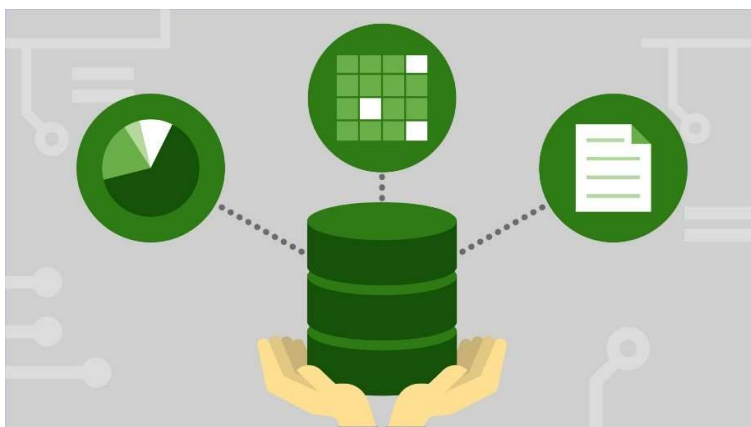


به نام خدا



دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده برق و کامپیوتر



آزمایشگاه پایگاه داده

دستور کار شماره ۴

پرنیان فاضل

۸۱۰۱۹۸۵۱۶

بهار ۱۴۰۲

ایجاد و ایمپورت داده‌های دیتابیس Northwind

مطابق مراحل گفته شده در دستور کار دیتابیس را ایمپورت کردم. برای نمونه داده‌های جدول categories به صورت زیر است:

category_id	category_name	description	picture
1	Beverages	Soft drinks, coffees, teas, beers, and ales	
2	Condiments	Sweet and savory sauces, relishes, spreads, and dressings	
3	Confections	Desserts, candies, and sweet breads	
4	Dairy Products	Cheeses	
5	Grains/Cereals	Breads, crackers, pasta, and cereal	
6	Meat/Poultry	Prepared meats	
7	Produce	Dried fruit and bean curd	
8	Seafood	Seaweed and fish	

بخش ۱

همه خریدارانی که سفارشی در سال ۱۹۹۸ ثبت کرده اند.

برای پیدا کردن این خریداران در واقع باید روی فیلد date شرط بگذاریم که در سال ۱۹۹۸ باشد. میتوان از between استفاده کرد.

من از روش زیر برای پیدا کردن این خریداران استفاده کردم (هر دو روش درست است):

```
--1
--SELECT DISTINCT c.customer_id, c.contact_name
--FROM orders o, customers c
--WHERE o.order_date >= '1998-01-01' and o.order_date < '1999-01-01' and o.customer_id = c.customer_id;

--1
SELECT DISTINCT c.customer_id, c.contact_name
FROM orders o, customers c
WHERE o.order_date between date('1998-01-01') and date('1998-12-31') and o.customer_id = c.customer_id;
```

customer_id	contact_name
1	LACOR Daniel Tonini
2	THERBI Liz Nixon
3	BERGS Christina Berglund
4	RICAR Janete Limeira
5	QUEEN Lúcia Carvalho
6	ISLAT Helen Bennett
7	QUEDE Bernardo Batista
8	EASTC Ann Devon
9	LILAS Carlos González
10	COMMI Pedro Afonso
11	CACTU Patricio Simpson
12	BSBEV Victoria Ashworth
13	HUNGO Patricia McKenna
14	SANTG Jonas Bergulfsen
15	PERIC Guillermo Fernández
16	SAVEA Jose Pavarotti
17	CONSH Elizabeth Brown
18	CDERN Dominique Perrier

* با توجه به اینکه گفته نشده بود دقیقاً چه اطلاعاتی مد نظر است من customer_id و contact_name را نشان دادم. خروجی شامل ۸۱ سطر است که بخشی از نشان داده شد.

