

a. Belirli Kolonları Seçme

- Çalışanların sadece FirstName, LastName ve Salary bilgilerini getiren bir SQL sorgusu yazınız.

```
SELECT FirstName, LastName, Salary  
FROM employees;
```

→ İkinci satırda kolonlar hangi tablodan getirilecek onu belirttik.

→ İlk satırda hangi kolonların getirileceğini belirtiyoruz.

| | firstname character varying (50) | lastname character varying (50) | salary numeric (10,2) |
|---|-------------------------------------|------------------------------------|--------------------------|
| 1 | John | Doe | 55000.00 |
| 2 | Jane | Smith | 65000.00 |
| 3 | Sam | Brown | 52000.00 |
| 4 | Lisa | White | 70000.00 |
| 5 | Mark | Black | 75000.00 |
| 6 | Lucy | Green | 60000.00 |

Çıktı olarak bu tabloyu elde ediyoruz.

b. DISTINCT Komutu ile Tekrarları Önleme

- Çalışanların çalıştıkları departmanları benzersiz olarak listeleyen bir SQL sorgusu yazınız.

```
SELECT DISTINCT departmentid  
FROM employees;
```

→ “DISTINCT” komutu farklı olan ifadeleri her birinden kaç tane var onu bize gösteren komuttur.

→ -Bu yazmış olduğumuz kod ile beraber “employees” tablosunun “departmentid” kolonundan bize hangi departmanda kaç tane personelin çalıştığı bilgisini getirecek.

| | departmentid integer |
|---|-------------------------|
| 1 | 3 |
| 2 | 2 |
| 3 | 1 |

→ Çıktımız da bu şekilde. Görmüş olduğumuz üzere 1 nolu departmandan yani “IT” departmanında çalışan 3 kişi, 2 nolu departmanda yani “HR” departmanında çalışan 2 kişi ve son olarak 3 nolu departmanda yani “Finans” departmanında çalışan 1 kişi bulunmaktadır.

c. Belirli Bir Departmana Ait Çalışanları Listeleme

- Sadece IT departmanında çalışanların bilgilerini getiren bir SQL sorgusu yazınız.

```
SELECT *  
FROM employees  
WHERE departmentid = 1;
```

- -İlk olarak "*" ifadesi ile çalışanların bütün kolonların listeleneceğini gösteriyoruz
- Daha sonrasında employees tablosundan bu kolonların alınacağını belirtiyoruz.
- -WHERE condition (şart) durumlarında kullanılan bir ifadedir. Burada biz de sadece IT bölümünde çalışan kişileri istediğimiz için "departmentid"si 1 olan çalışanların getirilmesi gerektiğini belirterek sorgumuzu bir şarta bağlıyoruz.

| | employeeid [PK] integer | firstname character varying (50) | lastname character varying (50) | age integer | departmentid integer | salary numeric (10,2) | joindate date |
|---|----------------------------|-------------------------------------|------------------------------------|----------------|-------------------------|--------------------------|------------------|
| 1 | 1 | John | Doe | 30 | 1 | 55000.00 | 2020-01-15 |
| 2 | 3 | Sam | Brown | 28 | 1 | 52000.00 | 2021-04-25 |
| 3 | 5 | Mark | Black | 50 | 1 | 75000.00 | 2015-11-05 |

- Çıktımız da yukarıdaki gibi oluyor. Görüldüğü üzere sadece 1 nolu departmanda çalışan kişileri listelemiş bulunuyoruz.

d. Maaşa Göre Sıralama

- Çalışanları maaşlarına göre büyükten küçüğe sıralayan bir SQL sorgusu yazınız.

```
SELECT *  
FROM employees  
ORDER BY salary DESC;
```

- Yine bir önceki soruda yaptığımız gibi kolon ve tablo işlemlerini hallediyoruz.
- ORDER BY komutu bizim verilerimizi sıralamamızı sağlar. Burda tek yapmamız gereken hangi kritere göre sıralama yapacağımızı belirtmek.
- "salary" kolonuna göre sıralama yapışacağını yukarda belirttik "DESC" ifadesi ile de büyükten küçüğe doğru sıralamanın düzenlenmesini sağlamış olduk.

| | employeeid [PK] integer | firstname character varying (50) | lastname character varying (50) | age integer | departmentid integer | salary numeric (10,2) | joindate date |
|---|----------------------------|-------------------------------------|------------------------------------|----------------|-------------------------|--------------------------|------------------|
| 1 | 5 | Mark | Black | 50 | 1 | 75000.00 | 2015-11-05 |
| 2 | 4 | Lisa | White | 35 | 3 | 70000.00 | 2019-03-18 |
| 3 | 2 | Jane | Smith | 45 | 2 | 65000.00 | 2018-07-20 |
| 4 | 6 | Lucy | Green | 40 | 2 | 60000.00 | 2017-10-10 |
| 5 | 1 | John | Doe | 30 | 1 | 55000.00 | 2020-01-15 |
| 6 | 3 | Sam | Brown | 28 | 1 | 52000.00 | 2021-04-25 |

→ Çıktımız da bu şekilde.

e. Kolonları Birleştirme (Concatenation)

- Çalışanların FirstName ve LastName alanlarını birleştirerek, tam adlarını içeren yeni bir kolon oluşturan bir SQL sorgusu yazınız.

```
SELECT FirstName || ' ' || LastName AS FullName
FROM employees;
```

- Concat işlemi “||” ifadesi ile gerçekleştirilir. İlk satırda firstname ve lastname kolonlarının aralarında bir adet boşluk bırakılarak birleştirileceği bir komut yazdık.
- Daha sonra bu yeni oluşacak kolona yeni bir isim verdik. Bu işlemi yaparken de “Alias”lardan yararlandık. Alias takma ad anlamına gelir ve “AS” kelimesi ile oluşturulur.

| | fullname text |
|---|------------------|
| 1 | John Doe |
| 2 | Jane Smith |
| 3 | Sam Bro... |
| 4 | Lisa White |
| 5 | Mark Black |
| 6 | Lucy Green |

- Çıktımız da bu şekilde.