



## VERİTABANI YÖNT. SİST. PRATİK NOTLAR 3 !! 😊

Celil ÖZTÜRK

16Kasım2020(Hafta 3-4-5)

### 3. Hafta Sorgu ve Ders İçi Örnekleri

❖ **Tablo oluşturma:** Yeni bir veritabanı tablosu oluşturma ,

```
create table ornektablo (  
ID varchar(5),  
ogrenci_no varchar(10),  
ogrenci_bilgileri varchar(50))
```

❖ **Kayıt ekleme:** Bir tabloya yeni kayıt ekleme,

```
insert into ornektablo values("22","2002","siber güvenlik bölümü")  
insert into instructor values ('33333', "test", "test dept", null)
```

❖ **Kayıt silme:** Bir tablo üzerinde bulunan kayıtların silinmesi,

```
SET SQL_SAFE_UPDATES=0;  
delete from ornektablo //( where ile kullanımı)
```

❖ **Sütun Ekleme :** Bir tabloya yeni bir sütun ekleme,

```
alter table ornektablo add eklenensutun varchar(11);
```

❖ **Sütun silme :** Bir tabloda bulunan sütunu silme,

```
alter table ornektablo drop eklenensutun
```

❖ **LIKE :** Bir kayıta bulunan karakter dizesi üzerinde karşılaştırma, dize eşleme,

```
select name from instructor where name like '%ar%'
```

```
select name from instructor where name like '____' //4 karakterli kayıtlar
```

- ❖ Aralık Between işlemi: Bir kayıta bulunan karakter dizesi üzerinde karşılaştırma, aralık belirtme,

```
select name,salary from instructor where salary between 90000 and 100000
```

- ❖ Order By (Sıralama): Alfabetik veya numerik sıralama

```
select name from instructor order by name // azalan için desc
```

- ❖ Group By (Gruplama):...

```
select dept_name from department group by dept_name;
```

- ❖ Boş değer olan kayıtları listeleme(is null)

```
select name from instructor where salary is null
```

- ❖ Tüm eğitimcilerin bölüm adlarını listele(kopyaları/tekrar eden satırları kaldırın)

```
SELECT distinct dept_name FROM instructor
```

- ❖ AVG (Ortalama Değer)

```
select avg (salary) from instructor where dept_name= 'Comp. Sci.';
```

- ❖ MIN-MAX Değer

```
select min(salary) from instructor
```

```
select max(salary) from instructor
```

- ❖ Count (Kayıt Sayısı)

```
select count(*) from course
```

- ❖ As (Takma Ad/Alias)

```
select count(*)as toplamkayit from course
```

UPDATE deyimi, bir tablodaki mevcut kayıtları güncellemek için kullanılır:

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value
```

```
UPDATE ornektablo SET ogrenci_bilgileri="Elektrik-Elektronik Mühendisliği" WHERE
ogrenci_no="190102"
```

## 4. Hafta Sorgu ve Ders İçi Örnekleri

**JOIN :SQL'de birbiri ile ilişkili tablolardan veri sorgusu için kullanılır.**

select \* from student **join** takes **on** student.ID = takes.ID →\*\*\*on koşulu genel belirtme izin verir.

- Natural Join

```
select name, course_id from student natural join takes
```

- Farklı Kullanım Benzer sonuç:

```
select name, course_id from student, takes where student.ID = takes.ID;
```

**\*\*NOT:** Using kullanılarak ilişkilendirilen tablolarda eşitlenmesi gereken sütunların belirtilmesi için "using" kullanılır.

```
select name, title from (student natural join takes) join course using (course_id)
```

**Çoklu ilişki kullanımı:** Example -- Öğretim elemanlarının adlarını, öğrencilerin aldıkları derslerin başlıklarıyla birlikte listeleyin.

```
select name, title from student natural join takes, course where takes.course_id =
course.course_id;
```

**Outer join kullanımı:** Tablodaki eşleşen kayıtlar ile birlikte, eşleşmeyen kayıtları da ekler.

- select \* from course natural right outer join prereq
- o SELECT \* FROM course natural join prereq
- o SELECT \* FROM course natural left outer join prereq //soldaki tablonun tamamını alır.

**View:** bir sorgunun kaydedilmesine olanak tanır. Sorgu sonucunda birden fazla tablodan oluşacak olan sonuçların tek bir tablo olarak saklanmasıdır. Diğer bir ifade ile, oluşturulan sorgu sonucunu saklamak için de view kullanılır.

View oluşturma:

- create view testview as select \* from course natural join student
- create view maas\_ortalama as select avg(salary) from instructor where dept\_name='Comp. Sci.';
- create view bolum\_maas\_toplamlari as select dept\_name, sum(salary) from instructor group by dept\_name

View okuma/çağırma:

Select \* from view\_adi

Select \* from testview

View Silme

Drop View test

**Diğer Viewler ile çalışma(View Genişletme):**

create view comp\_science\_maas\_toplam as select \* from bolum\_maas\_toplamlari where dept\_name="Comp. Sci."

## Sorgu birleştirme(Union)

2017 Sonbahar veya 2018 İlkbaharında verilen dersleri bulan sql sorgusu/sogularını yazınız.

select course\_id from section where semester = 'Fall' and year = 2017

union

select course\_id from section where semester = 'Spring' and year = 2018