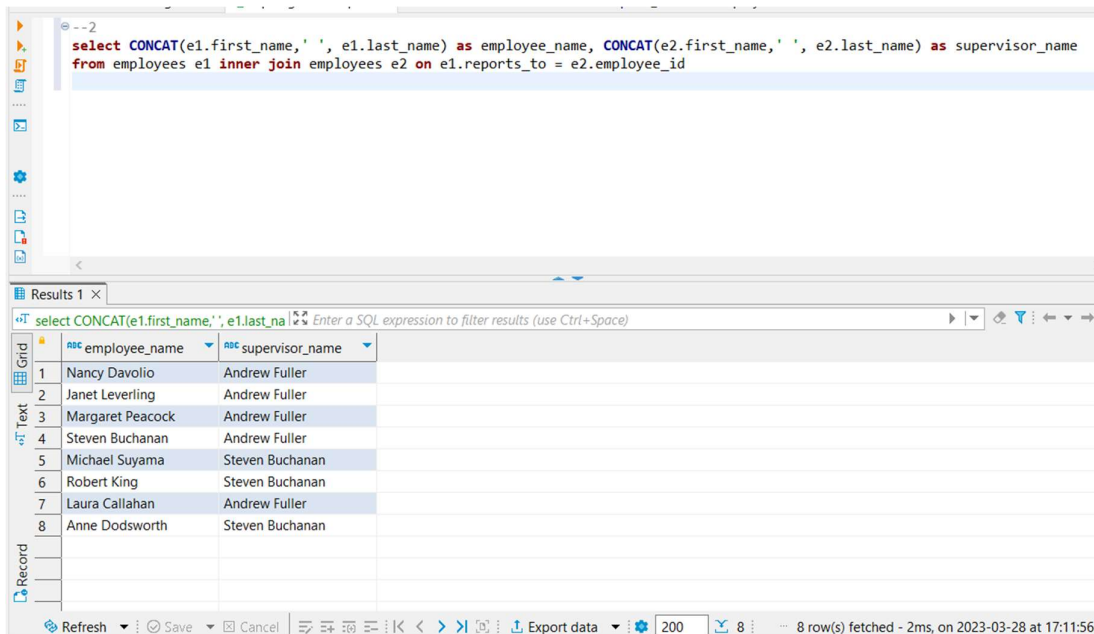
بخش ۲

لیست همه کارمندان (employees) را به همراه ناظران (supervisors) آنها را بدست آورید. نام و نام خانوادگی افراد باید با یک فاصله بهم دیگر متصل شده باشد.

با استفاده از دستور زیر و inner join که در دستور کار گفته شده بود، داده‌های مربوطه را پیدا می‌کنیم.

```
--2
select CONCAT(e1.first_name,' ', e1.last_name) as employee_name, CONCAT(e2.first_name,' ', e2.last_name) as supervisor_name
from employees e1 inner join employees e2 on e1.reports_to = e2.employee_id
```

خروجی به صورت زیر است:



The screenshot shows a SQL query execution interface. The query is:


```
--2
select CONCAT(e1.first_name,' ', e1.last_name) as employee_name, CONCAT(e2.first_name,' ', e2.last_name) as supervisor_name
from employees e1 inner join employees e2 on e1.reports_to = e2.employee_id
```

 The results are displayed in a table grid with 8 rows and 2 columns: employee_name and supervisor_name. The data is as follows:

employee_name	supervisor_name
Nancy Davolio	Andrew Fuller
Janet Leverling	Andrew Fuller
Margaret Peacock	Andrew Fuller
Steven Buchanan	Andrew Fuller
Michael Suyama	Steven Buchanan
Robert King	Steven Buchanan
Laura Callahan	Andrew Fuller
Anne Dodsworth	Steven Buchanan

 The interface also shows a status bar at the bottom indicating '8 row(s) fetched - 2ms, on 2023-03-28 at 17:11:56'.

بخش ۳

میانگین کل هر سفارش را به ازای هر مشتری در خروجی به صورت نزولی نمایش دهید.
ابتدا مقدار هزینه کل هر سفارش را برای مشتریان پیدا کرده و آن را با استفاده از AVG() میانگین گرفتم و در نهایت آن را به صورت نزولی مرتب کردم. خروجی ۸۹ سطر دارد و به صورت زیر است:

```
-- 3
select t1.customer_id, AVG(t1.total_order_cost) as average
from (
  select o.customer_id, (od.unit_price * od.quantity * (1 - od.discount)) as total_order_cost
  from orders o, order_details od, customers c
  where o.order_id = od.order_id and o.customer_id = c.customer_id
) as t1
group by t1.customer_id
order by average desc;
```

