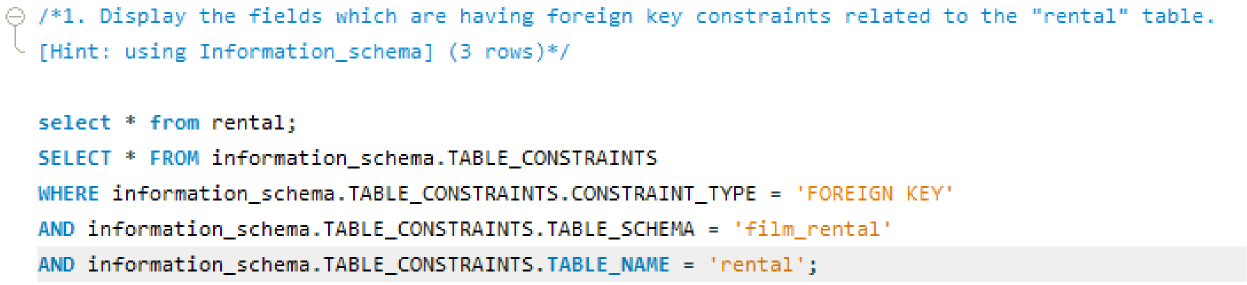
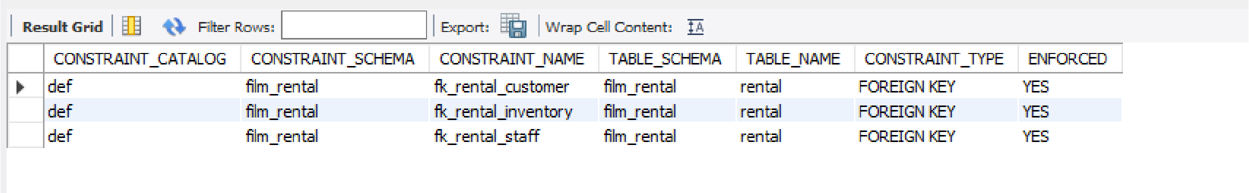
## RIDDHI SIL

## Week 3_graded project.

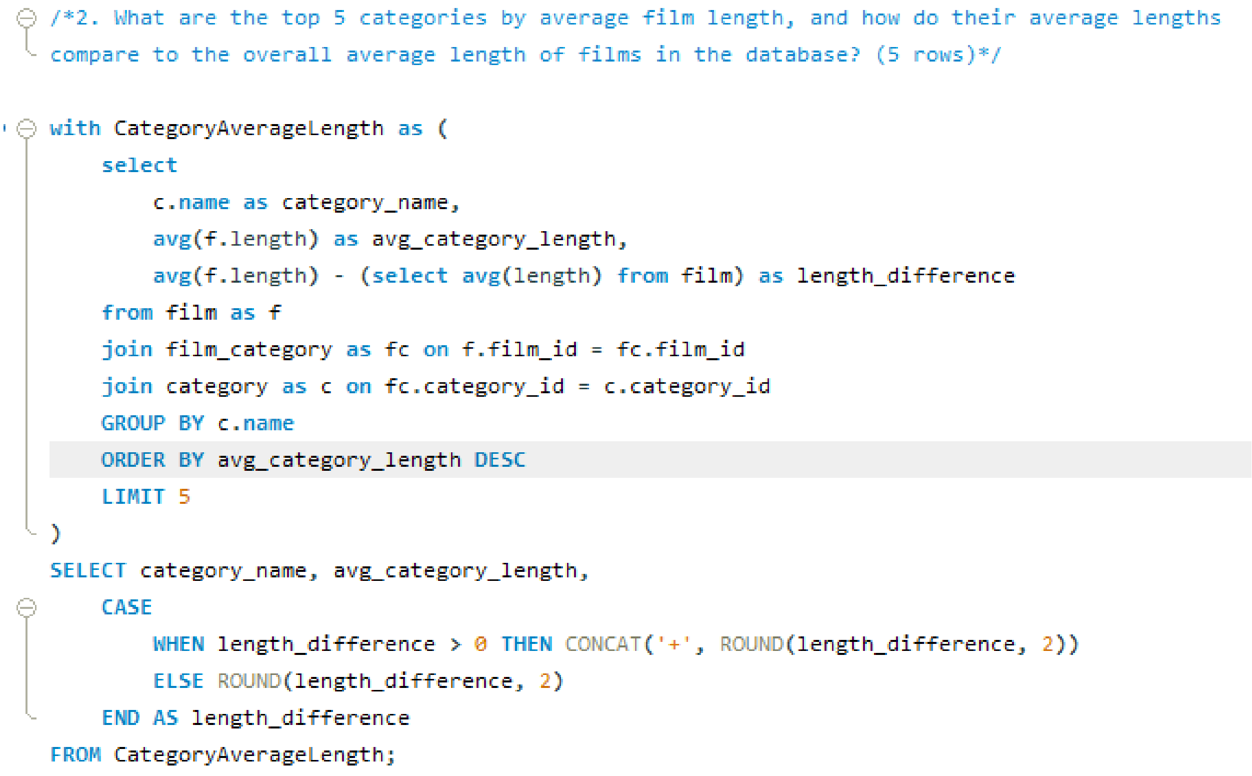


```
/*1. Display the fields which are having foreign key constraints related to the "rental" table.
[Hint: using Information_schema] (3 rows)*/

select * from rental;
SELECT * FROM information_schema.TABLE_CONSTRAINTS
WHERE information_schema.TABLE_CONSTRAINTS.CONSTRAINT_TYPE = 'FOREIGN KEY'
AND information_schema.TABLE_CONSTRAINTS.TABLE_SCHEMA = 'film_rental'
AND information_schema.TABLE_CONSTRAINTS.TABLE_NAME = 'rental';
```

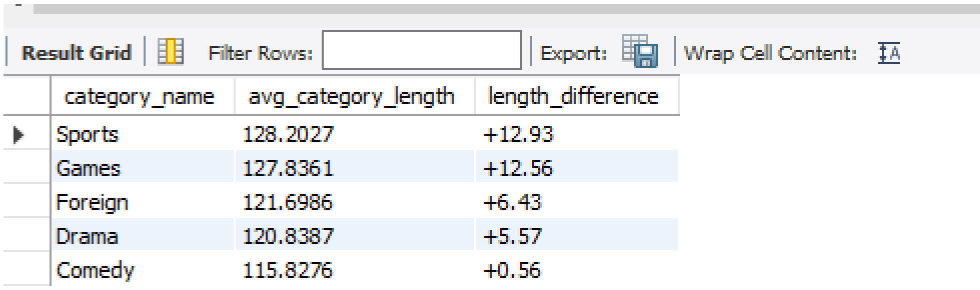


| CONSTRAINT_CATALOG | CONSTRAINT_SCHEMA | CONSTRAINT_NAME | TABLE_SCHEMA | TABLE_NAME | CONSTRAINT_TYPE | ENFORCED |
|---|---|---|---|---|---|---|
| def | film_rental | fk_rental_customer | film_rental | rental | FOREIGN KEY | YES |
| def | film_rental | fk_rental_inventory | film_rental | rental | FOREIGN KEY | YES |
| def | film_rental | fk_rental_staff | film_rental | rental | FOREIGN KEY | YES |

