```
use sakila;

select count(*) from (select f.title,
f.description,
f.release_year,
f.rental_duration,
a.first_name,
a.last_name,
l.name
from film f
join film_actor fa on f.film_id = fa.film_id
join actor a on fa.actor_id = a.actor_id
join language l on f.language_id = l.language_id) as temp;

drop view big_query;

create view big_query AS( select f.title,
f.description,
f.release_year,
f.rental_duration,
a.first_name,big_querybig_query
a.last_name,
l.name
from film f
join film_actor fa big_queryon f.film_id = fa.film_id
join actor a on fa.actor_id = a.actor_id
join language l on f.language_id = l.language_id);

select count(*) from big_query;

create or replace view big_query AS (select f.title,
f.description,
f.release_year,
f.rental_duration,
a.first_name,
l.name
from film f
join film_actor fa on f.film_id = fa.film_id
join actor a on fa.actor_id = a.actor_id
join language l on f.language_id = l.language_id);

create view xyz as (select * from film);

create view abc as (select * from film_actor);
```

```
select * from xyz
join abc on xyz.film_id = abc.film_id;

select * from (select * from film)
join (select * from film_actor) on xyz.film_id = abc.film_id;


desc employee;

select * from employee;

select
   id,
   AVG(salary)
from
        employee
group by
        department;

select
        AVG(salary)
from
        employee;

-- id dept salary avg_salary_dept

select e.id, e.department, e.salary , avg_salary
from employee e
join (select department,AVG(salary) as avg_salary from employee
group by department) as temp on temp.department = e.department;



select
        id,
   department,
   salary,
   AVG(salary) over(partition by department) as avg_dept_salary
from employee;

select
        id,
   department,
```

```
    salary,
    AVG(salary) over() as avg_dept_salary
from employee;

select * from (select
            id,
            department,
    salary,
    AVG(salary) over(partition by department) as avg_dept_salary,
    AVG(salary) over() as overall_avg_salary
from employee) as temp
where salary >= avg_dept_salary;

select
            id,
            department,
    salary,
    AVG(salary) over(partition by department) as avg_dept_salary,
    AVG(salary) over() as overall_avg_salary,
    DENSE_RANK() over(partition by department order by salary desc) as salary_rank
from employee;


select
            id,
            department,
    salary,
    RANK() over(order by salary desc) as salary_rank
from employee;


update employee
set salary = 72000.00
where id = 13;

select
distinct user_id
FROM
(
select
            user_id,
            time_stamp,
TIMESTAMPDIFF(
hour,
```

```
lag(time_stamp) over (partition by user_id order by time_stamp),time_stamp
) as time_diff
from
confirmations
) as t
where
time_diff <= 24
order by
user_id;
```

```
1 x NULL
1 y
1 z
```