customer_id	average
QUICK	1,282.2942445395
SIMOB	1,121.1398408137
ERNSH	1,028.1860602321
HANAR	1,026.2928108893
PICCO	1,005.6026093467
HUNGO	908.7255469363
MEREP	902.2559423787
SAVEA	899.6719787965
GREAL	841.2477259512
KOENE	792.5226633984
FOLIG	729.181255579
RATTC	719.6873356094
BLONP	712.849233821
BOLID	705.4750031034
EASTC	702.906425249
WHITC	684.0901225053
FOLKO	657.0569442228
RICSU	644.7926306774
QUEEN	642.9374401243
OLDWO	632.3942709733
SEVES	623.666347851

بخش ۴

ده مشتری برتر با بیشترین میزان درآمد را مشخص کنید. مشخصات مشتری به همراه کمپانی و مجموع درآمد باید در خروجی گزارش شود.

* طبق پیام گروه تلگرامی، منظور از درآمد همان هزینه مصرفی برای محصول است و این فرض برای حل سوال در نظر گرفته شده است. ابتدا مجموع هزینه هر سفارش را محاسبه کردم (جدول t1). سپس با گروه بندی بر اساس customer_id مجموع هزینه‌ی هر مشتری را محاسبه کردم و بعد داده‌ها را مرتب کرده و ۱۰ مشتری که بیشترین هزینه (درآمد) را داشتند، انتخاب کردم. در نهایت اطلاعات خواسته شده صورت سوال (اطلاعات مشتری و کمپانی و مجموع هزینه (درآمد)) را نشان دادم. خروجی به صورت زیر است:

```
--4
select c1.*, t2.total_customer_payment
from (
  select t1.customer_id, SUM(t1.total_order_cost) as total_customer_payment
  from (
    select o.customer_id, (od.unit_price * od.quantity * (1 - od.discount)) as total_order_cost
    from orders o, order_details od
    where o.order_id = od.order_id
  ) as t1
  group by t1.customer_id
  order by total_customer_payment desc
  limit 10
) as t2, customers c1
where c1.customer_id = t2.customer_id
```

	customer_id	company_name	contact_name	contact_title	address	city	region	postal_code	country	phone	fax	total_customer_payment
1	ERNSH	Ernst Handel	Roland Mendel	Sales Manager	Kirchgasse 6	Graz	[NULL]	8010	Austria	7675-3425	7675-3426	104,874,978,143,677,5
2	FOLKO	Folk och fa HB	Maria Larsson	Owner	Akergatan 24	Bracke	[NULL]	S-844 67	Sweden	0695-34 67 21	[NULL]	29,567,562,490,026,7
3	HANAR	Hanari Carnes	Mario Pontes	Accounting Manager	Rua do Paço, 67	Rio de Janeiro	[NULL]	05454-876	Brazil	(21) 555-0091	(21) 555-0876	32,841,369,948,457,5
4	HUNGO	Hungry Owl All-Night Gr	Patricia McKenna	Sales Associate	8 Johnstown Road	Cork	[NULL]	[NULL]	Ireland	2967 542	2967 3333	49,979,905,081,495,5
5	KOENE	Königlich Essen	Philip Cramer	Sales Associate	Maubelstr. 90	Brandenburg	[NULL]	14776	Germany	0555-09876	[NULL]	30,908,383,872,538,4
6	MEREP	Mère Paillard	Jean Fresnière	Marketing Assistant	43 rue St. Lauren	Montréal	[NULL]	H1J 1C3	Canada	(514) 555-8054	(514) 555-8054	28,872,190,156,118,4
7	QUICK	QUICK-Stop	Horst Kloss	Accounting Manager	Taucherstraße 1c	Cunewalde	[NULL]	01307	Germany	0372-035188	[NULL]	110,277,305,030,393,8
8	RATTIC	Rattlesnake Canyon Groc	Paula Wilson	Assistant Sales Repre	2817 Milton Dr.	Albuquerque NM	[NULL]	87110	USA	(505) 555-5939	(505) 555-3939	51,097,800,828,262,2
9	SAVEA	Save-a-lot Markets	Jose Pavarotti	Sales Representative	187 Suffolk Ln.	Boise	[NULL]	83720	USA	(208) 555-8097	[NULL]	104,361,949,540,394
10	WHITC	White Clover Markets	Karl Jablonski	Owner	305 - 14th Ave. S	Seattle	[NULL]	98128	USA	(206) 555-4112	(206) 555-4112	27,363,604,900,210,8