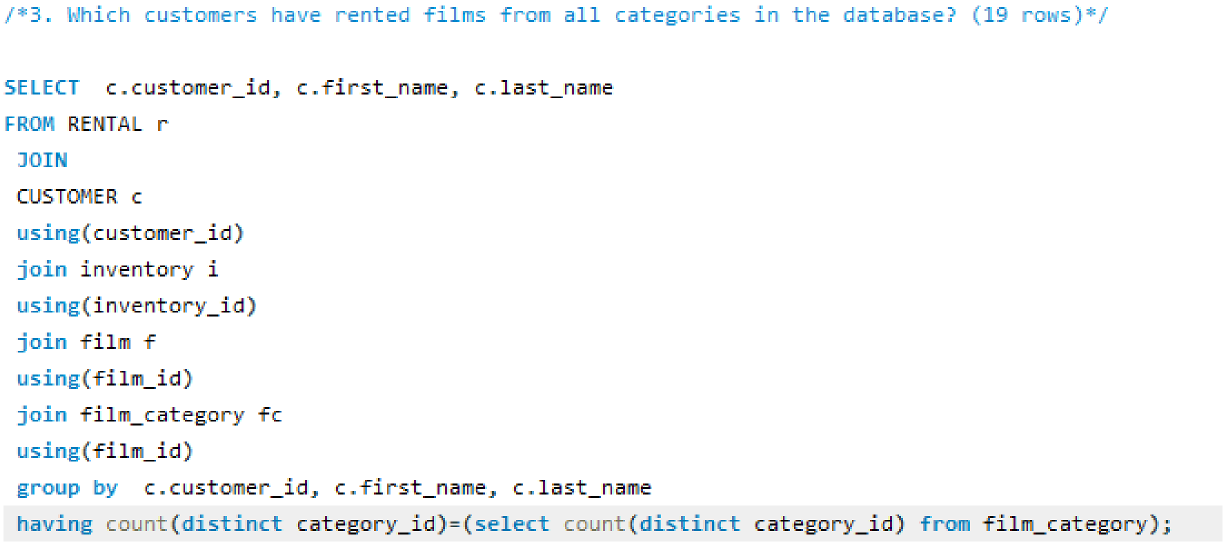
---



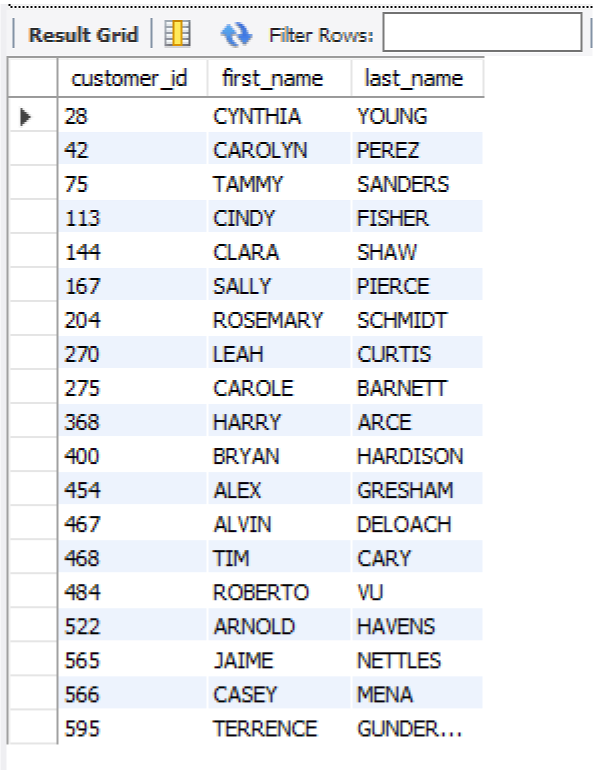
```
/*2. What are the top 5 categories by average film length, and how do their average lengths
compare to the overall average length of films in the database? (5 rows)*/

with CategoryAverageLength as (
    select
        c.name as category_name,
        avg(f.length) as avg_category_length,
        avg(f.length) - (select avg(length) from film) as length_difference
    from film as f
    join film_category as fc on f.film_id = fc.film_id
    join category as c on fc.category_id = c.category_id
    GROUP BY c.name
    ORDER BY avg_category_length DESC
    LIMIT 5
)
SELECT category_name, avg_category_length,
    CASE
        WHEN length_difference > 0 THEN CONCAT('+', ROUND(length_difference, 2))
        ELSE ROUND(length_difference, 2)
    END AS length_difference
FROM CategoryAverageLength;
```



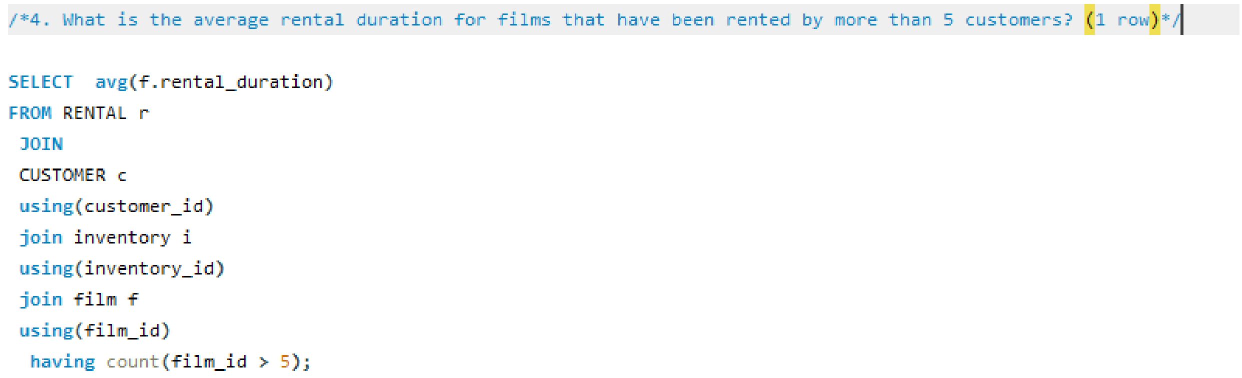
| category_name | avg_category_length | length_difference |
|---|---|---|
| Sports | 128.2027 | +12.93 |
| Games | 127.8361 | +12.56 |
| Foreign | 121.6986 | +6.43 |
| Drama | 120.8387 | +5.57 |
| Comedy | 115.8276 | +0.56 |



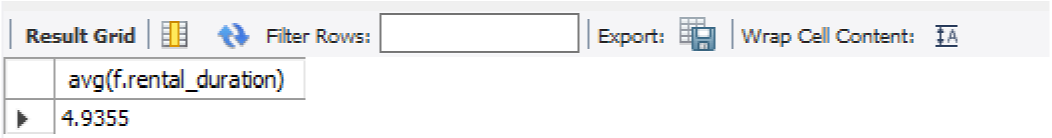
```
/*3. Which customers have rented films from all categories in the database? (19 rows)*/

SELECT  c.customer_id, c.first_name, c.last_name
FROM RENTAL r
 JOIN
 CUSTOMER c
 using(customer_id)
 join inventory i
 using(inventory_id)
 join film f
 using(film_id)
 join film_category fc
 using(film_id)
 group by  c.customer_id, c.first_name, c.last_name
 having count(distinct category_id)=(select count(distinct category_id) from film_category);
```



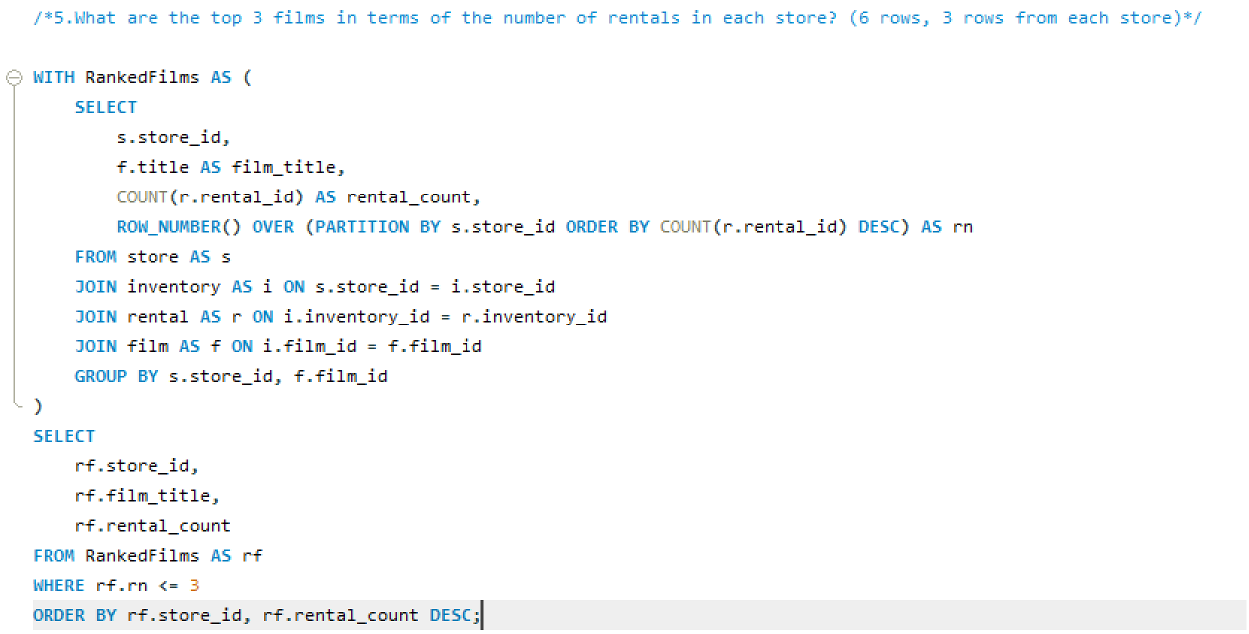
| customer_id | first_name | last_name |
|---|---|---|
| 28 | CYNTHIA | YOUNG |
| 42 | CAROLYN | PEREZ |
| 75 | TAMMY | SANDERS |
| 113 | CINDY | FISHER |
| 144 | CLARA | SHAW |
| 167 | SALLY | PIERCE |
| 204 | ROSEMARY | SCHMIDT |
| 270 | LEAH | CURTIS |
| 275 | CAROLE | BARNETT |
| 368 | HARRY | ARCE |
| 400 | BRYAN | HARDISON |
| 454 | ALEX | GRESHAM |
| 467 | ALVIN | DELOACH |
| 468 | TIM | CARY |
| 484 | ROBERTO | VU |
| 522 | ARNOLD | HAVENS |
| 565 | JAIME | NETTLES |
| 566 | CASEY | MENA |
| 595 | TERRENCE | GUNDER... |



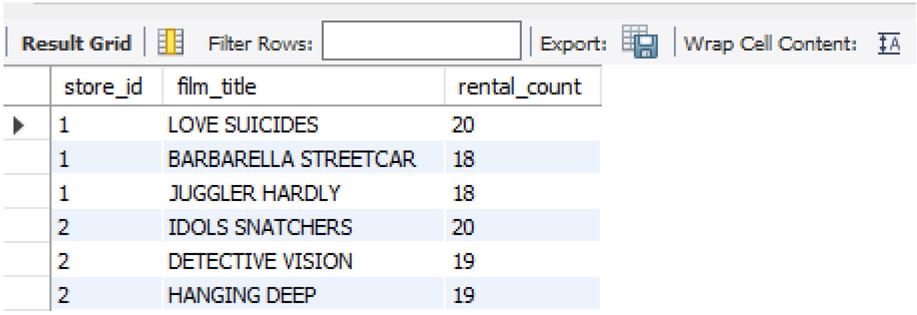
```
/*4. What is the average rental duration for films that have been rented by more than 5 customers? (1 row)*/

SELECT  avg(f.rental_duration)
FROM RENTAL r
 JOIN
 CUSTOMER c
 using(customer_id)
 join inventory i
 using(inventory_id)
 join film f
 using(film_id)
  having count(film_id > 5);
```



