Employee Management System - Project Documentation

# Project Title:

Employee Management System

# Project Description:

This is a console-based Java application that allows a user (typically HR or admin) to manage employee records stored in a MySQL database.

The system enables Create, Read, Update, and Delete (CRUD) operations for employee data through a command-line interface.

# Technologies Used:

- Java (JDK 8 or above) – Backend programming  
- MySQL – Database  
- JDBC (Java Database Connectivity) – Connect Java with MySQL  
- IDE (e.g., IntelliJ IDEA, VS Code) – Development Environment

# Database Design:

Table: employees  
  
| Column Name | Data Type | Description |  
|-------------|----------------|------------------------------|  
| id | INT (Primary) | Employee ID (Auto Increment) |  
| name | VARCHAR(100) | Employee Full Name |  
| salary | DOUBLE | Employee Salary |  
| department | VARCHAR(50) | Department Name |

# Features:

1. Add Employee  
 - Enter employee details (name, email, salary, department)  
 - Store them into the database  
  
2. View All Employees  
 - Fetch all employee records from the database  
3. Update Employee  
 - Modify the existing employee information  
  
4. Delete Employee  
 - Remove an employee record using ID  
  
5. Exit  
 - Terminate the application

# Project Structure:

EmployeeManagementSystem/  
│  
├── Employee.java // Model class  
├── EmployeeDAO.java // Database access logic (CRUD operations)

└── Main.java // User menu and logic flow

# How It Works:

1. The user is shown a menu in the console.  
2. Based on input, the corresponding operation is executed using JDBC.  
3. Database interactions are handled in the DAO layer.  
4. Each operation gives feedback via the console (e.g., "Employee added successfully").

# Prerequisites:

- MySQL installed and running  
- Java installed  
- JDBC Driver (Connector/J) added to the project classpath  
- A MySQL database created with an employees table

# Sample Console Output:

====== Employee Management System ======  
1. Add Employee  
2. View All Employees  
3. Update Employee  
4. Delete Employee  
5. Exit  
Enter your choice:

# Future Improvements (Optional):

- GUI Interface using Swing or JavaFX  
- Authentication (Login System)  
- Export data to CSV or PDF  
- Advanced filtering/search options