بخش ۵

لیست کارمندانی که حداقل یک گزارش دارند به همراه تعداد گزارش‌ها نمایش دهید. کارمندانی که گزارش گرفته‌اند را از reports_to پیدا کردم. دقت شود که نباید null باشد. سپس آن‌ها را گروه‌بندی کرده و تعداد گزارش‌ها را با استفاده از COUNT() محاسبه می‌کنیم. خروجی ۲ سطر دارد و به صورت زیر است (شناسه کارمند (reports_to) به همراه تعداد گزارش (count):

The screenshot shows a PostgreSQL query editor with the following SQL query:

```
select e1.reports_to, COUNT(e1.reports_to)
from employees e1
where e1.reports_to is not null
group by e1.reports_to;
```

The results are displayed in a table with two columns: reports_to and count.

reports_to	count
1	5
2	5

بخش ۶

۵ محصولی که در یک دسته بندی مشخص (مثلا category_id = 5) بیشترین میزان فروش را داشته‌اند را مشخص کنید. باید خروجی بر حسب میزان فروش به صورت صعودی مرتب شده باشد. ابتدا محصولات مربوط به دسته با شناسه ۵ را پیدا کردم و سپس بر اساس شناسه محصول گروه‌بندی کردم و بعد با استفاده از SUM() روی فیلد quantity، مجموع تعداد فروش آن محصول را محاسبه کردم. در صورت سوال خواسته شده به صورت صعودی مرتب شود. ابتدا ۵ محصول پر فروش را پیدا کرده و به عنوان جدولی به نام t1 در نظر گرفتم و بعد این ۵ محصول را به صورت صعودی مرتب می‌کنیم. خروجی به صورت زیر است:

The screenshot shows a PostgreSQL query editor with the following SQL query:

```
select *
from (select od.product_id, SUM(od.quantity) as total_order_count
from order_details od, products p
where od.product_id = p.product_id and p.category_id = 5
group by od.product_id
order by total_order_count desc
limit 5
) as t1
order by t1.total_order_count asc;
```

The results are displayed in a table with two columns: product_id and total_order_count.

product_id	total_order_count
52	500
23	580
42	697
64	740
56	1,263

بخش ۷

لیست محصولاتی که توسط بیشتر از ۲۰ مشتری سفارش داده شده‌اند را به همراه تعدادی که سفارش داده شده و قیمت هر unit را در خروجی نشان دهید.

* من فرض کردم منظور سوال ۲۰ مشتریان متفاوت نبوده است.

ابتدا جداول orders و order_details را جوین کرده و سپس بر اساس product_id گروه‌بندی کردم و شرط تعداد خریداران بیشتر از ۲۰ را با استفاده از having بر روی count() قرار دادم. در نهایت داده‌های خواسته شده در صورت سوال را نشان دادم. خروجی ۵۲ سطر دارد و به صورت زیر است:

```

select t1.*, p.unit_price
from (
  select od.product_id, count(o.customer_id) as order_count
  from orders o, order_details od
  where o.order_id = od.order_id
  group by (od.product_id)
  having count(o.customer_id) > 20
) as t1, products p
where p.product_id = t1.product_id;

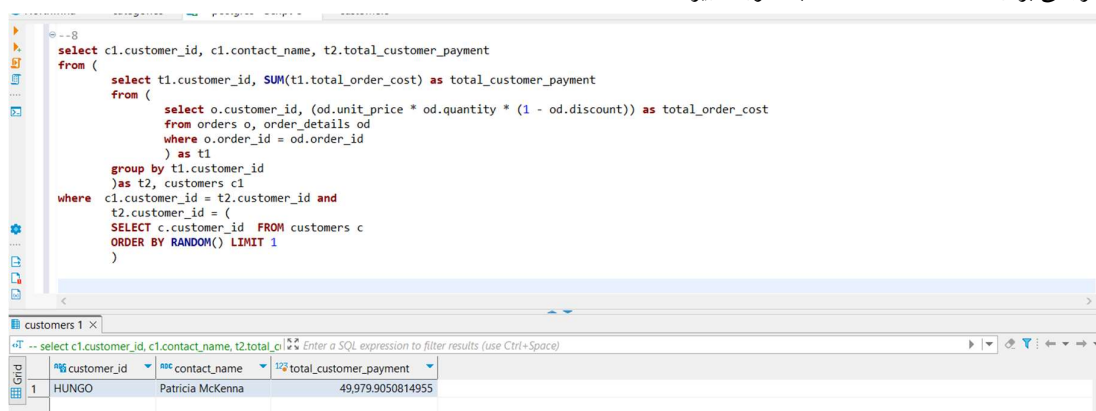
```

