SQL Final project for CodersLab

Project Description: Financial Database Analysis Using SQL

Overview

This project focuses on financial data analysis using SQL, exploring different aspects of loans, client behavior, account balances, and credit card expiration tracking. The dataset consists of real anonymized banking data, including over 5,300 clients, nearly 700 granted loans, and approximately 900 issued credit cards. The analysis covers loan history, repayment status, client demographics, and banking transactions.

Key Analyses and Queries

1. History of Granted Loans

- **Objective:** Summarize loans by year, quarter, and month to track lending trends.
- **Method:** Uses `ROLLUP` to aggregate total loan amount, average loan amount, and total loan count across different time periods.
- **Key Insight:** `NULL` values in results represent aggregated totals (e.g., `NULL` in `month_of_loan` means a quarterly total).

2. Loan Status Analysis

- **Objective:** Identify the number of granted vs. non-granted loans.
- **Finding:**
  - 682 loans exist in the database.
  - 606 loans (A & C statuses) are fully repaid.
  - 76 loans (B & D statuses) were not repaid.
- **Conclusion:** Loan statuses A & C represent repaid loans, while B & D represent unpaid loans.

3. Account Ranking Analysis

- **Objective:** Rank accounts based on:
  - The number of granted loans.
  - The total amount of loans.
  - The average loan amount.
- **Method:** Uses `CTE` and `ROW_NUMBER()` for ranking, instead of simple `ORDER BY`.

4. Fully Paid Loans by Client Gender

- **Objective:** Determine the number of fully repaid loans by gender.
- **Finding:**
    - Males have repaid more loans than females.
    - The analysis filters only the primary loan owner (`disp.type = 'OWNER'`).

5. Client Age Analysis

- **Objective:** Calculate the average age of borrowers.
- **Finding:**
    - Male average age: 77 years
    - Female average age: 75 years
- **Method:** Uses a temporary table (`client_analysis1`) for calculations.

6. Loan Analysis by Region

- **Objective:** Identify:
    - The district with the highest number of clients.
    - The district where the most loans were paid.
    - The district where the highest total loan amount was repaid.
- **Method:** Uses `CTE` and window functions (`SUM OVER()`) to determine each district's loan share.

7. Client Selection (High Balance, Multiple Loans, Young Age)

- **Objective:** Find clients who:
    - Have account balances over 1,000.
    - Have more than 5 loans.
    - Were born after 1990.
- **Finding:**
    - No clients match all conditions because no one in the dataset was born after 1990.
    - The maximum number of loans per client is 1.

8. Credit Card Expiration Analysis

- **Objective:** Track expiring credit cards to ensure timely replacements.
- **Finding:**
    - The most recent card was issued on 29-Dec-1998.
    - Since cards expire after 3 years, the latest expiration was in 2001.
    - Using `CURDATE()` (2025) doesn't return any results, as all cards are long expired.
    - Solution: We used '2001-01-01' as a simulated date for historical analysis.

9. Stored Procedure: Expiring Cards Notification

- **Objective:** Identify clients whose cards are set to expire soon (issued + 3 years - 7 days).
- **Implementation:**
  - Created procedure `expiring_cards_VELCOVSKY` to check expiring cards.
  - Results are stored in `cards_expiration_procedure` table.
  - Simulated date '2001-01-01' is used for testing, since no cards exist beyond 2001.

Key Takeaways

- Advanced SQL techniques were used, including:
  - `WITH ROLLUP` for hierarchical aggregation.
  - `CTE` for structured queries.
  - `ROW_NUMBER()` for ranking.
  - `HAVING` to filter grouped data.
  - `WINDOW FUNCTIONS` for percentage-based district analysis.
- Dynamic stored procedures were created for:
  - Loan analysis (client ranking, status tracking).
  - Credit card expiration management.
- Challenges & Solutions:
  - No clients were born after 1990, making certain filters ineffective.
  - Using `CURDATE()` for credit card expiration returned no results → solution was using a historical test date (`2001-01-01`).
  - Loan records showed a maximum of 1 loan per client, so "more than 5 loans" filter was invalid.

Final Thought

This project successfully analyzed financial data, tracked loan history, identified key trends, and implemented stored procedures for automation. The SQL-based approach enables efficient data retrieval, ranking, and financial insights.