

DVD Rental Database Project - SQL Analysis

This project demonstrates my SQL skills by performing various queries on a DVD rental database. It showcases my ability to retrieve and analyze data using a range of SQL commands. The queries cover essential database operations, including filtering, aggregation, and table creation.

Project Overview:

The project analyzes customer, film, and payment data within a DVD rental context. I've explored the data to answer business-related questions, providing insights into customer behavior, film popularity, and financial transactions. The project also includes the design and creation of a new table to manage potential customer leads.

Key Queries and Insights:

Payment Transactions > \$5.00: I determined the number of payment transactions exceeding \$5.00, giving insight into common transaction values. This uses COUNT and a WHERE clause for filtering.

Actors with First Name Starting with 'P': I identified the count of actors whose first name begins with 'P', demonstrating string pattern matching using the LIKE operator and wildcards.

Unique Customer Districts: I calculated the number of distinct districts our customers come from, using the COUNT and DISTINCT keywords to identify unique values.

List of Distinct Districts: I retrieved the names of those unique districts, building on the previous query and showcasing how to extract specific data.

Films with "Truman" in Title: I counted the films with "Truman" in the title, again using the LIKE operator and wildcards to search within text fields.

Potential Leads Table Creation: I designed and created a potential_leads table, including appropriate data types (INT, VARCHAR, DATE), constraints (PRIMARY KEY, AUTO_INCREMENT, NOT NULL, UNIQUE), and an auto-incrementing lead_id. This demonstrates database design principles and the creation of efficient tables for data management.

Customer with Highest ID (Name starts with 'E', Address ID < 500): I identified the customer with the highest ID meeting specific criteria, showcasing combined filtering with WHERE, string matching with LIKE, numerical comparisons, and the MAX function for efficiency.

Customers Spending \$110+ with Staff ID 2: I identified customers who spent at least \$110 with a specific staff member, using GROUP BY, aggregation with SUM, and the HAVING clause to filter aggregated results.

Films with Rating 'R' and Replacement Cost \$5-\$15: I counted films meeting criteria related to rating and replacement cost, demonstrating combined filtering with WHERE and the BETWEEN operator.

Maximum Payment Transaction: I found the highest payment transaction amount, using the MAX function for direct retrieval of the maximum value.