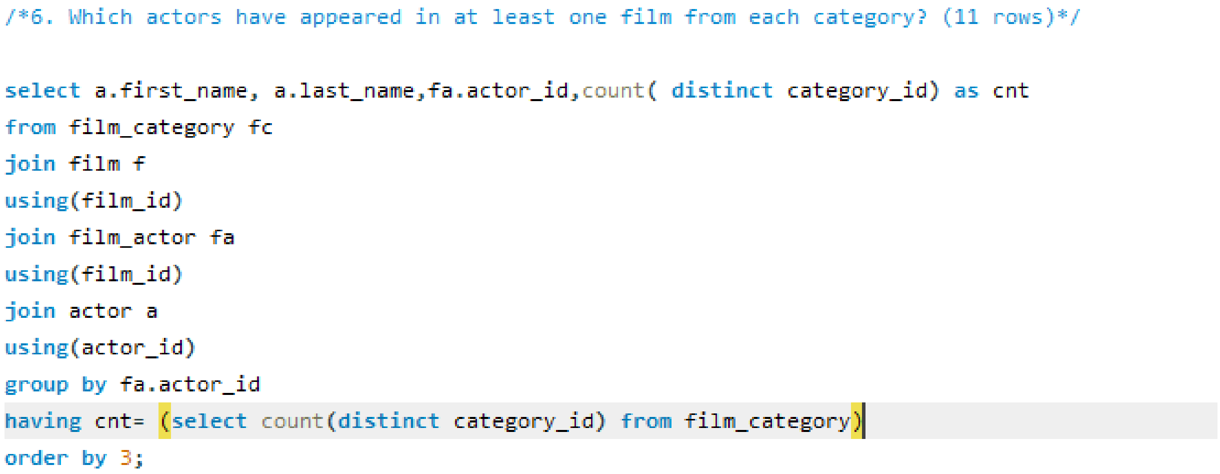
| avg(f.rental_duration) |
|---|
| 4.9355 |



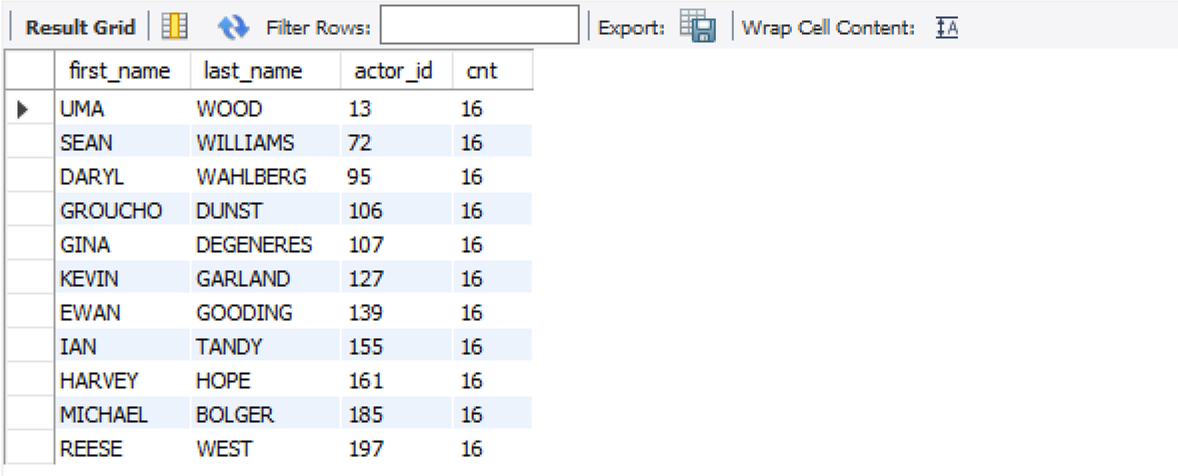
```sql
/*5.What are the top 3 films in terms of the number of rentals in each store? (6 rows, 3 rows from each store)*/

WITH RankedFilms AS (
    SELECT
        s.store_id,
        f.title AS film_title,
        COUNT(r.rental_id) AS rental_count,
        ROW_NUMBER() OVER (PARTITION BY s.store_id ORDER BY COUNT(r.rental_id) DESC) AS rn
    FROM store AS s
    JOIN inventory AS i ON s.store_id = i.store_id
    JOIN rental AS r ON i.inventory_id = r.inventory_id
    JOIN film AS f ON i.film_id = f.film_id
    GROUP BY s.store_id, f.film_id
)
SELECT
    rf.store_id,
    rf.film_title,
    rf.rental_count
FROM RankedFilms AS rf
WHERE rf.rn <= 3
ORDER BY rf.store_id, rf.rental_count DESC;
```



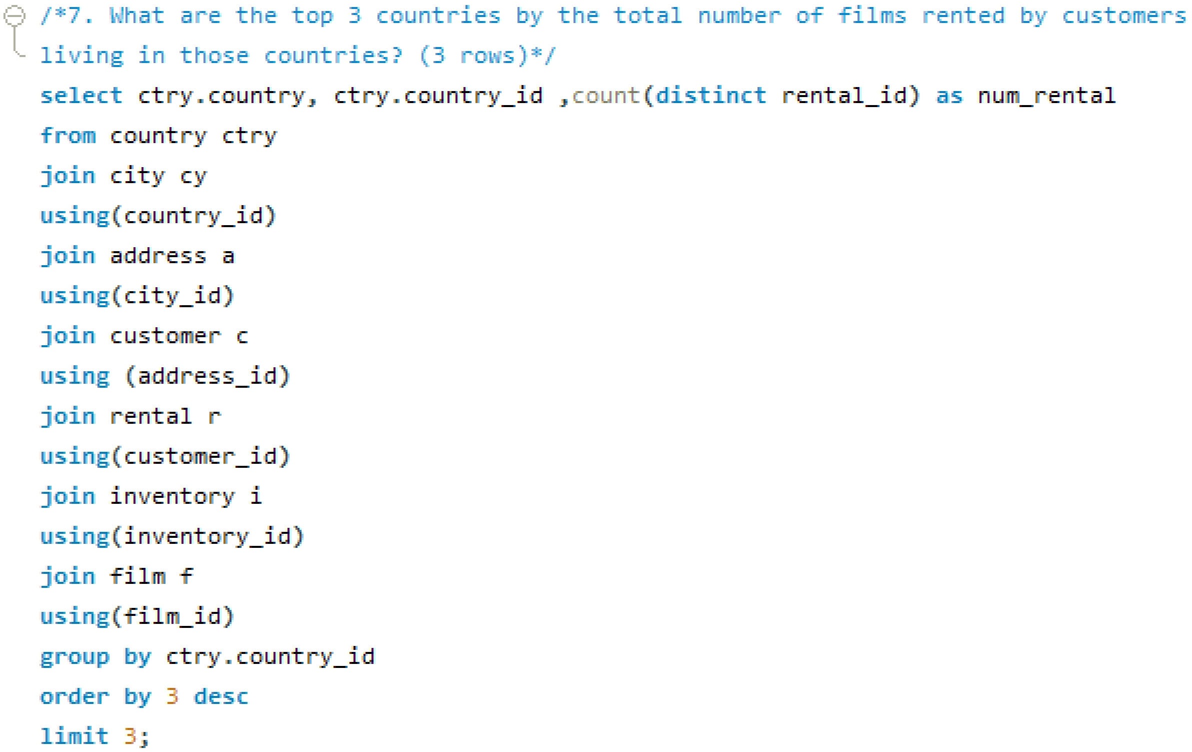
| store_id | film_title | rental_count |
|---|---|---|
| 1 | LOVE SUICIDES | 20 |
| 1 | BARBARELLA STREETCAR | 18 |
| 1 | JUGGLER HARDLY | 18 |
| 2 | IDOLS SNATCHERS | 20 |
| 2 | DETECTIVE VISION | 19 |
| 2 | HANGING DEEP | 19 |



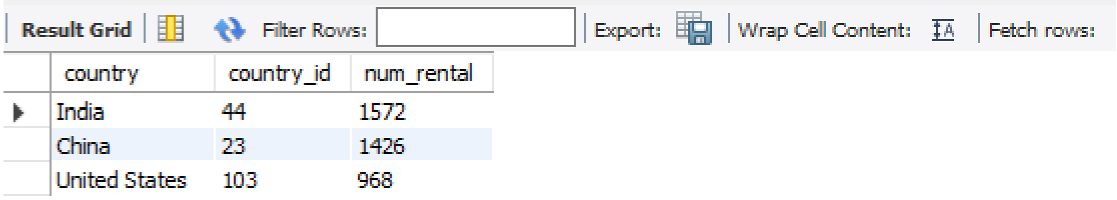
```sql
/*6. Which actors have appeared in at least one film from each category? (11 rows)*/

select a.first_name, a.last_name,fa.actor_id,count( distinct category_id) as cnt
from film_category fc
join film f
using(film_id)
join film_actor fa
using(film_id)
join actor a
using(actor_id)
group by fa.actor_id
having cnt= (select count(distinct category_id) from film_category)
order by 3;
```



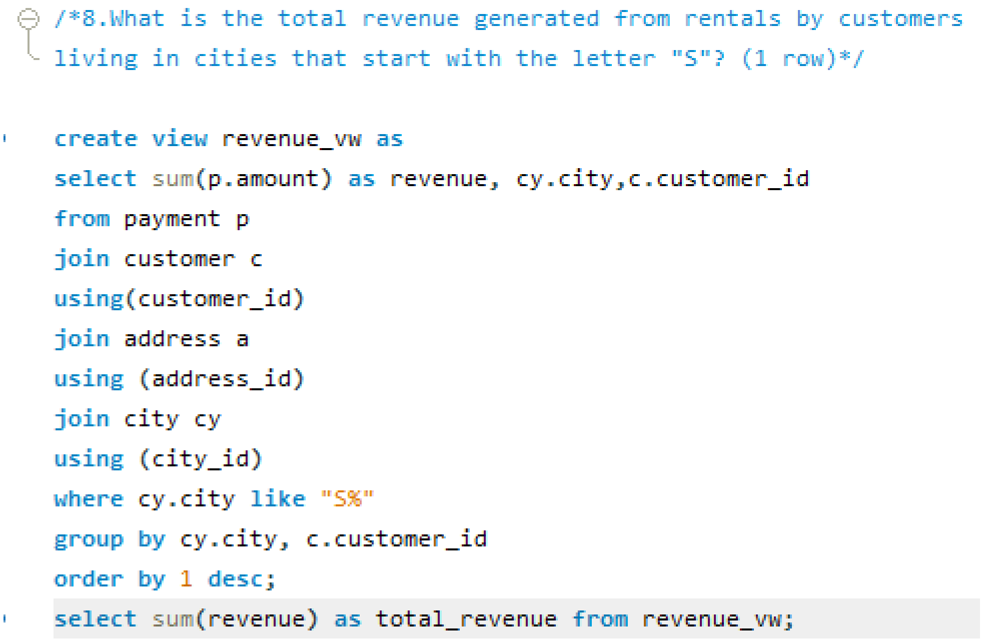
| first_name | last_name | actor_id | cnt |
|---|---|---|---|
| UMA | WOOD | 13 | 16 |
| SEAN | WILLIAMS | 72 | 16 |
| DARYL | WAHLBERG | 95 | 16 |
| GROUCHO | DUNST | 106 | 16 |
| GINA | DEGENERES | 107 | 16 |
| KEVIN | GARLAND | 127 | 16 |
| EWAN | GOODING | 139 | 16 |
| IAN | TANDY | 155 | 16 |
| HARVEY | HOPE | 161 | 16 |
| MICHAEL | BOLGER | 185 | 16 |
| REESE | WEST | 197 | 16 |



