

Employee Management and Attendance Tracker

Introduction:

In modern organizations, managing employee records and attendance data is a critical task for HR and operations departments. Traditional systems are either manual or semi-automated and often lack scalability and reporting features.

This project aims to create a robust and normalized SQL database system that efficiently tracks employee information, department associations, roles, daily attendance (login/logout), and enables various HR-related analytics using PostgreSQL and pgAdmin.

Abstract:

The goal of this project is to design and implement a scalable database for managing employee details and attendance logs. The system uses PostgreSQL for its reliability and strong relational capabilities. Key functionalities include auto-marking late arrivals via triggers, calculating total working hours using custom SQL functions, and generating reports using SQL queries.

The database includes four core tables: Employees, Department, Roles, and Attendance. Dummy data for 200 employees and their 3-day attendance was generated using Python (Faker library) and inserted via SQL scripts.

Tools Used:

PostgreSQL – Database Engine

pgAdmin 4 – Database Management GUI

Python (Faker) – Data generation

SQL – Schema design, query writing, and automation

GitHub – Code versioning and project hosting

Steps Involved in Building the Project:

1. Schema Design:

Tables created for Departments, Roles, Employees, and Attendance with proper primary/foreign key constraints and normalization.

2. Data Generation:

Used Python Faker to create 200 employees with random names, join dates, and contacts. Each employee was assigned to one of 5 departments and roles. Three attendance records were created for each employee.

3. Data Insertion:

SQL scripts were written for inserting department and role data, followed by generated employee and attendance records.

4. Automation with Triggers:

A trigger was added to the Attendance table to automatically classify records as "Late" if login was after 9:00 AM.

5. Custom Functions:

A SQL function was created to calculate total work hours for any employee by summing login-logout time differences.

6. Reporting Queries:

Queries were written to generate:

Monthly attendance per employee

Late arrival report

Total work hours summary

Conclusion:

This project provided hands-on experience with relational database design, SQL scripting, and database automation using triggers and functions. It demonstrates the practical use of SQL in managing HR and operations data. The project is scalable, extendable to include leave tracking, payroll, and user roles.

This project enhanced my understanding of database normalization, SQL performance, and real-time reporting capabilities using PostgreSQL.