2026

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND BUSINESS SYSTEMS

B.TECH – II YEAR

21CSE269T
JAVA MINI PROJECT

Project 1: Design and Implementation of Employee Payroll System Using Java

REPORT SUBMITTED BY:
ATHMAJA A
RA2411042020011

OBJECTIVE

The primary objective of my project is to design and implement a simple Employee Payroll System using the Java programming language. The system aims to automate the process of salary calculation and employee record management.

My project is developed to achieve the following objectives:

- To understand and apply Object-Oriented Programming (OOP) concepts.
- To implement constructors and method overloading in a real-world application.
- To perform file handling operations for storing and retrieving data.
- To design a menu-driven application using Java.
- To calculate salary components such as HRA and DA programmatically.
- To create a simple system that allows searching employee records efficiently.

Through my project, practical knowledge of Java programming concepts is strengthened by applying them to a real-life payroll management scenario.

PROBLEM DESCRIPTION

In many small organizations, employee salary records are managed manually or maintained in spreadsheets. This approach can lead to errors in salary calculations, difficulty in maintaining records, and inefficiency in searching employee details.

Payroll management involves:

- Storing employee information
- Calculating salary components
- Managing persistent records
- Retrieving employee data when required

Manual calculation of salary components such as House Rent Allowance (HRA) and Dearness Allowance (DA) may result in computational mistakes. Additionally, without proper storage mechanisms, retrieving employee data becomes difficult.

Therefore, my project proposes a simple automated solution using Java that:

1. Stores employee details including ID, name, and basic salary.
2. Calculates:

- HRA = 20% of Basic Salary
- DA = 10% of Basic Salary
- Net Salary = Basic Salary + HRA + DA

3. Stores employee records in a file.
4. Allows users to search for employees using Employee ID.
5. Displays employee details in a structured format.

The system is menu-driven and user-friendly, allowing interaction through the console.

SYSTEM DESIGN AND CLASS DESIGN

Overall System Design:

The system follows a simple Object-Oriented design consisting of:

- One class to represent Employee data.
- One main class to manage the payroll operations.

The program is structured to separate data representation from operational logic.

➔ Employee Class Design

Data Members:

1. empId – Integer variable to store Employee ID.
2. name – String variable to store Employee Name.
3. basicSalary – Double variable to store Basic Salary.

Constructors:

1. Default Constructor
   - Initializes employee details with default values.
2. Parameterized Constructor
   - Accepts employee ID, name, and basic salary as parameters and assigns them to instance variables.

Methods:

- calculateSalary()
  - Calculates net salary by adding HRA and DA to the basic salary.
- calculateSalary(double bonus)
  - Overloaded method that adds bonus to the net salary.
- display()

- Displays employee details in a formatted manner.

- toFileString()
  - Converts employee details into a comma-separated string for file storage.

➔ PayrollSystem Class Design

This class contains:

- The main() method.
- A menu-driven interface.
- File handling methods.

Methods in PayrollSystem:

- addEmployee()
  - Accepts user input and writes employee data to file.
- viewEmployees()
  - Reads all employee records from file and displays them.
- searchEmployee()
  - Searches for an employee by ID in the file.
  - The system uses linear search while reading records from the file.

IMPLEMENTATION AND CODE EXPLANATION

Salary Calculation Logic

The salary is calculated using the following formulas:

- HRA = 20% of Basic Salary
- DA = 10% of Basic Salary
- Net Salary = Basic Salary + HRA + DA

These calculations are implemented inside the calculateSalary() method of the Employee class.

Method Overloading

Method overloading is demonstrated using two methods:

1. calculateSalary()
2. calculateSalary(double bonus)

The second method allows calculation of salary including an additional bonus amount, thereby demonstrating polymorphism.

File Handling

The system uses text file storage (employees.txt) to store employee records.

- FileWriter is used to write data into the file.
- BufferedReader is used to read data from the file.
- Records are stored in comma-separated format:

This ensures that employee data remains available even after the program is closed.

Searching Mechanism

The search functionality works as follows:

1. The program reads each line from the file.
2. Splits the line using comma as delimiter.
3. Compares the employee ID with the entered search ID.
4. Displays the employee details if a match is found.