product_id	order_count	unit_price
1	1	38
2	2	44
3	7	29
4	10	33
5	11	38
6	13	40
7	14	22
8	16	43
9	17	37
10	18	27
11	19	37
12	21	39
13	24	51
14	26	32
15	28	33
16	29	32
17	30	32
18	31	51
19	32	32
20	33	32
21	34	32
22	35	32
23	36	32
24	37	32
25	38	32
26	39	32
27	40	32
28	41	32
29	42	32
30	43	32
31	44	32
32	45	32
33	46	32
34	47	32
35	48	32
36	49	32
37	50	32
38	51	32
39	52	32
40	53	32
41	54	32
42	55	32
43	56	32
44	57	32
45	58	32
46	59	32
47	60	32
48	61	32
49	62	32
50	63	32
51	64	32
52	65	32

بخش ۸

مجموع درآمدی که توسط یک مشتری با آیدی رندم بدست آمده است را نشان دهید. در خروجی نام و نام خانوادگی مشتری به همراه مجموع درآمد را نشان دهید.

* طبق پیام گروه تلگرامی، منظور از درآمد همان هزینه مصرفی برای محصول است و این فرض برای حل سوال در نظر گرفته شده است. همچنین فرض کردم که نام و نام خواندگی در ستون contact_name برای مشتری قرار دارد. برای ساخت شناسه رندوم از RANDOM() استفاده کردم. همچنین از کدی که برای سوال ۴ زده بودم استفاده کردم و آن سطری را انتخاب کردم که با استفاده از تابع رندوم ایجاد شده بود. در نهایت ستونهای خواسته شده سوال را نشان دادم. برای نمونه، خروجی برای شناسه "HUNGO" به صورت زیر است:



```

select c1.customer_id, c1.contact_name, t2.total_customer_payment
from (
  select t1.customer_id, SUM(t1.total_order_cost) as total_customer_payment
  from (
    select o.customer_id, (od.unit_price * od.quantity * (1 - od.discount)) as total_order_cost
    from orders o, order_details od
    where o.order_id = od.order_id
  ) as t1
  group by t1.customer_id
) as t2, customers c1
where c1.customer_id = t2.customer_id and
t2.customer_id = (
  SELECT c.customer_id FROM customers c
  ORDER BY RANDOM() LIMIT 1
)

```

	customer_id	contact_name	total_customer_payment
1	HUNGO	Patricia McKenna	49,979,905,081,4955

بخش ۹

لیست ۵ مشتری که در سال گذشته هیچ اوردری ثبت نکرده‌اند را بدست آورید (بر حسب مدت زمان سفارش ندادن به صورت نزولی نشان دهید).

