

TURK STUDENT COMMUNITY SQL BOOTCAMP

BİRİNCİ HAFTA ÖDEVİ

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

Query

Query History

```
1  -- a. Belirli Kolonları Seçme
2  -- employees tablosundan çalışanların firstname, lastname ve salary
3  -- bilgilerini getirmek için select komutundan sonra ilgili kolonların adını
4  -- girip, from ile hangi tablodan çektiğimizi belirttik.
5  select firstname, lastname, salary from employees;
```

Data Output

Messages

Notifications

+

SQL

	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

b) Çalışanların çalıştıkları departmanları benzersiz olarak listeleyen bir SQL sorgusu yazınız.

Query

Query History

```
8  -- b) DISTINCT Komutu ile Tekrarları Önleme
9  -- çalışılan departmanların tekrarsız bir şekilde filtrelenmesi
10 -- için distinct komutunu kullandık.
11 select distinct departmentname from departments;
12
```

Data Output

Messages

Notifications

+

SQL

	departmentname character varying (50)
1	Finance
2	IT
3	HR

c) Sadece IT departmanında çalışanların bilgilerini getiren bir SQL sorgusu yazınız.

Query

Query History

```
14 -- c) Belirli Bir Departmana Ait Çalışanları Listeleme
15 -- departments tablosunda IT departmanının id'si 1,
16 -- employees tablosunda departmenid bilgisi 1 olan çalışanları getireceğiz
17 -- where komutu verilen şarta göre filtreleme sağlar.
18 -- where komutundan sonra aradığımız özelliğin değerini belirterek istenilen sorguya ulaştık.
19 select * from employees where departmentid =1;
20
```

Data Output

Messages

Notifications

≡

📄

▼

📋

▼

🗑️

🔄

⬇️

📈

SQL

	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

d) Çalışanları maaşlarına göre büyükten küçüğe sıralayan bir SQL sorgusu yazınız.

Query

Query History

```

22 -- d) Maaşa Göre Sıralama
23 -- order by komutu verileri default olarak küçükten büyüğe doğru sıralar,
24 -- order by kullanarak sıralamak istediğimiz ilgili kolonun adından sonra
25 -- desc komutunu eklersek varsayılanın tersi yönünde yani büyükten küçüğe doğru
26 -- sıralama yapmış oluruz.
27 ✓ select * from employees
28 order by salary desc;
29

```

Data Output

Messages

Notifications

≡

📄

▼

📋

▼

🗑️

🔄

📶

⬇️

📈

SQL

	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

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

Query Query History

```
31 -- e) Kolonları Birleştirme (Concatenation)
32 -- çalışanların firstname ve lastname alanlarını concat (||) operatörü ile
33 -- birleştirdik ve 'as' komutu kullanarak fullname etiketi ile tek kolonda
34 -- gösterilmesini sağladık.
35 v select employeeid, firstname || ' ' || lastname as fullname,
36 age, departmentid, salary, joindate from employees;
37
```

Data Output Messages Notifications

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