

1. Invoice tablosunda, tüm değerleri NULL olan kayıtların sayısını bulmanız isteniyor.

Bu işlemi tek bir sorgu ile yapmalısınız. Sorguyu yazdıktan sonra, PostgreSQL'in sol alt kısmındaki Row sayısını, SQL sorgunuzda yorum satırında belirtmeniz gerekmektedir.

```
SELECT COUNT(*) AS null_kayit_sayisi
FROM Invoice
WHERE invoice_id IS NULL
AND customer_id IS NULL
AND invoice_date IS NULL
AND billing_address IS NULL
AND billing_city IS NULL
AND billing_state IS NULL
AND billing_country IS NULL
AND billingpostal_code IS NULL
AND total IS NULL;
```

Count fonksiyonu ile değerleri Null olan kayıtların sayısını buluruz.” Where “ ile Null olmasını kontrol ederiz.

2. Koordinasyondaki kişiler, Total değerlerinde bir hata olduğunu belirtiyorlar. Bu değerlerin iki katını görmek ve eski versiyonlarıyla birlikte karşılaştırmak için bir sorgu yazmanız isteniyor. Ayrıca, verilerin daha rahat takip edilebilmesi için, tablonun yeni versiyonuna ait kolona göre büyükten küçüğe sıralama yapılması isteniyor.

```
SELECT invoice_id, customer_id, total AS eski_total, total * 2 AS yeni_total
FROM Invoice
ORDER BY yeni_total DESC;
```

total ve 2 katını (yeni\_total olarak) getiriyoruz ve Order BY DESC fonksiyonu ile yeni totale göre büyükten küçüğe sıralarız.

3. Adres kolonundaki verileri, soldan 3 karakter ve sağdan 4 karakter olarak birleştirmeniz ve "Açık Adres" olarak yazmanız isteniyor. Ayrıca, bu yeni açık adresi 2013 yılı ve 8. ay'a göre filtrelemeniz gerekiyor.

```
SELECT invoice_id,
       LEFT(billing_address, 3) || ' ' || RIGHT(billing_address, 4) AS "Açık Adres",
       invoice_date
FROM Invoice
WHERE EXTRACT(YEAR FROM invoice_date) = 2013
AND EXTRACT(MONTH FROM invoice_date) = 8;
```

LEFT ve RIGHT fonksiyonları ile soldan 3 ve sağdan 4 karakter olarak birleştirir. “ WHERE “ ile 2013 yılı ve 8. ay'a göre filtreleriz.