من با استفاده از with ابتدا جدول customers_with_no_order_in_the_last_year را تعریف می‌کنم. این جدول را به این صورت تعریف کردم که مشتریانی هیچ اوردری ثبت نکرده‌اند را پیدا کرده و همچنین مشتریانی که اوردر ثبت کرده‌اند اما از آخرین ثبت سفارش آن‌ها یک سال می‌گذرد را پیدا کرده و union می‌گیریم. مشتریانی که اوردر ثبت کرده‌اند اما از آخرین ثبت سفارش آن‌ها یک سال می‌گذرد را به این صورت پیدا می‌کنیم که اختلاف تاریخ آخرین ثبت سفارش موجود در کل دیتابیس را با تاریخ آخرین ثبت سفارش هر مشتری محاسبه کرده (group by روی شناسه مشتریان) و چک می‌کنیم که از ۱ سال بیشتر است یا نه، که این کار را با 'interval' انجام دادم. در نهایت این اطلاعات را بر اساس تاریخ آخرین ثبت سفارش آن‌ها به صورت صعودی مرتب می‌کنم. توجه شود که چون فرض کردم کسانی که هیچ سفارشی ثبت نکرده‌اند را هم باید در نظر بگیرم، در order by، nulls first را هم قرار دادم چون منطقاً کسانی که هیچ سفارشی ثبت نکرده‌اند باید بالاتر از همه تاریخ ثبت سفارش‌ها قرار بگیرند. و در نهایت با 5 limit این ۵ مشتری را نشان دادم. کد مربوطه و خروجی به صورت زیر است:

```
--9
with customers_with_no_order_in_the_last_year as(
  select o.customer_id, max(o.order_date) as last_order
  from customers c left join orders o on c.customer_id = o.customer_id
  group by o.customer_id
  having extract(year from MAX(o.order_date)) <= extract(year from (select max_order_date - INTERVAL '1 year'
                                                                    from (
                                                                      select MAX(order_date) as max_order_date
                                                                      from orders) as last_order_date
                                                                    ))
)
union
select c.customer_id, o.order_date
from customers c
left join orders o on c.customer_id = o.customer_id
where o.order_id is null
)
select customer_id, last_order
from customers_with_no_order_in_the_last_year
order by last_order asc nulls first
limit 5;
```

customer_id	last_order
PARIS	[NULL]
FISSA	[NULL]
CENTC	1996-07-18
LAZYK	1997-05-22
HUNGC	1997-09-08

* اگر فرض طراح سوال بر این بوده که مشتریانی که هیچ ثبت سفارشی نداشته‌اند را نشان ندهیم من همه‌ی این مشتریان را در زیر نشان دادم تا در صورت نیاز بررسی شود:

with customers_with_no_order_in_the_last_year		
	customer_id	last_order
1	PARIS	[NULL]
2	FISSA	[NULL]
3	CENTC	1996-07-18
4	LAZYK	1997-05-22
5	HUNGC	1997-09-08
6	MEREP	1997-10-30
7	FAMIA	1997-10-31
8	VINET	1997-11-12
9	GROSR	1997-12-18
10	FOLIG	1997-12-22

بخش ۱۰

می‌خواهیم سفارش جدیدی برای محصولاتی که تعداد آنها (UnitsInStock) از حد مجاز (ReorderLevel) کمتر شده است، بدهیم. این محصولات را بیابید و آنها بر اساس میزان موجودی (به صورت صعودی) نمایش دهید. یک ویو برای ذخیره این دستور ایجاد کنید که همواره برای مشاهده محصولاتی که نیاز به خرید دارند، این ویو، استفاده شود. یک ویو با نام insufficient_products تعریف کردم که در آن محصولاتی که موجودی آنها از reorder_level کمتر است نشان داده می‌شوند و خروجی را بر اساس موجودی به صورت صعودی مرتب می‌کنیم. خروجی ۱۸ سطر دارد و به صورت زیر است:

The screenshot shows a SQL IDE interface. The top pane contains the following SQL code:

```
--10
create or replace view insufficient_products as
(
    select p
    from products p
    where p.units_in_stock < p.reorder_level
    order by p.units_in_stock asc
);
select * from insufficient_products;
```

The bottom pane displays the results of the query in a grid format. The grid has 10 columns: product_id, product_name, supplier_id, category_id, quantity_per_unit, unit_price, units_in_stock, and units_on_order. The data is sorted by units_in_stock in ascending order.

	product_id	product_name	supplier_id	category_id	quantity_per_unit	unit_price	units_in_stock	units_on_order
1	31	Gorgonzola Telino	14	4	12 - 100 g pkgs	12.5	0	
2	21	Sir Rodney's Scones	8	3	24 pkgs. x 4 pieces	10	3	
3	74	Longlife Tofu	4	7	5 kg pkg.	10	4	
4	66	Louisiana Hot Spiced C	2	2	24 - 8 oz jars	17	4	
5	45	Rogede sild	21	8	1k pkg.	9.5	5	
6	68	Scottish Longbreads	8	3	10 boxes x 8 pieces	12.5	6	
7	32	Mascarpone Fabioli	14	4	24 - 200 g pkgs.	32	9	
8	30	Nord-Ost Matjeshering	13	8	10 - 200 g glasses	25.88999939	10	
9	49	Maxilaku	23	3	24 - 50 g pkgs.	20	10	

The bottom status bar indicates that 18 rows were fetched.

