUES(123, 'Ali Can', 'Hasan',75);

INSERT INTO talebe VALUES(124, 'Merve Gul', 'Ayse',85);

INSERT INTO talebe VALUES(125, 'Kemal Yasa', 'Hasan',85);

INSERT INTO talebe VALUES(126, 'Nesibe Yılmaz', 'Ayse',95);

INSERT INTO talebe VALUES(127, 'Mustafa Bak', 'Can',99);

INSERT INTO not1 VALUES ('123','kimya',75);

INSERT INTO not1 VALUES ('124', 'fizik',65);

INSERT INTO not1 VALUES ('125', 'tarih',90);

INSERT INTO not1 VALUES ('126', 'Matematik',90);

select \* from talebeler2;

select \* from notlar2;

-- Notlar tablosundan talebe\_id'si 123 olan datayı silelim

DELETE FROM notlar WHERE talebe\_id='123';

-- Talebeler tablosundan id'si 126 olan datayı silelim

DELETE FROM talebeler2 WHERE id='126';

CREATE TABLE talebe

(

id CHAR(3) primary key,

isim VARCHAR(50),

veli\_isim VARCHAR(50),

yazili\_notu int

);

CREATE TABLE not1(

talebe\_id char(3),

ders\_adi varchar(30),

yazili\_notu int,

CONSTRAINT notlar\_fk FOREIGN KEY (talebe\_id) REFERENCES talebe(id)

on delete cascade

);

INSERT INTO talebe VALUES(123, 'Ali Can', 'Hasan',75);

INSERT INTO talebe VALUES(124, 'Merve Gul', 'Ayse',85);

INSERT INTO talebe VALUES(125, 'Kemal Yasa', 'Hasan',85);

INSERT INTO talebe VALUES(126, 'Nesibe Yılmaz', 'Ayse',95);

INSERT INTO talebe VALUES(127, 'Mustafa Bak', 'Can',99);

INSERT INTO not1 VALUES ('123','kimya',75);

INSERT INTO not1 VALUES ('124', 'fizik',65);

INSERT INTO not1 VALUES ('125', 'tarih',90);

INSERT INTO not1 VALUES ('126', 'Matematik',90);

select \* from talebe

select \* from not1

-- Talebeler tablosundan id'si 126 olan datayı silelim

DELETE FROM talebe WHERE id='126'

/\*

Her defasında önce child tablodaki verileri silmek yerine ON DELETE CASCADE silme özelliği ile

parent tablo dan da veri silebiliriz. Yanlız ON DELETE CASCADE komutu kullanımında parent tablodan sildiğimiz

data child tablo dan da silinir

\*/

-- IN CONDITION

DROP TABLE if exists musteriler

CREATE TABLE musteriler (

urun\_id int,

musteri\_isim varchar(50),

urun\_isim varchar(50)

);

INSERT INTO musteriler VALUES (10, 'Mark', 'Orange');

INSERT INTO musteriler VALUES (10, 'Mark', 'Orange');

INSERT INTO musteriler VALUES (20, 'John', 'Apple');

INSERT INTO musteriler VALUES (30, 'Amy', 'Palm');

INSERT INTO musteriler VALUES (20, 'Mark', 'Apple');

INSERT INTO musteriler VALUES (10, 'Adem', 'Orange');

INSERT INTO musteriler VALUES (40, 'John', 'Apricot');

INSERT INTO musteriler VALUES (20, 'Eddie', 'Apple');

SELECT \* FROM musteriler

-- Musteriler tablosundan urun ismi orange, apple veya apricot olan tüm verileri listeleyiniz

SELECT \* FROM musteriler WHERE urun\_isim='Orange' OR urun\_isim='Apple' OR urun\_isim='Apricot';

-- IN CONDITION

SELECT \* FROM musteriler WHERE urun\_isim IN ('Orange','Apple','Apricot');

-- NOT IN --> Yazdığımız verilerin dışındakileri getirir

SELECT \* FROM musteriler WHERE urun\_isim NOT IN ('Orange','Apple','Apricot');

SELECT \* FROM musteriler WHERE urun\_isim='Orange' and urun\_id=10

--BETWEEN CONDITION

-- Musteriler tablosundan urun\_id'si 20 ile 40 arasında olan tüm verileri listeleyiniz

SELECT \* FROM musteriler WHERE urun\_id>=20 and urun\_id<=40;

