Project Report: Personal Finance Tracker Database

Introduction

Managing personal finances is essential in everyday life. A structured database system helps users track income, expenses, and overall balance effectively. The Personal Finance Tracker project was designed to provide insights into financial health using SQL queries and database management principles.

Abstract

This project focuses on building a normalized SQL database schema to manage personal finances. It categorizes income and expenses, calculates balances, and generates meaningful insights through SQL queries and views. The project demonstrates the use of normalization, joins, aggregate functions, and reporting capabilities in SQL.

Tools Used

- MySQL / PostgreSQL for database management
- dbdiagram.io for ER diagram design
- SQL Workbench for query execution and reporting

Steps Involved in Building the Project:

1. Database Schema Design

- Identified key entities: **Users, Categories, Income, Expenses, Balance**.
- Designed the schema using **dbdiagram.io** to visualize relationships.
- Applied **normalization up to 3NF** to avoid redundancy.

Example: Instead of storing "Food" text in every expense row, a separate **Categories** table was created.

2. Table Creation (DDL)

- Created tables in **MySQL** with proper constraints:
  - `PRIMARY KEY` for unique identification.
  - `FOREIGN KEY` for relationships between tables.
  - `CHECK` constraints for valid values (e.g., `category_type` = 'Income' or 'Expense').

This ensured **data integrity** and prevented invalid entries.

3. Data Population (DML)

- Inserted **70+ rows** of dummy financial data for 5 users.

- Added realistic income categories (*Salary, Freelance, Investments, Gifts, Bonus*) and expense categories (*Food, Rent, Transport, Shopping, Entertainment*).
- Used multiple payment methods (*Cash, Card, UPI*).

This gave enough data to test **reports and queries**.

4. Analytical SQL Queries

- Wrote queries for **monthly expense summaries** using `GROUP BY`.
- Created **category-wise analysis** to understand top spending areas.
- Used **aggregate functions** (`SUM`, `AVG`, `COUNT`) for reports.
- Example queries:
  - Monthly expenses per user.
  - Percentage of spending by category.
  - Users spending more than their income.

These queries provide **insights into user financial habits**.

5. Views for Balance Tracking

- Created **SQL Views** to simplify reporting:
  - `UserBalance` → total income, total expenses, net balance per user.
  - `MonthlyBalance` → month-by-month balance tracking.

Views acted as **virtual tables**, making it easy to query balances without recalculating every time.

6. Report Generation & Export

- Combined income and expense queries to generate monthly financial reports.
- Exported results from SQL Workbench into CSV/Excel/PDF.
- Reports show:
  - Income vs Expenses per month.
  - Net Balance trends.
  - Top categories where money is spent.

These reports simulate real-world financial statements**.**

This workflow demonstrates database design, SQL queries, normalization, data analysis, and reporting - exactly the skills interviewers look for when you explain projects.

Conclusion

The *Personal Finance Tracker* project successfully demonstrates how databases can be used to solve real-life problems such as managing personal budgets. By designing a normalized schema, inserting realistic data, and writing analytical SQL queries, the system provides clear insights into income, expenses, and savings.

Through **monthly reports, category-wise spending breakdowns, and balance tracking views**, users can understand their financial habits and make better decisions.

From a technical perspective, this project helped in strengthening concepts of:

- **Database normalization (3NF)** to remove redundancy,
- **Joins and aggregate functions** for data analysis,
- **Views** for simplified reporting, and
- **Exporting reports** for real-world usability.

Overall, the project not only improved database design and SQL query skills but also provided practical knowledge in financial data management, making it a valuable experience for both academic learning and professional interviews.