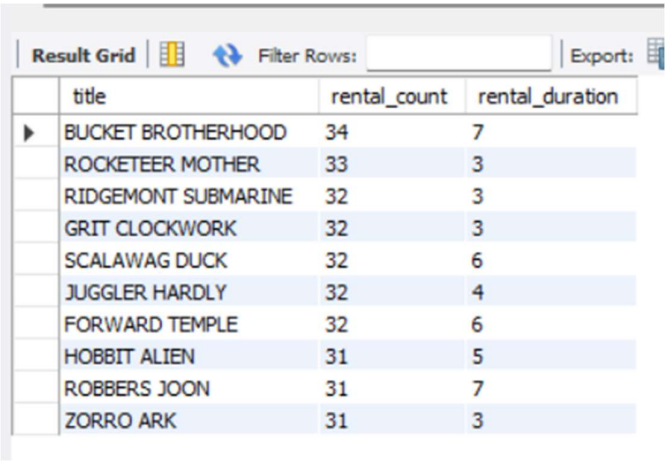
1. The film table stores information about movies. By appropriately querying the film table, find the top 10 most popular films of all time, based on the number of rentals. Your output should have 2 columns containing the film name and rental duration respectively.

```
SELECT f.title, COUNT(r.customer_id) AS rental_count, f.rental_duration
FROM film f
JOIN inventory i ON f.film_id = i.film_id
JOIN rental r ON i.inventory_id = r.inventory_id
GROUP BY f.title, f.rental_duration
ORDER BY rental_count DESC
LIMIT 10;
```



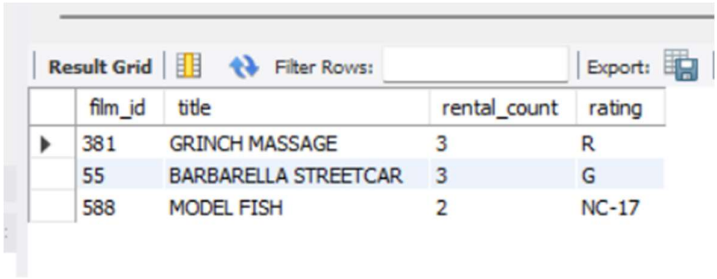
Result Grid | Filter Rows: | Export:

title	rental_count	rental_duration
BUCKET BROTHERHOOD	34	7
ROCKETEER MOTHER	33	3
RIDGEMONT SUBMARINE	32	3
GRIT CLOCKWORK	32	3
SCALAWAG DUCK	32	6
JUGGLER HARDLY	32	4
FORWARD TEMPLE	32	6
HOBBIT ALIEN	31	5
ROBBERS JOON	31	7
ZORRO ARK	31	3

2. Which are the top 3 most popular films among customers who have also rented the film TEQUILA PAST? Your output should have 4 columns- film id, film name, rental count (in descending order) and rating.

```
SELECT f.film_id, f.title, COUNT(r.customer_id) AS rental_count, f.rating
FROM rental r
JOIN inventory i ON r.inventory_id = i.inventory_id
JOIN film f ON i.film_id = f.film_id
WHERE r.customer_id IN (
    SELECT DISTINCT c.customer_id
    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
    WHERE f.title = 'TEQUILA PAST'
)
AND f.title != 'TEQUILA PAST'
GROUP BY f.title
ORDER BY rental_count DESC
LIMIT 3;
```

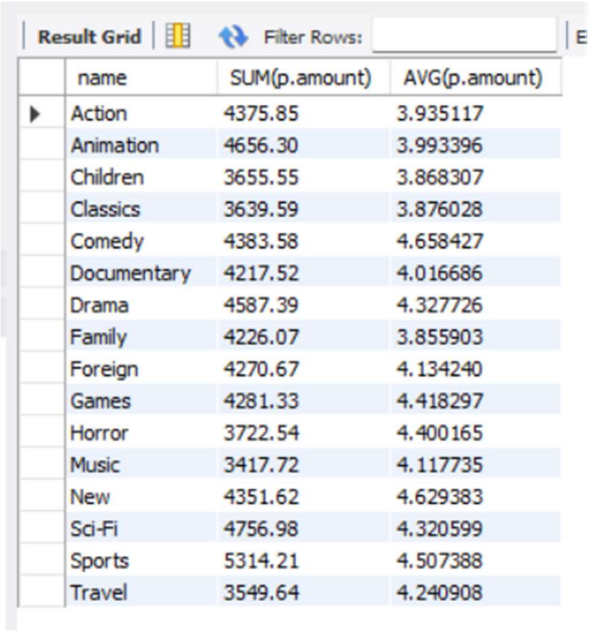


Result Grid | Filter Rows: | Export:

film_id	title	rental_count	rating
381	GRINCH MASSAGE	3	R
55	BARBARELLA STREETCAR	3	G
588	MODEL FISH	2	NC-17