SELECT \* FROM musteriler WHERE urun\_id BETWEEN 20 and 40;

SELECT \* FROM musteriler WHERE urun\_id NOT BETWEEN 20 and 40;

-- SUBQUERIES --> Sorgu içinde sorgu

CREATE TABLE calisanlar2

(

id int,

isim VARCHAR(50),

sehir VARCHAR(50),

maas int,

isyeri VARCHAR(20)

);

INSERT INTO calisanlar2 VALUES(123456789, 'Ali Seker', 'Istanbul', 2500, 'Vakko');

INSERT INTO calisanlar2 VALUES(234567890, 'Ayse Gul', 'Istanbul', 1500, 'LCWaikiki');

INSERT INTO calisanlar2 VALUES(345678901, 'Veli Yilmaz', 'Ankara', 3000, 'Vakko');

INSERT INTO calisanlar2 VALUES(456789012, 'Veli Yilmaz', 'Izmir', 1000, 'Pierre Cardin');

INSERT INTO calisanlar2 VALUES(567890123, 'Veli Yilmaz', 'Ankara', 7000, 'Adidas');

INSERT INTO calisanlar2 VALUES(456789012, 'Ayse Gul', 'Ankara', 1500, 'Pierre Cardin');

INSERT INTO calisanlar2 VALUES(123456710, 'Fatma Yasa', 'Bursa', 2500, 'Vakko');

CREATE TABLE markalar

(

marka\_id int,

marka\_isim VARCHAR(20),

calisan\_sayisi int

);

INSERT INTO markalar VALUES(100, 'Vakko', 12000);

INSERT INTO markalar VALUES(101, 'Pierre Cardin', 18000);

INSERT INTO markalar VALUES(102, 'Adidas', 10000);

INSERT INTO markalar VALUES(103, 'LCWaikiki', 21000);

INSERT INTO markalar VALUES(104, 'Nike', 19000);

select \* from calisanlar2

select \* from markalar

-- Çalisan sayisi 15.000’den cok olan markalarin isimlerini

--ve bu markada calisanlarin isimlerini ve maaşlarini listeleyin.

SELECT isim,maas,isyeri FROM calisanlar2

WHERE isyeri IN (SELECT marka\_isim FROM markalar WHERE calisan\_sayisi>15000);

-- marka\_id’si 101’den büyük olan marka çalişanlarinin

-- isim, maaş ve şehirlerini listeleyiniz

SELECT isim,maas,sehir FROM calisanlar2

WHERE isyeri IN (SELECT marka\_isim FROM markalar WHERE marka\_id>101);

- Çalisan sayisi 15.000’den cok olan markalarin isimlerini ve bu markada calisanlarin

-- isimlerini ve maaşlarini listeleyin.

SELECT isim,maas,isyeri FROM calisanlar2

WHERE isyeri IN (SELECT marka\_isim FROM markalar WHERE calisan\_sayisi>15000);

-- marka\_id’si 101’den büyük olan marka çalişanlarinin

-- isim, maaş ve şehirlerini listeleyiniz.

SELECT isim,maas,sehir FROM calisanlar2

WHERE isyeri IN (SELECT marka\_isim FROM markalar WHERE marka\_id>101);

/\*

--ÖDEV- Ankara’da calisani olan markalarin marka id'lerini ve calisan sayilarini listeleyiniz.

\*/

-- AGGREGATE METHOD

--Calisanlar tablosunda maksimum maası listeleyiniz

SELECT max(maas) AS maksimum\_maas FROM calisanlar2;

/\*

Eğer bir sütuna geçici olarak bir isim vermek istersek AS komutunu yazdıktan sonra vermek

istediğimiz ismi yazarız

\*/

--Calisanlar tablosunda minimum maası listeleyiniz

SELECT min(maas) AS minimum\_maas FROM calisanlar2;

--Calisanlar tablosundaki maasların toplamını listeleyiniz

SELECT sum(maas) AS maas\_toplamı FROM calisanlar2;

--Calisanlar tablosundaki maasların ortalamasını listeleyiniz

SELECT avg(maas) AS maas\_ortalaması FROM calisanlar2;

SELECT round(avg(maas),2) AS maas\_ortalaması FROM calisanlar2;

--Calisanlar tablosundaki maasların sayısı

SELECT count(maas) AS maas\_sayısı FROM calisanlar2;

/\*

\*/