بخش ۱۱

بر درآمد ترین ماه را با در نظر گرفتن تعداد سفارش ها و مبلغ آنها را بدست آورید. (اطلاعات ماه را به همراه تعداد سفارش و مجموع مبلغ سفارش ها نمایش دهید).

* من منظور از "تعداد سفارش" که در خروجی خواسته شده را متوجه نشدم که منظور مجموع quantity های سفارش هاست یا تعداد هر سفارش و بنابراین هر دو (orders_count و total_quantity_sold) را در خروجی نشان دادم.

مجموع درآمد در هر ماه با در نظر گرفتن discount به صورت زیر است:

Grid	123 mon	123 orders_count	123 total_quantity_sold	123 total_month_income
1	1	88	5,867	155,480.1803749143
2	2	83	5,247	137,898.9223333325
3	3	103	5,835	143,401.3751098838
4	4	105	6,592	176,831.6346444887
5	5	46	3,085	72,114.9202573343
6	6	30	1,635	36,362.8023348025
7	7	55	3,516	78,882.7526482664
8	8	58	3,183	72,772.9447589985
9	9	60	3,467	82,010.6425343369
10	10	64	4,417	104,264.950721206
11	11	59	3,591	89,133.8539787021
12	12	79	4,882	116,638.0589570982

ابتدا با استفاده از دستور `EXTRACT(MONTH FROM o.order_date)` ماه هر سفارش را استخراج کرده و آن را در ستون جداگانه‌ای به نام `mon` قرار دادم. همچنین مجموع درآمد هر سفارش هم در ستون دیگری به نام `total_order_cost` قرار دادم. و نام این را جدول جدید را `t1` گذاشتم. در نهایت با استفاده از گروه‌بندی براساس ماه و محاسبه مجموع هزینه‌های سفارش هر ماه با استفاده از `SUM()` و همچنین مرتب کردن به صورت نزولی و انتخاب اولین رکورد، پردرآمدترین ماه را پیدا کردم. پردرآمدترین ماه، ماه ۴ است. خروجی به صورت زیر است:

```
--11
select t1.mon, count(distinct t1.order_id) as orders_count, SUM(quantity) as total_quantity_sold, SUM(t1.total_order_cost) as total_month_income
from(
    select o.order_id, o.customer_id, EXTRACT(MONTH FROM o.order_date) as mon,
           od.quantity, (od.unit_price * od.quantity * (1 - od.discount)) as total_order_cost
    from orders o, order_details od
    where o.order_id = od.order_id
) as t1
group by t1.mon
order by total_month_income desc
limit 1;
```

بخش ۱۲

مشتریانی را بیابید که شماره فکس آنها در سیستم موجود نیست.
 مشتریانی که فیلد fax آن‌ها null است را پیدا می‌کنیم. خروجی ۲۲ سطر دارد و به صورت زیر است:

customers 1 x

select * from customers c where c.fax is null

	customer_id	company_name	contact_name	contact_title	address	city	region	postal_code	country
1	ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner	Mataderos 2312	México D.F.	[NULL]	05023	Mexico
2	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus	London	[NULL]	EC2 5NT	UK
3	CHOPS	Chop-suey Chinese	Yang Wang	Owner	Hauptstr. 29	Bern	[NULL]	3012	Switzerland
4	COMMI	Comércio Mineiro	Pedro Afonso	Sales Associate	Av. dos Lusíadas, 1	Sao Paulo	SP	05432-043	Brazil
5	FAMIA	Familia Arquibaldo	Aria Cruz	Marketing Assistant	Rua Orós, 92	Sao Paulo	SP	05442-030	Brazil
6	FOLKO	Folk och få HB	Maria Larsson	Owner	Åkergatan 24	Bräcke	[NULL]	S-844 67	Sweden
7	GODOS	Godos Cocina Típica	José Pedro Freyre	Sales Manager	C/ Romero, 33	Sevilla	[NULL]	41101	Spain
8	GOURL	Gourmet Lanchonetes	André Fonseca	Sales Associate	Av. Brasil, 442	Campinas	SP	04876-786	Brazil
9	GREAL	Great Lakes Food Market	Howard Snyder	Marketing Manager	2732 Baker Blvd.	Eugene	OR	97403	USA
10	ISLAT	Island Trading	Helen Bennett	Marketing Manager	Garden House Crc	Cowes	Isle of Wight	PO31 7PJ	UK
11	KOENF	Königlich Essen	Philipp Cramer	Sales Associate	Maubelstr. 90	Brandenburg	[NULL]	14776	Germany