3. Calculate the total revenue generated by each film category (ex: Action, Drama, Sports, etc) and list the category name along with their total revenue, as well as the average revenue per film in each category. Your output shall thus have 3 columns

```
SELECT DISTINCT c.name, SUM(p.amount), AVG(p.amount)
FROM category c
JOIN film_category fc ON c.category_id = fc.category_id
JOIN film f ON fc.film_id = f.film_id
JOIN inventory i ON f.film_id = i.film_id
JOIN rental r ON i.inventory_id = r.inventory_id
JOIN payment p ON r.rental_id = p.rental_id
GROUP BY c.name;
```



Result Grid | Filter Rows:

name	SUM(p.amount)	AVG(p.amount)
Action	4375.85	3.935117
Animation	4656.30	3.993396
Children	3655.55	3.868307
Classics	3639.59	3.876028
Comedy	4383.58	4.658427
Documentary	4217.52	4.016686
Drama	4587.39	4.327726
Family	4226.07	3.855903
Foreign	4270.67	4.134240
Games	4281.33	4.418297
Horror	3722.54	4.400165
Music	3417.72	4.117735
New	4351.62	4.629383
Sci-Fi	4756.98	4.320599
Sports	5314.21	4.507388
Travel	3549.64	4.240908

4. Use a nested select query to find the first name and the last name of customers who paid more than 10 dollars in any transaction. You have to look at the payment table and then match the customer id with that in the customer table. How many such customers are there?

```
SELECT first_name, last_name

FROM customer

WHERE customer_id IN (SELECT DISTINCT customer_id

        FROM payment

        WHERE amount > 10);
```

Result Grid | Filter Rows

first_name	last_name
PATRICIA	JOHNSON
LINDA	WILLIAMS
NANCY	THOMAS
KAREN	JACKSON
MICHELLE	CLARK
ANGELA	HERNANDEZ
ANNA	HILL
JANET	PHILLIPS
JOYCE	EDWARDS
DIANE	COLLINS
TERESA	ROGERS
IRENE	PRICE
LORI	WOOD
JACQUEL...	LONG
WANDA	PATTERSON
RITA	GRAHAM
GRACE	ELLIS
VICTORIA	GIBSON
ELAINE	STEVENS
CARRIE	PORTER
ANITA	MORALES
EVA	RAMOS
ELEANOR	HUNT
VALERIE	BLACK
CATHY	SPENCER
REGINA	BERRY
MARION	SNYDER
BRITTANY	RILEY
VANESSA	SIMS

Result Grid

count(*)
107

5. Determine the top 10 customers who have rented the most films. Your output should include their total rental count and the total amount they've spent on rentals.

```
SELECT c.customer_id, c.first_name, c.last_name,
   (SELECT COUNT(*) FROM rental r WHERE r.customer_id = c.customer_id) AS rental_count,
   (SELECT SUM(p.amount) FROM payment p WHERE p.customer_id = c.customer_id) AS
total_amount_spent
FROM customer c
ORDER BY rental_count DESC
LIMIT 10;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Con

customer_id	first_name	last_name	rental_count	total_amount_spent
148	ELEANOR	HUNT	46	216.54
526	KARL	SEAL	45	221.55
144	CLARA	SHAW	42	195.58
236	MARCIA	DEAN	42	175.58
75	TAMMY	SANDERS	41	155.59
197	SUE	PETERS	40	154.60
469	WESLEY	BULL	40	177.60
178	MARION	SNYDER	39	194.61
137	RHONDA	KENNEDY	39	194.61
468	TIM	CARY	39	175.61

6. Now, request a solution from a large language model (LLM) like ChatGPT or BARD for each of these problems. After obtaining the solutions, compare them with your solutions. (Advice: LLMs do not always generate an appropriate code required. Therefore, it is recommended to solve the questions in the order provided, first by yourself, and then prompt the LLM).

-- question 1 -

```
SELECT title, rental_duration
FROM film
JOIN inventory ON film.film_id = inventory.film_id
JOIN rental ON inventory.inventory_id = rental.inventory_id
GROUP BY title, rental_duration
ORDER BY COUNT(*) DESC
LIMIT 10;
```

Result Grid | Filter Rows:

title	rental_duration
BUCKET BROTHERHOOD	7
ROCKETEER MOTHER	3
RIDGEMONT SUBMARINE	3
GRIT CLOCKWORK	3
SCALAWAG DUCK	6
JUGGLER HARDLY	4
FORWARD TEMPLE	6
HOBBIT ALIEN	5
ROBBERS JOON	7
ZORRO ARK	3

-- question 2 -

```
SELECT f.film_id AS Film_ID, f.title AS Film_Name, COUNT(r.rental_id) AS Rental_Count, f.rating AS
Rating

