**Retail Sales Database**

**Final Project Documentation - Filipa Borges**

**Project Overview**

This project aims to develop a relational database system designed to manage and analyze retail sales activities, focusing on employee performance, daily sales, and the effectiveness of implemented sales strategies. The system, named RetailSalesDB, will enable management to track employee sales, customers served, receipts issued, units sold, and compare these figures against set daily objectives for each employee.

**Database Structure**

The RetailSalesDB consists of the following main tables:

**Employees:** Stores employee details, including ID and name.

**DailyCustomers:** Records the number of customers each employee serves daily.

**DailyReceipts:** Tracks the number of receipts each employee issues each day.

**UnitsSold:** Keeps count of the product units each employee sells daily.

**SalesByEmployee:** Details the sales value made by each employee, expressed in euros.

**DailyObjectives:** Sets daily sales targets for each employee.

**AdditionalMetrics:** Includes additional metrics such as conversion rate, average receipt value, cross-selling rate, and objective completion percentage.

Relationships between tables are established through foreign keys that reference EmployeeID in the Employees table, ensuring data integrity and relational structure.

**Features and Queries**

**Views:**

A view named DailySalesView will be created to combine data from the Employees, DailyReceipts, UnitsSold, and SalesByEmployee tables, providing a unified view of daily sales activities.

**Stored Functions and Procedures:**

A stored function, CalculateCrossSellingRate, will be implemented to calculate the cross-selling rate for an employee on a specific date using data from the DailySalesView.

**Triggers and Events:**

The project will include the creation of a trigger to automatically update the AdditionalMetrics table after new sales data is entered.

An event will be scheduled to generate daily sales reports summarizing the day's activities and performance metrics.

**Advanced Queries:**

A subquery example will be provided to demonstrate how to extract the percentage of sales objectives achieved by each employee on a given date, combining data from the DailyObjectives and SalesByEmployee tables.

A query utilizing GROUP BY and HAVING clauses will be prepared to analyze sales data, providing insights into employee performance and sales trends.

**Database Diagram**

The database diagram will illustrate all tables, fields, primary and foreign keys, and the relationships between tables. This visual representation will aid in understanding the database structure and data flow.

**Advanced Features and Expectations**

**Stored Procedure:**

**Objective:** Implement a stored procedure named UpdateDailySalesMetrics to automate the calculation and updating of sales metrics at the end of each day.

**Expectation:** This procedure will streamline the update process for sales metrics, ensuring real-time accuracy in reporting and analysis, and reducing manual workload.

**Trigger:**

**Objective:** Create a trigger to monitor insertions or updates in the SalesByEmployee table and automatically adjust relevant AdditionalMetrics.

**Expectation:** The trigger will maintain the integrity and timeliness of the additional metrics, providing instant updates as sales data changes, and ensuring data consistency without manual intervention.

**Event**

**Objective:** Set up an event to automatically generate summarized sales reports at a regular interval, such as daily, into a new table DailySalesSummary.

**Expectation:** This event will facilitate regular sales performance reviews, allowing for quick adjustments to sales strategies and objectives based on the latest data, enhancing decision-making processes.

**Complex View:**

**Objective:** Develop a complex view that integrates data from Employees, SalesByEmployee, DailyObjectives, and AdditionalMetrics to provide a comprehensive overview of employee performance against their objectives.

**Expectation:** This view will serve as a powerful analytical tool, enabling detailed performance analysis and identifying areas for improvement or investment, ultimately driving sales growth and efficiency.

**Advanced Query with GROUP BY and HAVING:**

**Objective:** Craft an advanced query utilizing GROUP BY and HAVING to analyze sales data, grouping by relevant criteria such as employee, product category, or time period, and filtering results based on specific performance or sales criteria.

**Expectation:** The query will offer deep insights into sales trends, employee performance, and operational efficiency, guiding strategic decisions and identifying opportunities for optimization and growth.