

# TURK STUDENT COMMUNITY SQL BOOTCAMP

## ÜÇÜNCÜ HAFTA ÖDEVİ

1- *Invoice* tablosunda “USA” ülkesine ait, 2009 yılı içerisinde oluşturulmuş tüm faturaların toplamını listeleyen bir sorgu yazınız.

Query Query History

```
1 Select * from invoice
2
3 SELECT SUM(total)
4 FROM invoice
5 WHERE billing_country = 'USA'
6 AND EXTRACT(YEAR FROM invoice_date) = 2009;
7
8 /*
9 WHERE komutu ile billing_country kolonunda USA değerlerini filtreledik.
10 EXTRACT komutu ile 2009 olan değerleri ayıkladık,
11 (AND komutunda iki şartın da sağlanması beklenir)
12 SUM komutu ile total kolonunun değerini topladık.
13 */
```

Data Output Messages Notifications

SQL

sum numeric

1 103.95

2- *Track* tablosundaki tüm parça bilgilerini, bu parçaların ait olduğu *playlisttrack* ve *playlist* tablolarıyla birleştirerek (JOIN) listeleyen bir sorgu yazınız.

Query Query History

```
15 Select * from track
16 Select * from playlisttrack
17 Select * from playlist
18
19 SELECT
20 t.*, --bu ifade ile track tablosundaki tüm kolonları seçtik.
21 pt.playlist_id, --bu ifade ile playlisttrack tablosundaki playlist_id kolonunu seçtik
22 p.name AS playlist_type --bu ifade, playlist tablosundaki name kolonunu seçtik
23 --ve playlist_type olarak adlandırdık.
24 FROM track t
25 JOIN playlisttrack pt ON t.track_id = pt.track_id
26 JOIN playlist p ON pt.playlist_id = p.playlist_id;
27
28 --JOIN ile tablolar verilen kolonlar üzerinden birleştirdik.
```

Data Output Messages Notifications

SQL

track_id integer	name character varying (200)	album_id integer	mediatype_id integer	genre_id integer	composer character varying	milliseconds integer	bytes integer	unitprice numeric (10,2)	playlist_id integer	playlist_type character varying	
1	3402	Band Members Discuss Tracks from "Revelations"	271	3	23	[null]	294294	611188	0.99	1	Music
2	3389	Revelations	271	2	23	[null]	252376	411105	0.99	1	Music
3	3390	One and the Same	271	2	23	[null]	217732	355904	0.99	1	Music
4	3391	Sound of a Gun	271	2	23	[null]	260154	423499	0.99	1	Music
5	3392	Until We Fall	271	2	23	[null]	230758	376660	0.99	1	Music
6	3393	Original Fire	271	2	23	[null]	218916	357782	0.99	1	Music
7	3394	Revelation	271	2	23	[null]	230758	376660	0.99	1	Music

3- *Track* tablosundan "Let There Be Rock" adlı albüme ait tüm parçaları listeleyen, sanatçı (*Artist*) bilgisini de içeren bir sorgu yazınız. Ayrıca, sonuçları parça süresi (*milliseconds*) büyükten küçüğe sıralayınız.

Query Query History

```
31 Select * from track
32 Select * from album
33 Select * from artist
34
35 SELECT
36     track.name AS "Parça Adı",
37     artist.name AS "Sanatçı Adı",
38     track.milliseconds AS "Parça Süresi (ms)"
39 FROM track
40 JOIN album ON track.album_id = album.album_id
41 JOIN artist ON album.artist_id = artist.artist_id
42 WHERE album.title = 'Let There Be Rock'
43 ORDER BY track.milliseconds DESC;
44
45 /*
46 track.name kolonunu "Parça Adı" olarak,
47 artist.name "Sanatçı Adı" olarak,
48 track.milliseconds kolonunu "Parça Süresi (ms)" olarak
49 seçtik ve adlandırdık.
50 WHERE komutu ile album tablosundan title'ı 'Let There Be Rock'
51 olan değerleri seçerek JOIN ile tabloları
52 verilen kolonlar üzerinden birleştirdik.
53 Parça süresine göre, DESC komutu kullanak azalan yönde ORDER BY ile sıraladık.
54 */
```

Data Output Messages Notifications

≡

📄

▼

🗑️

▼

🗑️

📄

⬇️

📈

SQL

	Parça Adı character varying (200) 🔒	Sanatçı Adı character varying (120) 🔒	Parça Süresi (ms) integer 🔒
1	Overdose	AC/DC	369319
2	Let There Be Rock	AC/DC	366654
3	Go Down	AC/DC	331180
4	Problem Child	AC/DC	325041
5	Whole Lotta Rosie	AC/DC	323761
6	Bad Boy Boogie	AC/DC	267728
7	Hell Ain't A Bad Place To Be	AC/DC	254380
8	Dog Eat Dog	AC/DC	215196

### pgAdmin4 Uygulamasında Kullandığım Sorgular (Yorumlarıyla Birlikte):

- 1) Select \* from invoice  
  
SELECT SUM(total)  
FROM invoice  
WHERE billing\_country = 'USA'  
AND EXTRACT(YEAR FROM invoice\_date) = 2009;  
  
/\*  
WHERE komutu ile billing\_country kolonunda USA değerlerini filtreledik.  
EXTRACT komutu ile 2009 olan değerleri ayıkladık,  
(AND komutunda iki şartın da sağlanması beklenir)  
SUM komutu ile total kolonunun değerini topladık.  
\*/

- 2) Select \* from track  
Select \* from playlisttrack  
Select \* from playlist

```
SELECT
t.*,                                --bu ifade ile track tablosundaki tüm kolonları seçtik.
pt.playlist_id,                     --bu ifade ile playlisttrack tablosundaki playlist_id kolonunu seçtik
p.name AS playlist_type             --bu ifade, playlist tablosundaki name kolonunu seçtik
--ve playlist_type olarak adlandırdık.
```

```
FROM track t JOIN playlisttrack pt ON t.track_id = pt.track_id
JOIN playlist p ON pt.playlist_id = p.playlist_id;
--JOIN ile tablolar verilen kolonlar üzerinden birleştirdik.
```

- 3) Select \* from track  
Select \* from album  
Select \* from artist

```
SELECT
track.name AS "Parça Adı",
artist.name AS "Sanatçı Adı",
track.milliseconds AS "Parça Süresi (ms)"
FROM track
JOIN album ON track.album_id = album.album_id
JOIN artist ON album.artist_id = artist.artist_id
WHERE album.title = 'Let There Be Rock'
ORDER BY track.milliseconds DESC;
```

```
/*
    track.name kolonunu "Parça Adı" olarak,
    artist.name "Sanatçı Adı" olarak,
    track.milliseconds kolonunu "Parça Süresi (ms)" olarak seçtik ve adlandırdık.
    WHERE komutu ile album tablosundan title'ı 'Let There Be Rock'
    olan değerleri seçerçerek JOIN ile tabloları
    verilen kolonlar üzerinden birleştirdik.
    Parça süresine göre, DESC komutu kullanak azalan yönde ORDER BY ile sıraladık.
*/
```