1. \*\*Total revenue generated by each category:\*\*

    WITH CategoryRevenue AS (

        SELECT c.category\_id, c.name AS category\_name,

               SUM(p.amount) AS total\_revenue

        FROM payment p

        JOIN rental r ON p.rental\_id = r.rental\_id

        JOIN inventory i ON r.inventory\_id = i.inventory\_id

        JOIN film\_category fc ON i.film\_id = fc.film\_id

        JOIN category c ON fc.category\_id = c.category\_id

        GROUP BY c.category\_id, c.name

    )

    SELECT category\_name, total\_revenue

    FROM CategoryRevenue

    ORDER BY total\_revenue DESC;

2. \*\*Top 5 customers based on total rental amount:\*\*

    WITH CustomerRentals AS (

        SELECT c.customer\_id, CONCAT(c.first\_name, ' ', c.last\_name) AS customer\_name,

               SUM(p.amount) AS total\_amount

        FROM payment p

        JOIN customer c ON p.customer\_id = c.customer\_id

        GROUP BY c.customer\_id, customer\_name

    )

    SELECT customer\_name, total\_amount

    FROM CustomerRentals

    ORDER BY total\_amount DESC

    LIMIT 5;

3. \*\*Monthly rental count for each film category, organized by year:\*\*

    WITH MonthlyRentals AS (

        SELECT DATE\_FORMAT(r.rental\_date, '%Y-%m') AS rental\_month,

               fc.category\_id, c.name AS category\_name,

               COUNT(r.rental\_id) AS rental\_count

        FROM rental r

        JOIN inventory i ON r.inventory\_id = i.inventory\_id

        JOIN film\_category fc ON i.film\_id = fc.film\_id

        JOIN category c ON fc.category\_id = c.category\_id

        GROUP BY rental\_month, fc.category\_id, category\_name

    )

    SELECT rental\_month, category\_name, rental\_count

    FROM MonthlyRentals

    ORDER BY rental\_month, category\_id;

4. \*\*Average rental duration for each customer:\*\*

    WITH AvgRentalDuration AS (

        SELECT r.customer\_id, AVG(DATEDIFF(r.return\_date, r.rental\_date)) AS avg\_duration

        FROM rental r

        GROUP BY r.customer\_id

    )

    SELECT c.first\_name, c.last\_name, ard.avg\_duration

    FROM customer c

    JOIN AvgRentalDuration ard ON c.customer\_id = ard.customer\_id;

5. \*\*Films rented more than the average number of times:\*\*

    WITH AvgFilmRental AS (

        SELECT f.film\_id, COUNT(r.rental\_id) AS rental\_count

        FROM film f

        LEFT JOIN inventory i ON f.film\_id = i.film\_id

        LEFT JOIN rental r ON i.inventory\_id = r.inventory\_id

        GROUP BY f.film\_id

    ), AverageRentals AS (

        SELECT AVG(rental\_count) AS avg\_rentals

        FROM AvgFilmRental

    )

    SELECT f.title, afr.rental\_count

    FROM film f

    JOIN AvgFilmRental afr ON f.film\_id = afr.film\_id

    CROSS JOIN AverageRentals ar

    WHERE afr.rental\_count > ar.avg\_rentals;

6. \*\*Top 10 films generating the most revenue with category and rental count:\*\*

    WITH FilmRevenue AS (

        SELECT f.film\_id, f.title, c.name AS category, COUNT(r.rental\_id) AS rental\_count,

               SUM(p.amount) AS total\_revenue

        FROM film f

        JOIN film\_category fc ON f.film\_id = fc.film\_id

        JOIN category c ON fc.category\_id = c.category\_id

        LEFT JOIN inventory i ON f.film\_id = i.film\_id

        LEFT JOIN rental r ON i.inventory\_id = r.inventory\_id

        LEFT JOIN payment p ON r.rental\_id = p.rental\_id

        GROUP BY f.film\_id, f.title, c.name

    )

    SELECT title, category, rental\_count, total\_revenue

    FROM FilmRevenue

    ORDER BY total\_revenue DESC

    LIMIT 10;

7. \*\*Customers who rented the same film more than once in a month:\*\*

    WITH CustomerRentalCounts AS (

        SELECT customer\_id, film\_id, DATE\_FORMAT(rental\_date, '%Y-%m') AS rental\_month,

               COUNT(rental\_id) AS rentals\_in\_month

        FROM rental

        GROUP BY customer\_id, film\_id, rental\_month

    )

    SELECT CONCAT(c.first\_name, ' ', c.last\_name) AS customer\_name, f.title, crc.rental\_month

    FROM customer c

    JOIN CustomerRentalCounts crc ON c.customer\_id = crc.customer\_id

    JOIN film f ON crc.film\_id = f.film\_id

    WHERE crc.rentals\_in\_month > 1;

8. \*\*Films that have not been rented yet:

    WITH RentedFilms AS (

        SELECT DISTINCT inventory.film\_id

        FROM rental

        JOIN inventory ON rental.inventory\_id = inventory.inventory\_id

    )

    SELECT film\_id, title

    FROM film

    WHERE film\_id NOT IN (SELECT film\_id FROM RentedFilms);

9. \*\*Customers who rented at least one film from each category:\*\*

    WITH CustomerCategoryCounts AS (

        SELECT c.customer\_id, fc.category\_id, COUNT(DISTINCT f.film\_id) AS films\_in\_category

        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\_category fc ON i.film\_id = fc.film\_id

        JOIN film f ON fc.film\_id = f.film\_id

        GROUP BY c.customer\_id, fc.category\_id

    ), CategoryCount AS (

        SELECT category\_id, COUNT(\*) AS total\_categories

        FROM category

        GROUP BY category\_id

    )

    SELECT customer\_id, COUNT(\*) AS matched\_categories

    FROM CustomerCategoryCounts

    GROUP BY customer\_id

    HAVING COUNT(\*) = (SELECT COUNT(\*) FROM CategoryCount);

10. \*\*Breakdown of rentals made by each customer in their first month:\*\*

    WITH FirstMonthRentals AS (

        SELECT customer\_id, DATE\_FORMAT(rental\_date, '%Y-%m') AS rental\_month, COUNT(rental\_id) AS rentals

        FROM rental

        GROUP BY customer\_id, rental\_month

    )

    SELECT customer\_id, rental\_month, rentals

    FROM FirstMonthRentals

    WHERE rental\_month = (

        SELECT MIN(DATE\_FORMAT(rental\_date, '%Y-%m')) FROM rental

        GROUP BY customer\_id

    );