SQL Window Functions Assignment

Sakila Database

Objective

This assignment tests your knowledge of window functions including RANK(), DENSE_RANK(), ROW_NUMBER(), LAG(), and aggregate window functions. Consult the Sakila ERD diagram to identify the appropriate tables and their relationships. You will need to use joins, CTEs, and window functions to solve these problems.

Question 1: ROW_NUMBER - Easy

Assign a row number to each payment ordered by payment amount in descending order. Display payment_id, customer_id, amount, and the row number.

> ***Hint:*** *No joins needed, use the payments table directly.*

Question 2: RANK - Easy

Rank all films by their rental_duration in descending order. Show film_id, title, rental_duration, and the rank. Notice how films with the same rental duration receive the same rank.

> ***Hint:*** *No joins needed, single table query.*

Question 3: DENSE_RANK with Partition - Medium

For each film category, rank films by their rental_rate from highest to lowest using DENSE_RANK. Display category name, film title, rental_rate, and the rank within each category.

> ***Hint:*** *Find the category table, join to films through a junction table.*

Question 4: Top N per Group with CTE - Medium

Find the top 3 most expensive films (by replacement_cost) in each rating category (G, PG, PG-13, R, NC-17). Use a CTE to filter the window function results.

> ***Hint:*** *Single table with CTE needed to filter on the window function result.*

Question 5: LAG Function - Medium

For each customer, show their payment history with the previous payment amount. Display customer's first_name, last_name, payment_date, amount, and previous_amount (using LAG). Order by customer and payment date.

> ***Hint:*** *Join customer information to payment data.*

Question 6: Running Total - Medium

Calculate the running total of payments for each customer. Use a CTE to show only customers whose total payments exceed $150. Display customer name and their running total at each payment.

> ***Hint:*** *Join tables for customer names, aggregate using window functions, filter with CTE.*

Question 7: Complex Multi-Window Analysis - Hard

For each store, find the films that are ranked 2nd in terms of rental count. Your query should:
• Count how many times each film has been rented per store
• Rank films within each store by rental count (handle ties with DENSE_RANK)
• Use a CTE to filter only the films with rank = 2
• Display store_id, film title, rental count, and rank

> ***Hint:*** *Join rental records to inventory to link stores and films. Aggregate the rental counts, apply window function with partitioning, then use CTE to filter.*

Bonus Tip

Remember the three ranking functions:
• **ROW_NUMBER():** Always unique (1, 2, 3, 4, 5)
• **RANK():** Skips ranks after ties (1, 2, 2, 4, 5)
• **DENSE_RANK():** No gaps after ties (1, 2, 2, 3, 4)

Choose DENSE_RANK() when you need to ensure rank 2 or rank 3 actually exists!