

Ödev Çözümlerim

Soru 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.

Cevap 1

“select * from invoice” ifadesi invoice tablosundaki tüm verileri getirir.

“count” fonksiyonu sayıcı görevi görür. “as” ifadesi alias’tır, takma ad olarak kullanılır.

where ifadesi ile istediğim koşulları yazdım.

bu yazdığım koşulların “from” ifadesi ile invoice tablosundan gelmesini istedim.

“is null” ifadesi bir değerin null olup olmadığını sorgular.

bu sorgu sonucunda bana 0 satır döndü çünkü bu tabloda tüm verileri null olan bir kayıt yok.

The screenshot shows the pgAdmin 4 interface with a query editor and a results table.

Query Editor:

```
1 select * from invoice
2 --rows: 0
3 select count(*) as kolon_sayisi from invoice
4 where invoice_id is null
5 and customer_id is null
6 and invoice_date is null
7 and billing_address is null
8 and billing_city is null
9 and billing_state is null
10 and billing_country is null
11 and billing_postal_code is null
12 and total is null ;
13
```

Data Output:

kolon_sayisi	bigint
1	0

Total rows: 1 Query complete 00:00:00.105 CRLF

Soru 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 küçükten büyüğe sıralama yapılması isteniyor.

Cevap 2

select ifadesi ile istediğim sütunları seçtim ve as ifadesi ile istediğim sütuna takma ad verdim ve from ile bu seçtiğim sütunların geleceği tabloyu belirttim.

where ile totalın null olmadığını kayıtların gelmesini istedim.

“order by” ile istediğim sütunu “asc” ile küçükten büyüğe sıraladım. “asc” ifadesini yazmasam da olurdu çünkü order by kullandığında default değer asc’dir.

The screenshot shows the pgAdmin 4 interface. The top navigation bar includes 'Dashboard', 'Properties', 'SQL', 'Statistics', 'Dependencies', 'Dependents', 'Processes', and a connection tab for 'postgres/postgres@ödev1'. Below the tabs is a toolbar with various icons. The main area has two tabs: 'Query' (selected) and 'Scratch Pad'. The 'Query' tab contains the following SQL code:

```
1 select invoice_id, customer_id, total as eski_total, (total*2) as yeni_total
2   from invoice
3  where total is not null
4  order by yeni_total asc;
```

The 'Data Output' tab below displays the results of the query. The table has four columns: invoice_id, customer_id, eski_total, and yeni_total. The data is as follows:

	invoice_id [PK] integer	customer_id integer	eski_total numeric (10,2)	yeni_total numeric
1	104	38	0.99	1.98
2	174	5	0.99	1.98
3	69	25	0.99	1.98
4	300	40	0.99	1.98
5	216	56	0.99	1.98
6	181	43	0.99	1.98
7	314	57	0.99	1.98
8	111	17	0.99	1.98
9	377	45	0.99	1.98

Total rows: 412 Query complete 00:00:00.120 CRLI

Soru 3

Adres kolonundaki verileri, soldan 3 karakter ve sağdan 4 karakter alarak birleştirmeniz ve "Açık Adres" olarak yazmanız isteniyor. Ayrıca, bu yeni açık adresi 2013 yılı ve 10. ay'a göre filtrelemeniz gerekiyor.

Cevap 3

select ifadesi ile istedigim sütunları seçtim.

"substring" ifadesi ile bu kayıtta istedigim indeksten başlayıp istedigim indekse kadar olan değerleri döndürdüm. İlk substring ifadesi soldan 3 karakter almaya yarar. İkinci substring ifadesi length ile billing_addressin karakter uzunluğunu alır ve bundan 3 eksiltir ve bu sayede kaçinci karakter başlamam gerektiğini bana verir ve sonrasında sağdan 4 karakter aldım.

” || “ ile bu ikisini birleştirdim

where ile istedigim tarih aralığının koşulunu yazdım.

The screenshot shows the pgAdmin 4 interface with a query editor and a data output viewer.

Query Editor:

```
1 ✓ select invoice_id, billing_address,
2   substring(billing_address from 1 for 3) || substring(billing_address from length(billing_address)- 3 for 4) as acik_adres
3   from invoice
4   where invoice_date >= '2013-10-01'
5   and invoice_date < '2013-11-01' ;
```

Data Output:

	invoice_id	billing_address	acik_adres
1	396	627 Broadway	627dway
2	392	Ullevålsveien 14	Ulln 14
3	393	Rilsk 3174/6	Ril74/6
4	394	Grøtrystraat 63	Grøt 63
5	395	Praça Pio X, 119	Pra 119
6	397	1033 N Park Ave	103 Ave
7	398	11, Place Bellecour	11,cour

Total rows: 7 | Query complete 00:00:00.085

Bu ise farklı bir yoldan çözümü eğitime dahil olduğu için bunu da kullanmak istedim. yine select ile istediğim sütunları seçtim ve bu sefer “concat” kullandım. Bu da birleştirmeye yarıyor. İki ifade yazdım birisi soldan birisi sağdan karakterler alıyor ve concat bunları birleştiriyor. Üstteki çözümde yaptığım gibi yine where ile tarih sorgusu yaptım.

The screenshot shows the pgAdmin 4 interface. At the top, there's a navigation bar with tabs for Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes, and a connection tab for 'postgres/postgres@ödev1*'. Below the navigation bar is a toolbar with various icons for database management. The main area is divided into two sections: 'Query' and 'Data Output'.

Query:

```
1 v select invoice_id,billing_address,
2 concat(
3 left(billing_address,3),
4 right(billing_address,4)
5 ) as acik_adres
6 from invoice
7 where invoice_date >= '2013-10-01'
8 and invoice_date < '2013-11-01' ;
```

Data Output:

	invoice_id [PK] integer	billing_address character varying (70)	acik_adres text
1	396	627 Broadway	627dway
2	392	Ullevålsveien 14	Ulln 14
3	393	Rilsk 3174/6	Ril74/6
4	394	Grøntrystraat 63	Grøt 63
5	395	Praha Plo X, 119	Pra 119
6	397	1033 N Park Ave	103 Ave
7	398	11, Place Bellecour	11,cour

Total rows: 7 Query complete 00:00:00.061 CRLF