FROM film AS f

JOIN inventory AS i ON f.film_id = i.film_id

JOIN rental AS r ON i.inventory_id = r.inventory_id

WHERE r.customer_id IN (

   SELECT customer_id

   FROM rental

   JOIN inventory ON rental.inventory_id = inventory.inventory_id

   JOIN film ON inventory.film_id = film.film_id

   WHERE film.title = 'TEQUILA PAST'

)

AND f.title != 'TEQUILA PAST'

GROUP BY f.film_id, f.title, f.rating

ORDER BY Rental_Count DESC

LIMIT 3;
```


Result Grid | Filter Rows: | Export:

Film_ID	Film_Name	Rental_Count	Rating
381	GRINCH MASSAGE	3	R
55	BARBARELLA STREETCAR	3	G
588	MODEL FISH	2	NC-17

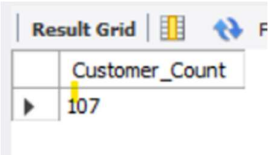
```
-- question 3 -


SELECT category.name, SUM(payment.amount) AS total_revenue, SUM(payment.amount) /
COUNT(DISTINCT film.film_id) AS average_revenue_per_film

FROM category

JOIN film_category ON category.category_id = film_category.category_id

JOIN film ON film_category.film_id = film.film_id

JOIN inventory ON film.film_id = inventory.film_id

JOIN rental ON inventory.inventory_id = rental.inventory_id
```

```
JOIN payment ON rental.rental_id = payment.rental_id
GROUP BY category.name
ORDER BY total_revenue DESC;
```

Result Grid | Filter Rows: | Export:

name	total_revenue	average_revenue_per_film
Sports	5314.21	72.797397
Sci-Fi	4756.98	80.626780
Animation	4656.30	72.754688
Drama	4587.39	75.203115
Comedy	4383.58	78.278214
Action	4375.85	71.735246
New	4351.62	72.527000
Games	4281.33	73.816034
Foreign	4270.67	63.741343
Family	4226.07	63.075672
Documen...	4217.52	66.944762
Horror	3722.54	70.236604
Children	3655.55	63.026724
Classics	3639.59	67.399815
Travel	3549.64	66.974340
Music	3417.72	67.014118

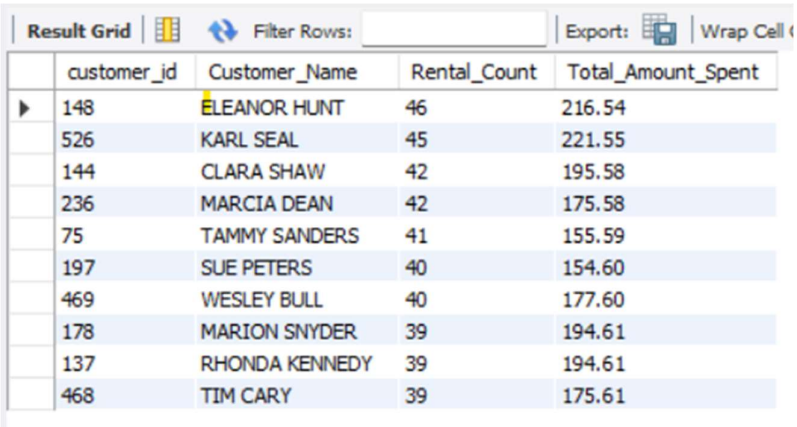
```
-- question 4 -

SELECT
   COUNT(*) AS Customer_Count
FROM
   customer
WHERE
   customer_id IN (
      SELECT DISTINCT
         p.customer_id
      FROM
         payment AS p
      WHERE
         p.amount > 10
   );
```



Customer_Count
107

-- question 5 -

```
SELECT
    c.customer_id,
    CONCAT(c.first_name, ' ', c.last_name) AS Customer_Name,
    COUNT(r.rental_id) AS Rental_Count,
    SUM(p.amount) AS Total_Amount_Spent
FROM
    customer AS c
JOIN
    rental AS r ON c.customer_id = r.customer_id
JOIN
    payment AS p ON r.rental_id = p.rental_id
GROUP BY
    c.customer_id, Customer_Name
ORDER BY
    Rental_Count DESC
LIMIT 10;
```



customer_id	Customer_Name	Rental_Count	Total_Amount_Spent
148	ELEANOR HUNT	46	216.54
526	KARL SEAL	45	221.55
144	CLARA SHAW	42	195.58
236	MARCIA DEAN	42	175.58
75	TAMMY SANDERS	41	155.59
197	SUE PETERS	40	154.60
469	WESLEY BULL	40	177.60
178	MARION SNYDER	39	194.61
137	RHONDA KENNEDY	39	194.61
468	TIM CARY	39	175.61