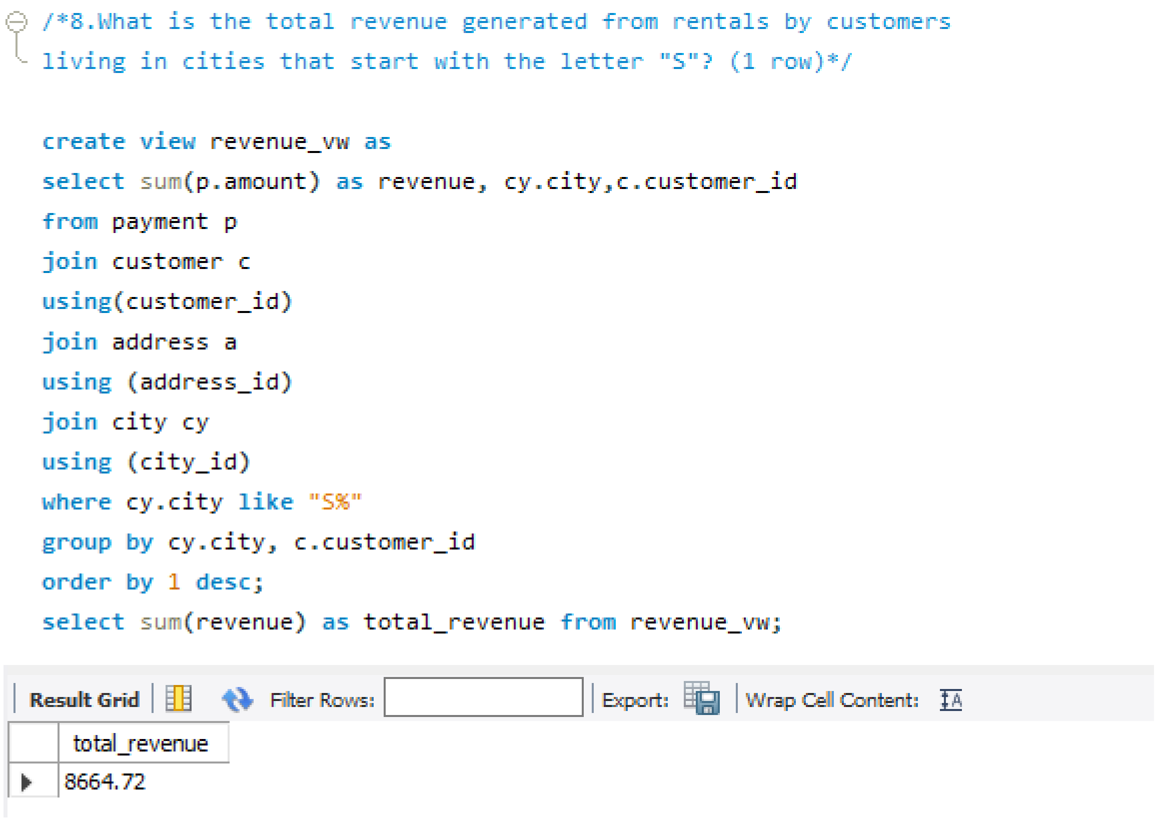
```
/*7. What are the top 3 countries by the total number of films rented by customers
living in those countries? (3 rows)*/
select ctry.country, ctry.country_id ,count(distinct rental_id) as num_rental
from country ctry
join city cy
using(country_id)
join address a
using(city_id)
join customer c
using (address_id)
join rental r
using(customer_id)
join inventory i
using(inventory_id)
join film f
using(film_id)
group by ctry.country_id
order by 3 desc
limit 3;
```



| country | country_id | num_rental |
|---|---|---|
| India | 44 | 1572 |
| China | 23 | 1426 |
| United States | 103 | 968 |

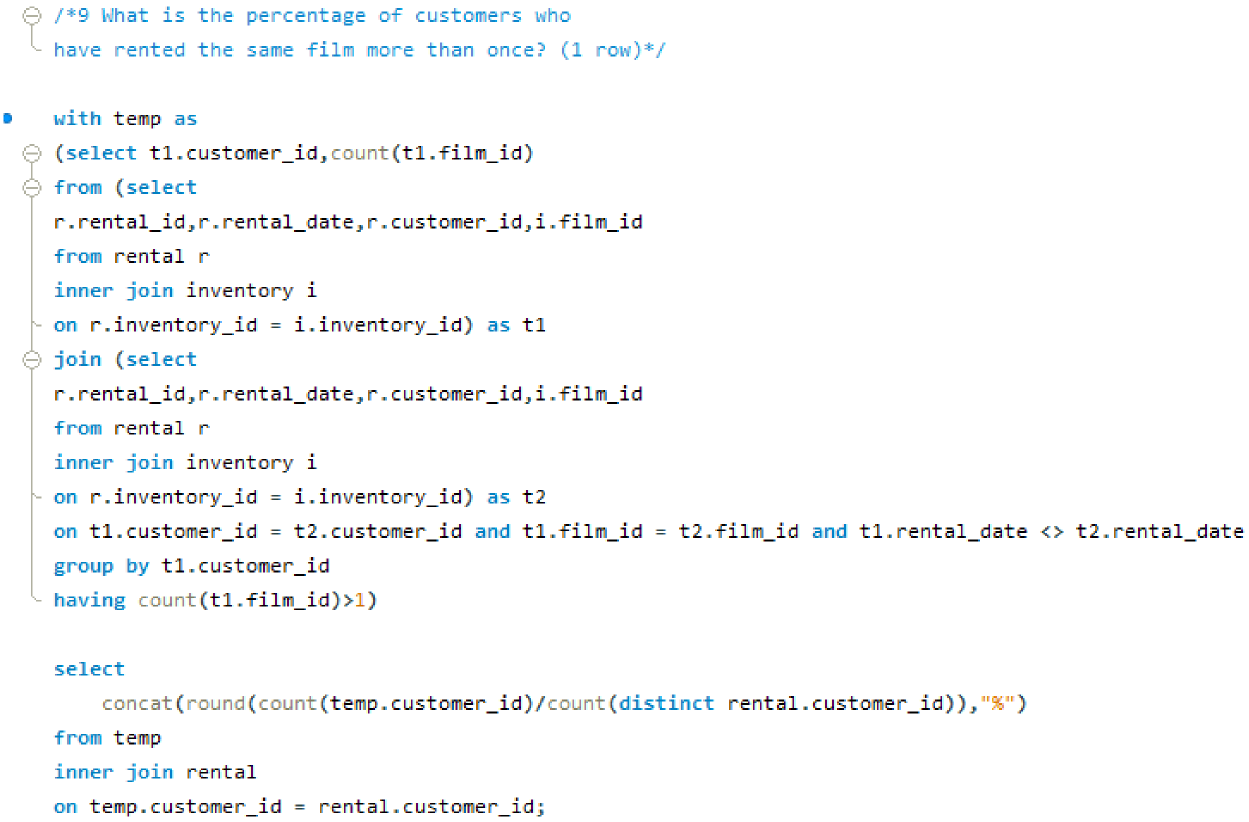


```
/*8.What is the total revenue generated from rentals by customers
living in cities that start with the letter "S"? (1 row)*/

create view revenue_vw as
select sum(p.amount) as revenue, cy.city,c.customer_id
from payment p
join customer c
using(customer_id)
join address a
using (address_id)
join city cy
using (city_id)
where cy.city like "S%"
group by cy.city, c.customer_id
order by 1 desc;
select sum(revenue) as total_revenue from revenue_vw;
```

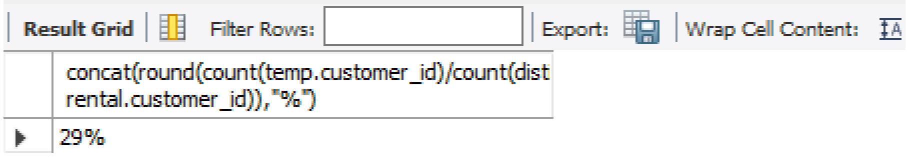


```
/*8.What is the total revenue generated from rentals by customers
living in cities that start with the letter "S"? (1 row)*/

create view revenue_vw as
select sum(p.amount) as revenue, cy.city,c.customer_id
from payment p
join customer c
using(customer_id)
join address a
using (address_id)
join city cy
using (city_id)
where cy.city like "S%"
group by cy.city, c.customer_id
order by 1 desc;
select sum(revenue) as total_revenue from revenue_vw;
```

