Project Report

Title:

Design and Implementation of a Relational Database System for Organizational Management

Author:

Riad Hossain
Course: CPS 541 – Database Systems
Institution: Central Michigan University

Abstract

This project focuses on designing and implementing a relational database system to model an organization's structure, encompassing employees, departments, projects, and dependents. The database was created using Oracle SQL to demonstrate the relationships between different entities in a company and to ensure data consistency through referential integrity. Multiple queries were executed to analyze relationships between employees and supervisors, assess managerial structures, identify departmental and project-based information, and generate organizational statistics. The project highlights the effectiveness of SQL in managing, retrieving, and analyzing complex organizational data, emphasizing the importance of relational database design in real-world business environments.

---

1. Introduction

The primary objective of this project was to design a structured relational database system that models a company's operational framework. Organizations rely heavily on efficient data management systems to store and access information about their employees, departments, and projects.
This project provides a systematic demonstration of how to build a relational schema that connects these entities while maintaining data accuracy and consistency. The implementation uses SQL, one of the most powerful tools for database management, to create and manage relational tables and to execute complex analytical queries that reflect real-world business scenarios.

The project aims to:

- Establish relationships between employees, departments, projects, and dependents.
- Demonstrate database normalization and data integrity through foreign key constraints.
- Retrieve and analyze meaningful information about employees, supervisors, departments, and salaries.
- Illustrate the use of SQL for data analysis and decision-making in organizational contexts.

---

2. Database Design and Methodology

2.1 Database Schema Overview

The database was structured around six main entities:

1. **Employee** – Contains details about employees, including personal data, salary, and departmental association.
2. **Department** – Represents organizational divisions, each managed by a specific employee.
3. **Department Location** – Specifies physical locations of departments.
4. **Project** – Holds information about ongoing organizational projects.
5. **Works_On** – Represents the relationship between employees and the projects they are involved in.
6. **Dependent** – Contains information about employees' dependents.

2.2 Relational Model

Each table was connected through primary and foreign key constraints to maintain referential integrity. For example:

- Each employee is assigned to one department.
- Departments are managed by an employee.
- Projects belong to specific departments.
- Employees can work on multiple projects.

- Dependents are linked to a specific employee.

This relational structure allows efficient organization of data and easy retrieval of complex interrelated information.

2.3 Implementation Tools

The project was implemented using:

- **Oracle SQL Developer** (for writing and executing SQL commands)
- **SQL Language** (for table creation, data manipulation, and data querying)

---

3. Implementation

The implementation phase involved three key steps:

3.1 Table Creation and Data Insertion

Tables were created using the CREATE TABLE command to define entities and establish relationships. Data was inserted into each table to represent employees, departments, projects, and dependents within an organization. Primary and foreign keys were used to ensure data consistency and prevent redundancy.

3.2 Referential Integrity

Each table was linked through constraints that enforce logical relationships between entities. This ensured data accuracy — for instance, a department cannot exist without a manager, and a dependent must be associated with an existing employee.

3.3 Query Execution

Various queries were executed to extract meaningful insights. Examples include:

- Identifying employees in specific departments (e.g., Research department).
- Retrieving details of department managers and their projects.
- Listing employees who do not have supervisors.
- Calculating salary statistics such as total, average, and maximum salary.
- Counting the number of employees per department or per project.
- Identifying projects where more than two employees are involved.
- Finding employees with dependents and their relationships.

These queries reflect real-world organizational information requests that managers or HR analysts might need.

---

4. Results and Discussion

The database successfully demonstrated how a relational structure can represent complex organizational relationships. The following key outcomes were observed:

- **Hierarchical Relationships:**
  The system accurately represented supervisor-employee hierarchies and department-manager relationships.
- **Departmental Insights:**
  Queries revealed which departments have the highest number of employees and the average salary per department.
- **Project Analysis:**
  The data identified how many employees are working on each project, highlighting high-collaboration projects.
- **Employee and Dependent Information:**
  Queries effectively linked employees to their dependents, enabling HR-level insight into family and dependent data.
- **Performance and Salary Insights:**
  Aggregate functions helped calculate salary distributions and departmental performance metrics.

This analysis proved that the designed schema was robust, efficient, and capable of handling both simple and complex data queries, demonstrating the power and versatility of SQL in managing organizational data.

---

5. Conclusion

This project successfully implemented a relational database model that effectively captures the structure and relationships within an organization. It demonstrates how SQL can be used not only to manage data but also to extract valuable insights that support decision-making.
Through the creation of multiple interrelated tables and the execution of complex queries, this project highlights the importance of database normalization, referential integrity, and

efficient query formulation.
The project's outcomes illustrate how structured databases serve as the backbone of modern organizations, ensuring data accuracy, integrity, and accessibility for operational and strategic purposes.

---

6. Limitations and Future Work

While the database structure was successfully implemented, future improvements may include:

- Implementing **stored procedures and triggers** to automate updates and enforce business rules.
- Adding **views** to simplify complex queries and enhance user accessibility.
- Integrating **front-end interfaces** for user-friendly data entry and reporting.
- Expanding the database to include payroll, attendance, and performance evaluation modules.

---

7. References

- Elmasri, R., & Navathe, S. (2016). *Fundamentals of Database Systems* (7th Edition). Pearson.
- Oracle SQL Documentation. (n.d.). *Oracle Database SQL Language Reference*.
- W3Schools. (n.d.). *SQL Tutorial*. Retrieved from https://www.w3schools.com/sql