Здесь решение лабораторной работы.

Коллеги, не удивляйтесь наличию нескольких решений в заданиях, а так же тому, что Ваше решение может отличаться от приведенных, при условии, что результаты совпадают - такова особенность реляционных БД: “Из одних и тех же таблиц одинаковые результаты можно получить разными способами)”

1. Рассчитайте совокупный доход всех магазинов на каждую дату.

Варианты решения:

1. select date(p.payment\_date), sum(amount)

from payment p

group by date(p.payment\_date)

order by date(p.payment\_date)

2. with pds as (

select cast(payment\_date as date) as payment\_date, sum(amount) as amount

from payment

group by cast(payment\_date as date)

)

select payment\_date, amount, sum(amount) over (order by payment\_date)

from pds

order by payment\_date;

2. Выведите наиболее и наименее востребованные жанры

(те, которые арендовали наибольшее/наименьшее количество раз),

число их общих продаж и сумму дохода

Варианты решения:

1.with trds as(

select c.name as category\_name,

count(\*) as cnt,

sum (p.amount) as sums

from category c

join film\_category fc

using (category\_id)

join film f

using (film\_id)

join inventory i

using (film\_id)

join rental r

using (inventory\_id)

join customer cu

using (customer\_id)

join payment p

using (rental\_id)

group by category\_name

)

select category\_name, cnt, sums

from trds

where cnt = (

select max(cnt)

from trds

)

or cnt = (

select min(cnt)

from trds

)

2. (

select 'наибольшее кол-во продаж - ' || c.name || ' в размере ' || count(p.rental\_id) || ' на сумму ' || sum(p.amount)

from payment p

inner join rental r on r.rental\_id = p.rental\_id

inner join inventory i on i.inventory\_id = r.inventory\_id

inner join film f on f.film\_id = i.film\_id

inner join film\_category fc on fc.film\_id = f.film\_id

inner join category c on c.category\_id = fc.category\_id

group by c.name

order by count(p.rental\_id) desc

limit 1

)

union all

(

select 'наименьшее кол-во продаж - ' || c.name || ' в размере ' || count(p.rental\_id) || ' на сумму ' || sum(p.amount)

from payment p

inner join rental r on r.rental\_id = p.rental\_id

inner join inventory i on i.inventory\_id = r.inventory\_id

inner join film f on f.film\_id = i.film\_id

inner join film\_category fc on fc.film\_id = f.film\_id

inner join category c on c.category\_id = fc.category\_id

group by c.name

order by count(p.rental\_id) asc

limit 1

)

3. Какова средняя арендная ставка для каждого жанра?

(упорядочить по убыванию, среднее значение округлить до сотых)

Решение:

select c.name as category\_name, round(avg(f.rental\_rate/f.rental\_duration), 2) as avr

from category c

join film\_category fc

using (category\_id)

join film f

using (film\_id)

group by c.name

order by avr desc

================= дополнительные задания =================

4. Cоставить список из 5 самых дорогих клиентов (арендовавших фильмы с 10 по 13 апреля).

формат списка:

'Имя\_клиента Фамилия\_клиента email address is: e-mail\_клиента'

Варианты решения:

1. select first\_name||' '||last\_name||'''s email address is: '||email as name\_and\_email

from customer

where customer\_id in (

select customer\_id from (

select distinct customer\_id, sum(amount)

from payment

where extract(month from payment\_date) = 4

and extract(day from payment\_date) between 10 and 13

group by customer\_id

order by sum(amount) desc

limit 5

) as top\_five

);

2. select c.first\_name || ’ ’ || c.last\_name || ’ email address is: ’ || c.email as clients\_list

from customer c

inner join payment p

on p.customer\_id = c.customer\_id

and p.payment\_date between ‘20070410’ and ‘20070413’

group by c.first\_name, c.last\_name, c.email

order by sum(p.amount) desc

limit 5

3. select

first\_name,

last\_name,

concat('email address is ', email) as email

from

payment as r

inner join customer

using (customer\_id)

where

payment\_date::date between '2007-04-10' and '2007-04-13'

group by first\_name, last\_name, email

order by sum(amount) desc

limit 5;

4. select ci.first\_name, ci.last\_name, concat('email address is:' ,ci.email) from customer as ci

inner join (

select pay.customer\_id, sum(pay.amount) from payment as pay

where pay.payment\_date between '2007-04-10' and '2007-04-13'

group by pay.customer\_id

order by sum(pay.amount) desc limit 5) as top

on top.customer\_id =ci.customer\_id;

5. select

format(’%s %s email address is: %s’, b.first\_name, b.last\_name, b.email) as “Список клиентов”

from public.customer b

inner join public.rental a ON b.customer\_id = a.customer\_id

inner JOIN public.payment c ON a.rental\_id = c.rental\_id

WHERE DATE (c.payment\_date) <= ‘2007-04-13’ AND

DATE (c.payment\_date) >= ‘2007-04-10’

GROUP BY b.first\_name, b.last\_name, b.email

order by sum(c.amount) desc

LIMIT 5;

5. Сколько арендованных фильмов было возвращено в срок,

до срока возврата и после, выведите максимальную разницу со сроком?

Варианты решения

1. with rdt as (

select inventory\_id, DATE\_PART('day', return\_date - rental\_date) as ddate

from rental

),

sttbl as (

select abs(rental\_duration - ddate) as absdif,

case

when rental\_duration > ddate then 'раньше'

when rental\_duration = ddate then 'в срок'

else 'позже'

end as status

from film f

join inventory i

using (film\_id)

join rdt

using (inventory\_id)

)

select status, count(\*) as cnt, round(max(absdif)) as maxdif

from sttbl

group by status

order by cnt, maxdif desc

2. with rental\_scheme as (

select

rental\_id,

rental\_duration as dur,

extract(day from return\_date - rental\_date) as back

from

rental as r

left join inventory as i using (inventory\_id)

left join film as f using (film\_id)

)

select

count(case when back = dur then rental\_id else null end) as return\_in\_line,

count(case when back < dur then rental\_id else null end) as return\_before\_line,

count(case when back > dur then rental\_id else null end) as return\_after\_line,

max(back - dur) as max\_late\_return

from

rental\_scheme