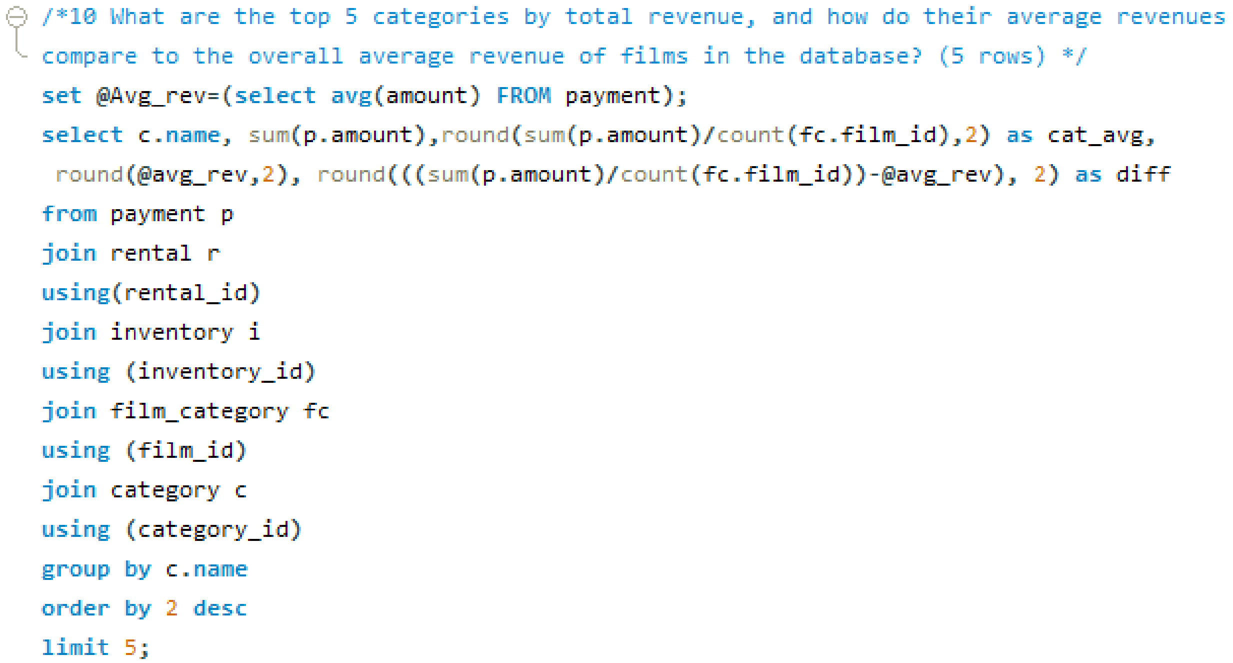
```
total_revenue
8664.72
```



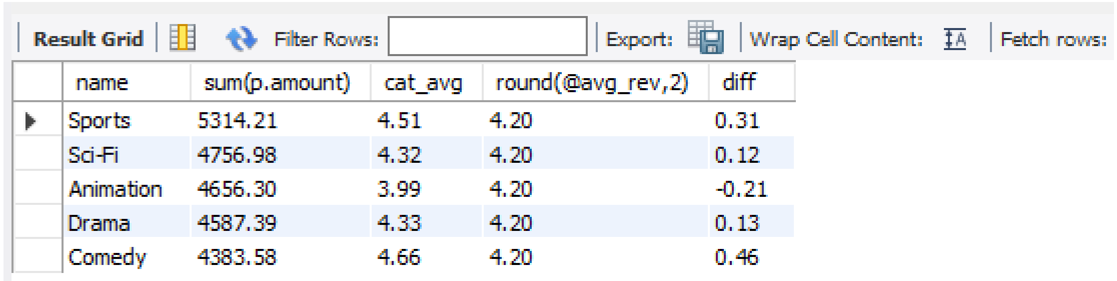
```
/*9 What is the percentage of customers who
have rented the same film more than once? (1 row)*/

with temp as
(select t1.customer_id,count(t1.film_id)
from (select
r.rental_id,r.rental_date,r.customer_id,i.film_id
from rental r
inner join inventory i
on r.inventory_id = i.inventory_id) as t1
join (select
r.rental_id,r.rental_date,r.customer_id,i.film_id
from rental r
inner join inventory i
on r.inventory_id = i.inventory_id) as t2
on t1.customer_id = t2.customer_id and t1.film_id = t2.film_id and t1.rental_date <> t2.rental_date
group by t1.customer_id
having count(t1.film_id)>1)

select
    concat(round(count(temp.customer_id)/count(distinct rental.customer_id)),"%")
from temp
inner join rental
on temp.customer_id = rental.customer_id;
```



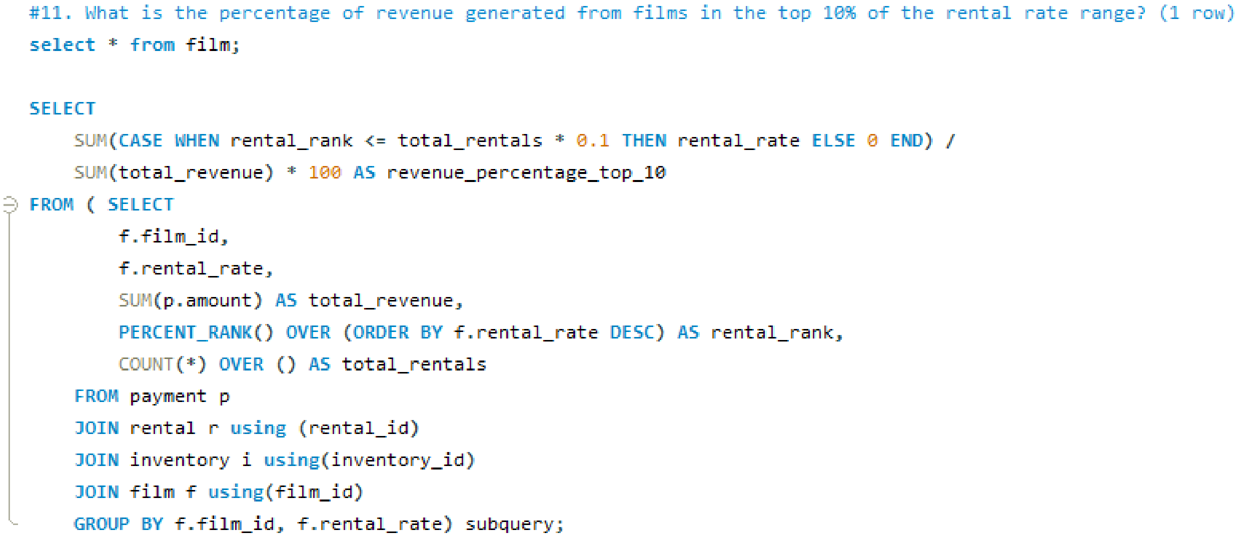
```
concat(round(count(temp.customer_id)/count(dist
rental.customer_id)),"%")
29%
```



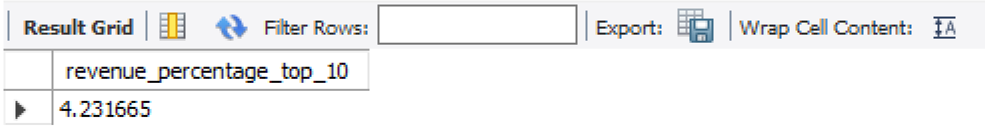
```
/*10 What are the top 5 categories by total revenue, and how do their average revenues
compare to the overall average revenue of films in the database? (5 rows) */
set @Avg_rev=(select avg(amount) FROM payment);
select c.name, sum(p.amount),round(sum(p.amount)/count(fc.film_id),2) as cat_avg,
 round(@avg_rev,2), round(((sum(p.amount)/count(fc.film_id))-@avg_rev), 2) as diff
from payment p
join rental r
using(rental_id)
join inventory i
using (inventory_id)
join film_category fc
using (film_id)
join category c
using (category_id)
group by c.name
order by 2 desc
limit 5;
```



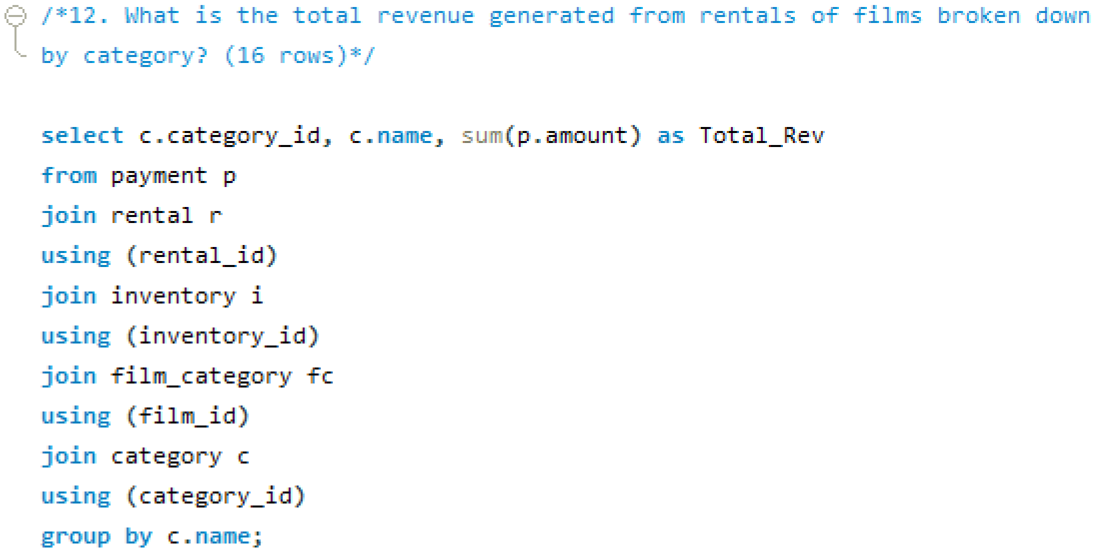
| name | sum(p.amount) | cat_avg | round(@avg_rev,2) | diff |
|---|---|---|---|---|
| Sports | 5314.21 | 4.51 | 4.20 | 0.31 |
| Sci-Fi | 4756.98 | 4.32 | 4.20 | 0.12 |
| Animation | 4656.30 | 3.99 | 4.20 | -0.21 |
| Drama | 4587.39 | 4.33 | 4.20 | 0.13 |
| Comedy | 4383.58 | 4.66 | 4.20 | 0.46 |