Refresh Save Cancel Export data 200 22 22 row(s) fetched

بخش ۱۳ (امتیازی)

جفت محصولاتی که همراه هم بیشتر از ۴ بار به فروش رفته‌اند.

* برای حل سوال فرض شده است که تعداد سفارش مد نظر است و نه quantity آن‌ها.

برای پیدا کردن این جفت محصولات، از order_details استفاده می‌کنیم. به این صورت که دو جدول order_details را جویین کرده و آن‌هایی که شناسه محصول‌شان متفاوت است را پیدا کرده و با استفاده از having، شرط مجموع تعداد محصولات بیشتر از ۴ را اعمال می‌کنیم. شرط $od1.product_id < od2.product_id$ برای این گذاشته است که از دادن اطلاعات تکراری و اشتباه جلوگیری شود. خروجی به صورت زیر است و ۱۸ سطر دارد.

```
--13
select od1.product_id as product_id1, od2.product_id as product_id2, COUNT(*) as cnt
from order_details od1, order_details od2
where od1.order_id = od2.order_id and od1.product_id < od2.product_id
group by od1.product_id , od2.product_id
having COUNT(*) > 4;
```

Grid	product_id1	product_id2	cnt
1	16	62	6
2	21	61	8
3	16	60	6
4	17	33	5
5	21	51	5
6	2	68	5
7	2	31	5
8	56	65	5
9	16	31	7
10	2	59	5
11	31	72	6
12	59	76	5
13	31	51	5
14	55	62	5
15	30	54	6
16	10	77	5
17	2	16	5
18	60	71	6

Refresh Save Cancel Export data 100 18 18 row(s) fetched - 4ms, on 2023-04-14 at 17:17:05

بخش ۱۴ (امتیازی)

لیست ۵ مشتری که بیشترین اختلاف میانگین قیمت سفارش خود را با میانگین قیمت کل سفارش‌ها را دارند. برای پیدا کردن این مشتریان باید میانگین قیمت کل سفارش‌ها و میانگین قیمت سفارش برای هر مشتری را محاسبه کنیم. میانگین قیمت کل سفارش‌ها را با استفاده از عبارت زیر بدست می‌آوریم:

```
select AVG(unit_price * quantity * (1 - discount)) from order_details
```

همچنین میانگین قیمت سفارش مشتری را با استفاده از دستور زیر و در نظر گرفتن group by محاسبه می‌کنیم:

```
AVG(od.unit_price * od.quantity * (1 - od.discount))
```

و در نهایت با محاسبه اختلاف (قدر مطلق (abs)) این ۲ برای هر مشتری می‌توانیم اطلاعات مورد نظر سوال را پیدا کنیم:

```
-- 14
select c.customer_id, AVG(od.unit_price * od.quantity * (1 - od.discount)) as avg_price,
       (abs(AVG(od.unit_price * od.quantity * (1 - od.discount)) - (
           select AVG(unit_price * quantity * (1 - discount))
           from order_details))) as difference
from customers c, orders o, order_details od
where c.customer_id = o.customer_id and o.order_id = od.order_id
group by c.customer_id
order by difference desc
limit 5;
```

customer_id	avg_price	difference
QUICK	1,282.2942445395	694.9192846075
CENTC	50.3999996185	536.9749603134
SIMOB	1,121.1398408137	533.7648808817
LAUGB	65.3124990463	522.0624608856
GALED	104.5874993801	482.7874605518

مشکلات و توضیحات تکمیلی

بعضی از سوالات ابهام داشت اما با پرسیدن سوالات از تی ای اکثر ابهامات برطرف شد.

آنچه آموختم / پیشنهادات

در این آزمایش با دستورات SQL به طور پیشرفته‌تری کار کردم و با دیتابیس northwind آشنا شدم.