EMPLOYEE PAYROLL SYSTEM

(Mini Project using Java, JDBC, DTO, DAO, DAOImpl & Main Class)

Introduction

The Employee Payroll System is a mini project developed using Java and JDBC to manage employee salary details in an organization. This system helps in storing, calculating, updating, and retrieving employee payroll information efficiently using a MySQL database.

The application follows a layered architecture consisting of DTO, DAO, DAO Implementation, Service, and Main classes. This structure separates data representation, database operations, and business logic, making the application easy to understand and maintain. JDBC is used to establish a secure connection between the Java application and the database.

This project helps beginners understand real-world Java application development using database connectivity and design patterns commonly used in enterprise applications.

Objectives

The main objective of this project is to understand how payroll data is managed using Java and relational databases through JDBC. It demonstrates how employee salary records are stored and processed using SQL queries executed from Java programs.

Another objective is to implement DTO and DAO design patterns to separate data handling from database access logic. This improves code reusability, readability, and maintainability.

The project also aims to provide hands-on experience with CRUD operations such as adding new employees, viewing payroll details, updating salary information, and deleting employee records.

Additionally, the project introduces students to structured application design using multiple layers, which is widely used in real-time software development.

Software Requirements

- Java Development Kit (JDK) 8 or above
- MySQL Database
- Eclipse / IntelliJ IDEA IDE
- MySQL JDBC Driver

Database Design

The database design is kept simple for learning purposes. A single table named employee is used to store payroll information.

The table includes fields such as employee ID, employee name, basic salary, allowances, deductions, and net salary. The employee ID uniquely identifies each employee. Net salary is calculated based on salary components.

This simple database structure enables efficient data storage and retrieval using JDBC and supports all payroll-related operations.

System Architecture

The Employee Payroll System follows a layered architecture consisting of DTO Layer, DAO Layer, DAO Implementation Layer, Service Layer, and Main Class. This architecture ensures clean code structure and easy maintenance.

Class Description

1. DTO (Data Transfer Object) Class
The DTO class represents employee payroll data. It contains variables such as employee ID, name, basic salary, allowance, deduction, and net salary along with getters and setters.

2. DAO (Data Access Object) Interface
The DAO interface defines methods for payroll-related database operations such as add, view, update, delete, and retrieve employee records.

3. DAO Implementation (DAOImpl) Class
The DAOImpl class implements the DAO interface and contains JDBC code to execute SQL queries using PreparedStatement and ResultSet.

4. Service Class
The Service class handles business logic such as net salary calculation and validation of employee data.

5. Main Class
The Main class provides a menu-driven console interface and interacts with the Service and DAO layers based on user input.

Functional Description

When the application starts, a menu is displayed on the console. The user selects an option to perform payroll operations and enters employee and salary details.

The Service layer processes salary calculations and passes DTO objects to the DAOImpl class, which performs database operations using JDBC. Retrieved details are displayed back to the user.

Advantages

• Easy to understand and suitable for beginners
• Uses industry-standard JDBC technology
• Clear separation of layers using DTO and DAO patterns
• Improves coding standards and maintainability

Limitations

• Console-based application without GUI
• No authentication or role-based access
• Suitable only for academic and learning purposes

Conclusion

The Employee Payroll System mini project successfully demonstrates the implementation of Java, JDBC, and design patterns such as DTO and DAO. It provides practical experience in managing payroll data using database connectivity.

The project helps students understand layered architecture and prepares them for developing advanced enterprise-level applications. With further enhancements like GUI, security, and reporting features, the system can be extended into a full-scale payroll solution.