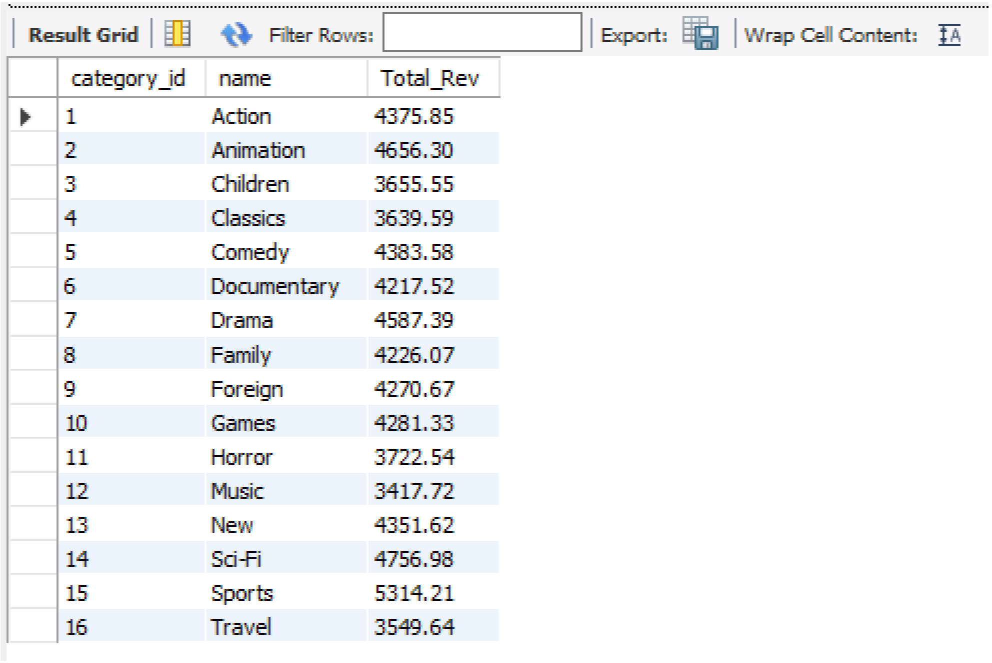
```
#11. What is the percentage of revenue generated from films in the top 10% of the rental rate range? (1 row)
select * from film;

SELECT
    SUM(CASE WHEN rental_rank <= total_rentals * 0.1 THEN rental_rate ELSE 0 END) /
    SUM(total_revenue) * 100 AS revenue_percentage_top_10
FROM ( SELECT
        f.film_id,
        f.rental_rate,
        SUM(p.amount) AS total_revenue,
        PERCENT_RANK() OVER (ORDER BY f.rental_rate DESC) AS rental_rank,
        COUNT(*) OVER () AS total_rentals
    FROM payment p
    JOIN rental r using (rental_id)
    JOIN inventory i using(inventory_id)
    JOIN film f using(film_id)
    GROUP BY f.film_id, f.rental_rate) subquery;
```



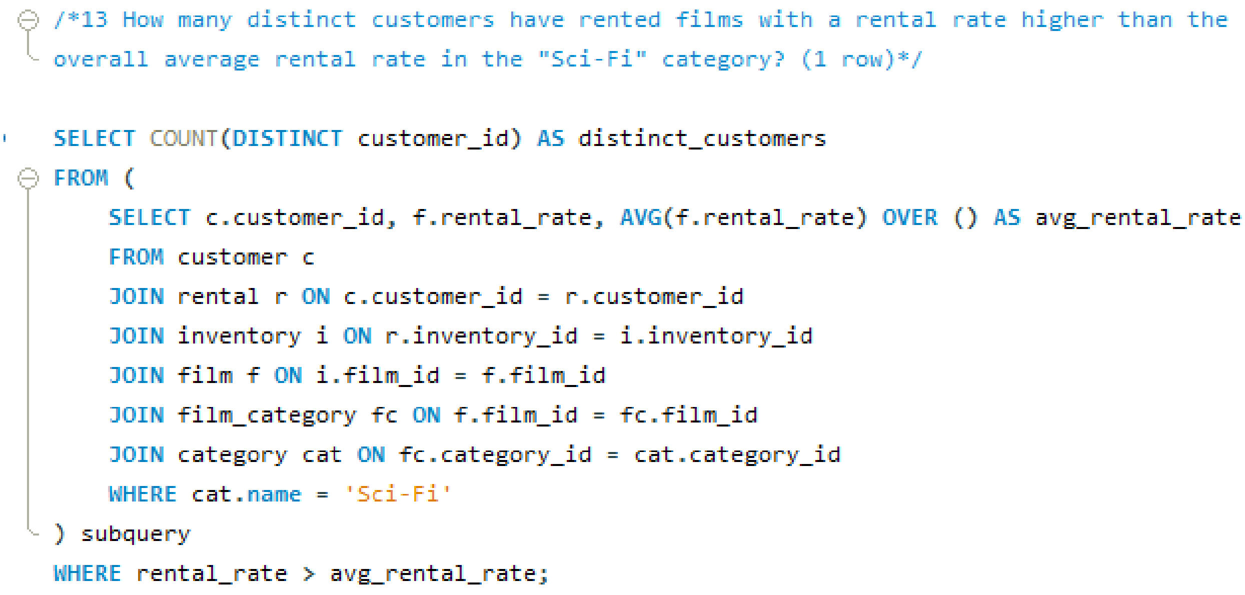
| revenue_percentage_top_10 |
|---|
| 4.231665 |



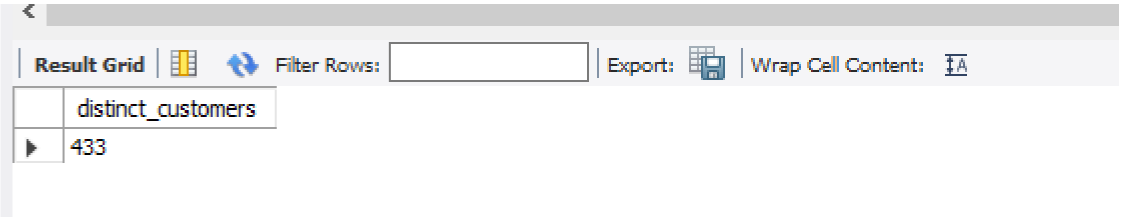
```
/*12. What is the total revenue generated from rentals of films broken down
by category? (16 rows)*/

select c.category_id, c.name, sum(p.amount) as Total_Rev
from payment p
join rental r
using (rental_id)
join inventory i
using (inventory_id)
join film_category fc
using (film_id)
join category c
using (category_id)
group by c.name;
```



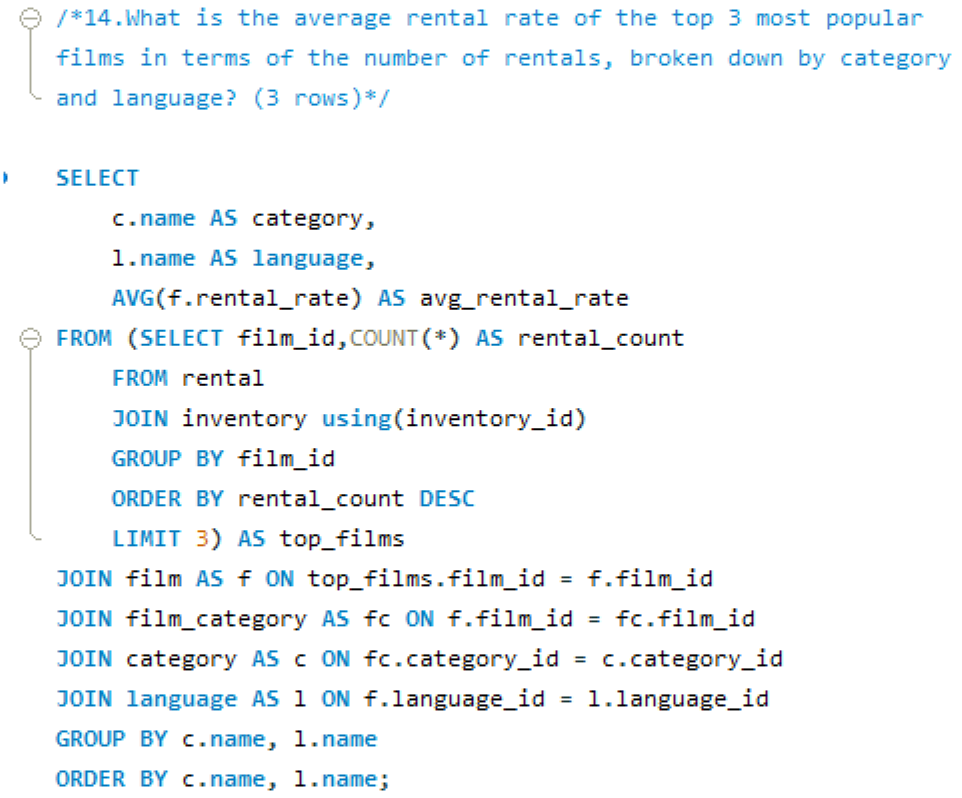
| category_id | name | Total_Rev |
|---|---|---|
| 1 | Action | 4375.85 |
| 2 | Animation | 4656.30 |
| 3 | Children | 3655.55 |
| 4 | Classics | 3639.59 |
| 5 | Comedy | 4383.58 |
| 6 | Documentary | 4217.52 |
| 7 | Drama | 4587.39 |
| 8 | Family | 4226.07 |
| 9 | Foreign | 4270.67 |
| 10 | Games | 4281.33 |
| 11 | Horror | 3722.54 |
| 12 | Music | 3417.72 |
| 13 | New | 4351.62 |
| 14 | Sci-Fi | 4756.98 |
| 15 | Sports | 5314.21 |
| 16 | Travel | 3549.64 |



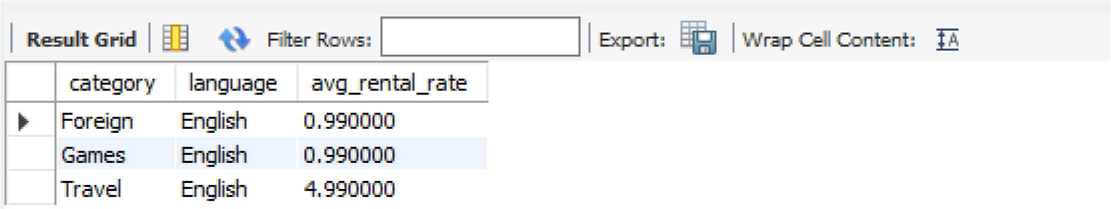
```
/*13 How many distinct customers have rented films with a rental rate higher than the
overall average rental rate in the "Sci-Fi" category? (1 row)*/

SELECT COUNT(DISTINCT customer_id) AS distinct_customers
FROM (
    SELECT c.customer_id, f.rental_rate, AVG(f.rental_rate) OVER () AS avg_rental_rate
    FROM customer c
    JOIN rental r ON c.customer_id = r.customer_id
    JOIN inventory i ON r.inventory_id = i.inventory_id
    JOIN film f ON i.film_id = f.film_id
    JOIN film_category fc ON f.film_id = fc.film_id
    JOIN category cat ON fc.category_id = cat.category_id
    WHERE cat.name = 'Sci-Fi'
) subquery
WHERE rental_rate > avg_rental_rate;
```



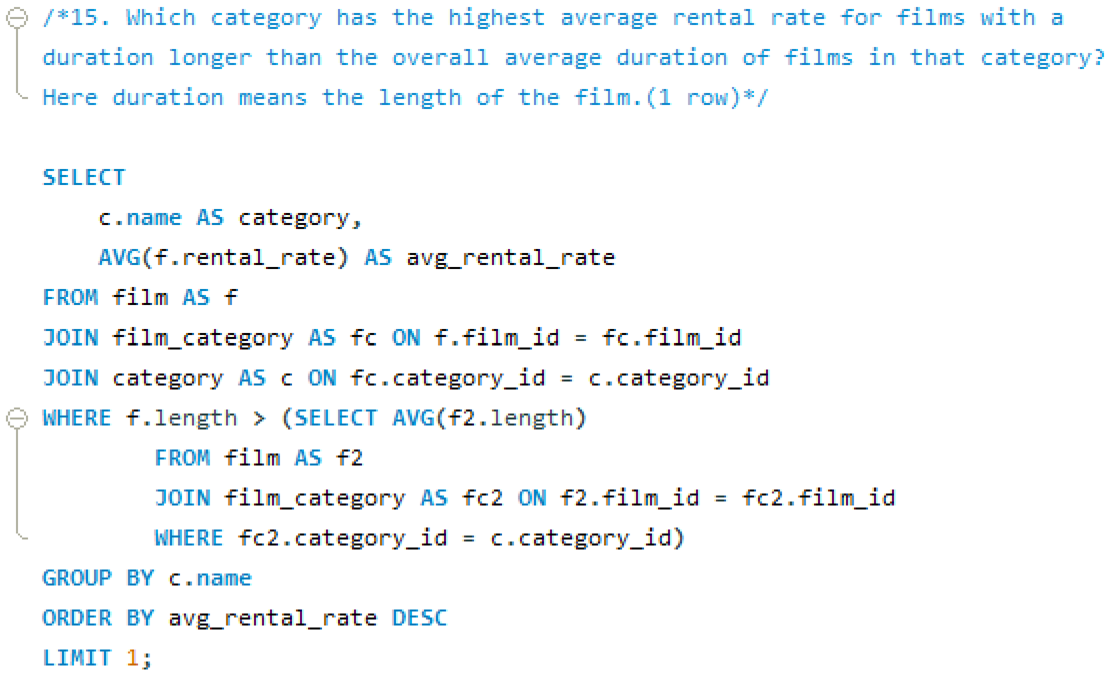
```
distinct_customers
433
```



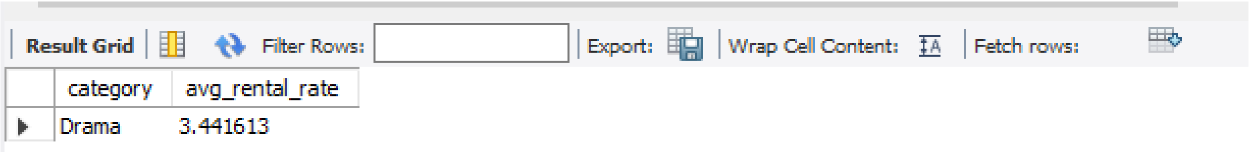
```
/*14.What is the average rental rate of the top 3 most popular
films in terms of the number of rentals, broken down by category
and language? (3 rows)*/

SELECT
    c.name AS category,
    l.name AS language,
    AVG(f.rental_rate) AS avg_rental_rate
FROM (SELECT film_id,COUNT(*) AS rental_count
    FROM rental
    JOIN inventory using(inventory_id)
    GROUP BY film_id
    ORDER BY rental_count DESC
    LIMIT 3) AS top_films
JOIN film AS f ON top_films.film_id = f.film_id
JOIN film_category AS fc ON f.film_id = fc.film_id
JOIN category AS c ON fc.category_id = c.category_id
JOIN language AS l ON f.language_id = l.language_id
GROUP BY c.name, l.name
ORDER BY c.name, l.name;
```



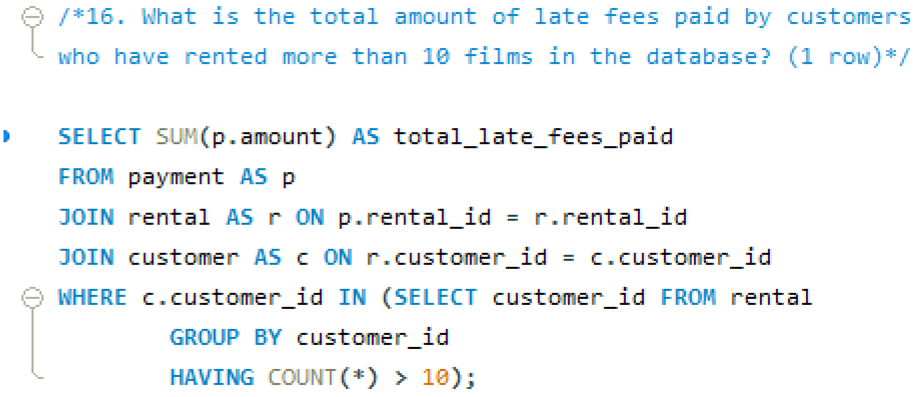
| category | language | avg_rental_rate |
|---|---|---|
| Foreign | English | 0.990000 |
| Games | English | 0.990000 |
| Travel | English | 4.990000 |



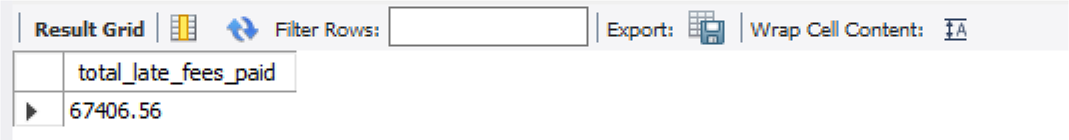
```
/*15. Which category has the highest average rental rate for films with a
duration longer than the overall average duration of films in that category?
Here duration means the length of the film.(1 row)*/

SELECT
    c.name AS category,
    AVG(f.rental_rate) AS avg_rental_rate
FROM film AS f
JOIN film_category AS fc ON f.film_id = fc.film_id
JOIN category AS c ON fc.category_id = c.category_id
WHERE f.length > (SELECT AVG(f2.length)
        FROM film AS f2
        JOIN film_category AS fc2 ON f2.film_id = fc2.film_id
        WHERE fc2.category_id = c.category_id)
GROUP BY c.name
ORDER BY avg_rental_rate DESC
LIMIT 1;
```



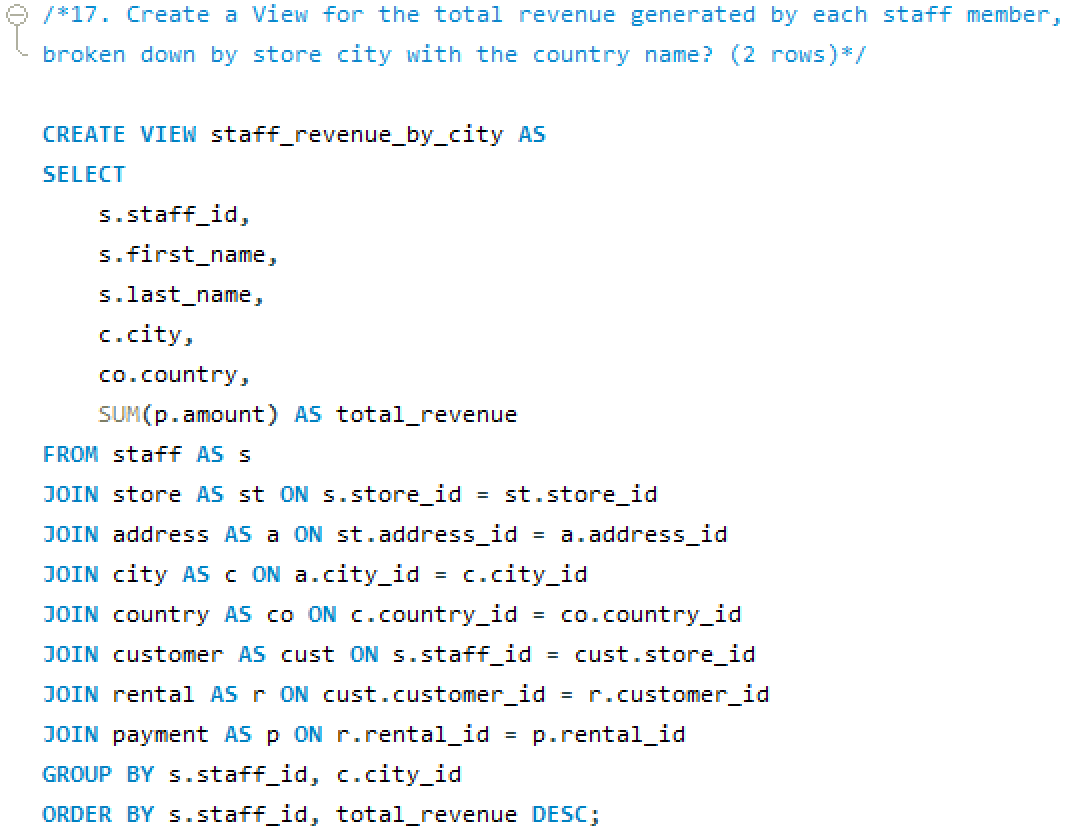
| category | avg_rental_rate |
|---|---|
| Drama | 3.441613 |



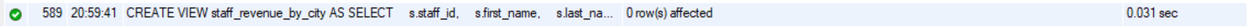
```
/*16. What is the total amount of late fees paid by customers
who have rented more than 10 films in the database? (1 row)*/

SELECT SUM(p.amount) AS total_late_fees_paid
FROM payment AS p
JOIN rental AS r ON p.rental_id = r.rental_id
JOIN customer AS c ON r.customer_id = c.customer_id
WHERE c.customer_id IN (SELECT customer_id FROM rental
        GROUP BY customer_id
        HAVING COUNT(*) > 10);
```



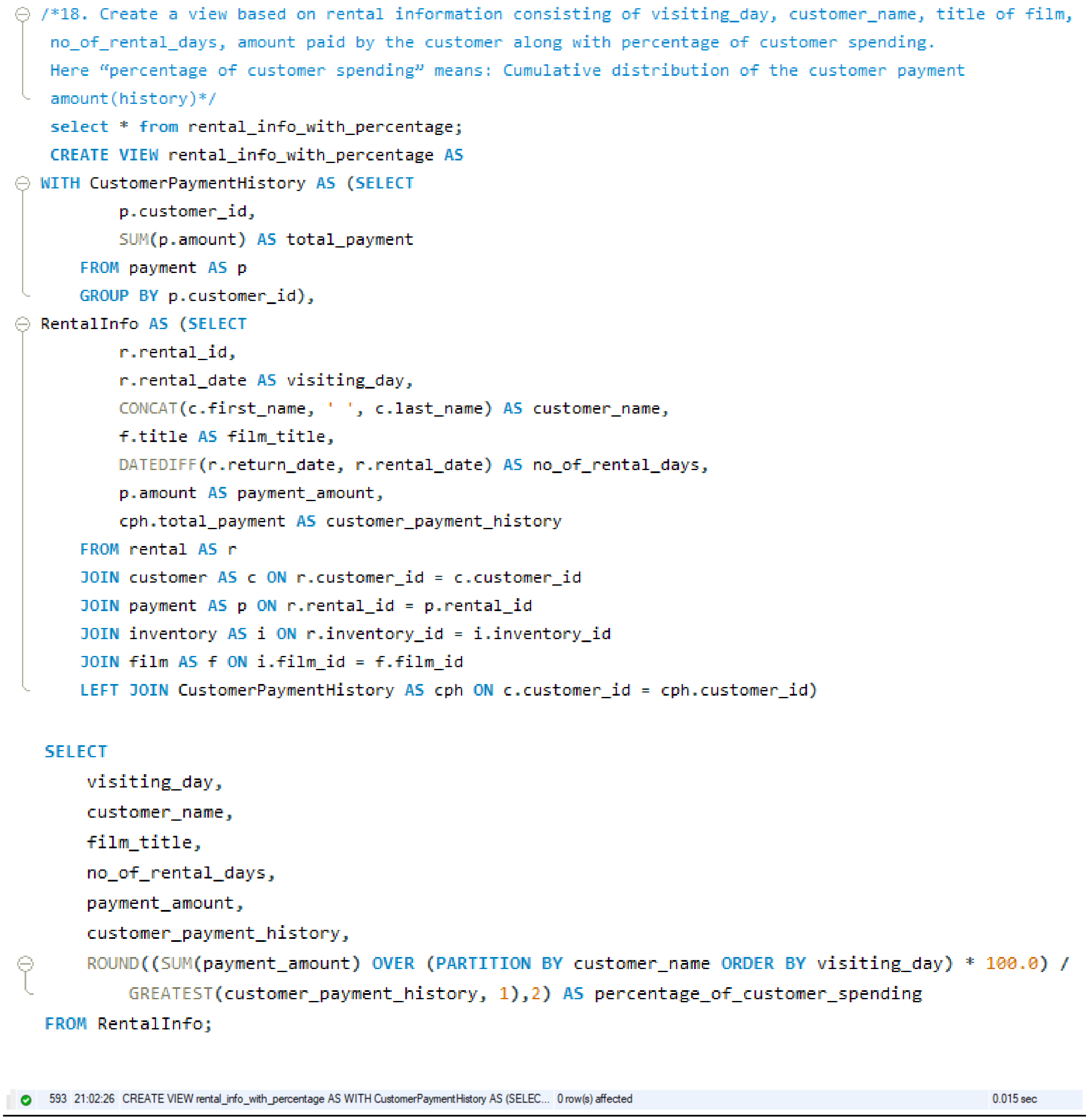
| total_late_fees_paid |
|---|
| 67406.56 |



```sql
/*17. Create a View for the total revenue generated by each staff member,
broken down by store city with the country name? (2 rows)*/

CREATE VIEW staff_revenue_by_city AS
SELECT
    s.staff_id,
    s.first_name,
    s.last_name,
    c.city,
    co.country,
    SUM(p.amount) AS total_revenue
FROM staff AS s
JOIN store AS st ON s.store_id = st.store_id
JOIN address AS a ON st.address_id = a.address_id
JOIN city AS c ON a.city_id = c.city_id
JOIN country AS co ON c.country_id = co.country_id
JOIN customer AS cust ON s.staff_id = cust.store_id
JOIN rental AS r ON cust.customer_id = r.customer_id
JOIN payment AS p ON r.rental_id = p.rental_id
GROUP BY s.staff_id, c.city_id
ORDER BY s.staff_id, total_revenue DESC;
```

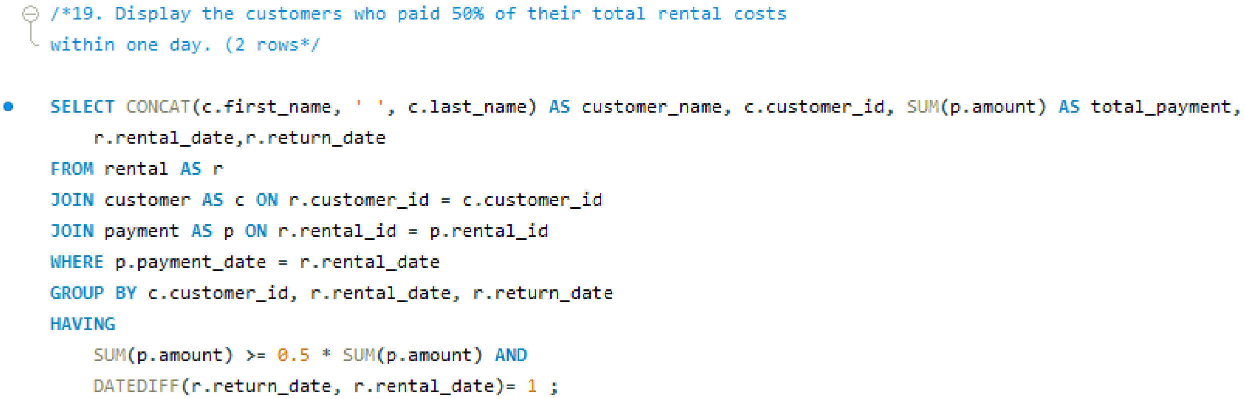


```
589  20:59:41  CREATE VIEW staff_revenue_by_city AS SELECT    s.staff_id,    s.first_name,    s.last_na...  0 row(s) affected                    0.031 sec
```



```
/*18. Create a view based on rental information consisting of visiting_day, customer_name, title of film,
 no_of_rental_days, amount paid by the customer along with percentage of customer spending.
 Here “percentage of customer spending” means: Cumulative distribution of the customer payment
 amount(history)*/
 select * from rental_info_with_percentage;
 CREATE VIEW rental_info_with_percentage AS
WITH CustomerPaymentHistory AS (SELECT
        p.customer_id,
        SUM(p.amount) AS total_payment
    FROM payment AS p
    GROUP BY p.customer_id),
RentalInfo AS (SELECT
        r.rental_id,
        r.rental_date AS visiting_day,
        CONCAT(c.first_name, ' ', c.last_name) AS customer_name,
        f.title AS film_title,
        DATEDIFF(r.return_date, r.rental_date) AS no_of_rental_days,
        p.amount AS payment_amount,
        cph.total_payment AS customer_payment_history
    FROM rental AS r
    JOIN customer AS c ON r.customer_id = c.customer_id
    JOIN payment AS p ON r.rental_id = p.rental_id
    JOIN inventory AS i ON r.inventory_id = i.inventory_id
    JOIN film AS f ON i.film_id = f.film_id
    LEFT JOIN CustomerPaymentHistory AS cph ON c.customer_id = cph.customer_id)

SELECT
    visiting_day,
    customer_name,
    film_title,
    no_of_rental_days,
    payment_amount,
    customer_payment_history,
    ROUND((SUM(payment_amount) OVER (PARTITION BY customer_name ORDER BY visiting_day) * 100.0) /
        GREATEST(customer_payment_history, 1),2) AS percentage_of_customer_spending
FROM RentalInfo;
```

```
593  21:02:26  CREATE VIEW rental_info_with_percentage AS WITH CustomerPaymentHistory AS (SELEC...  0 row(s) affected    0.015 sec
```



```
/*19. Display the customers who paid 50% of their total rental costs
within one day. (2 rows*/

SELECT CONCAT(c.first_name, ' ', c.last_name) AS customer_name, c.customer_id, SUM(p.amount) AS total_payment,
    r.rental_date,r.return_date
FROM rental AS r
JOIN customer AS c ON r.customer_id = c.customer_id
JOIN payment AS p ON r.rental_id = p.rental_id
WHERE p.payment_date = r.rental_date
GROUP BY c.customer_id, r.rental_date, r.return_date
HAVING
    SUM(p.amount) >= 0.5 * SUM(p.amount) AND
    DATEDIFF(r.return_date, r.rental_date)= 1 ;
```