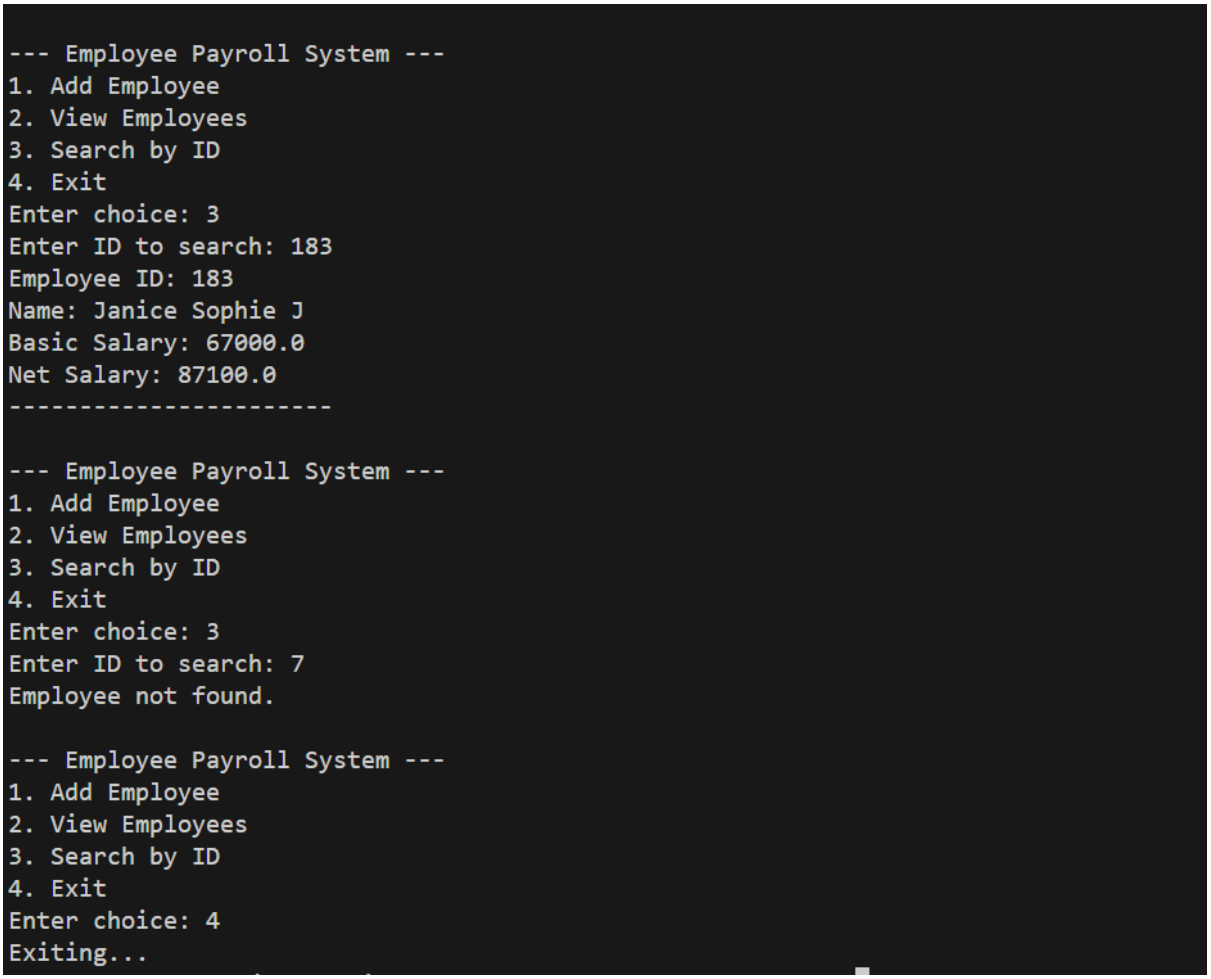
This approach ensures efficient retrieval for small-scale systems.

INPUT AND OUTPUT SCREEN SHOTS

```
PS C:\Users\Athmaja\OneDrive\Desktop\Java Employee_Payroll>  & 'C:\Program Files\Java
\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Athma
ja\AppData\Roaming\Code\User\workspaceStorage\8781733397424adb20c378d549ec8bf4\redhat
.java\jdt_ws\Java Employee_Payroll_de5aafe2\bin' 'PayrollSystem'

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 1
Enter Employee ID: 101
Enter Name: Ram Kumar
Enter Basic Salary: 65000
Employee added successfully!

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 1
Enter Employee ID: 261
Enter Name: Jithesh P
Enter Basic Salary: 30000
Employee added successfully!

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 1
Enter Employee ID: 047
Enter Name: Jayesh Reddy
```

```
--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 1
Enter Employee ID: 047
Enter Name: Jayesh Reddy
Enter Basic Salary: 55000
Employee added successfully!

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 183
Invalid choice

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 1
Enter Employee ID: 183
Enter Name: Janice Sophie J
Enter Basic Salary: 67000
Employee added successfully!

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
```

```
--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 2
Employee ID: 101
Name: Ram Kumar
Basic Salary: 65000.0
Net Salary: 84500.0
----------------------
Employee ID: 261
Name: Jithesh P
Basic Salary: 30000.0
Net Salary: 39000.0
----------------------
Employee ID: 47
Name: Jayesh Reddy
Basic Salary: 55000.0
Net Salary: 71500.0
----------------------
Employee ID: 183
Name: Janice Sophie J
Basic Salary: 67000.0
Net Salary: 87100.0
----------------------

--- Employee Payroll System ---
1. Add Employee
2. View Employees
3. Search by ID
4. Exit
Enter choice: 3
Enter ID to search: 183
Employee ID: 183
```



FUTURE ENHANCEMENTS

The system can be further enhanced by:

- Adding graphical user interface (GUI).
- Using database instead of text file.
- Implementing update and delete functionalities.
- Adding password-based authentication.
- Generating salary slips automatically.

CONCLUSION

The Employee Payroll System was successfully implemented using Java. The project effectively demonstrates key programming concepts such as:

- Object-Oriented Programming
- Constructors

- Method Overloading
- File Handling
- Exception Handling
- Menu-driven programming

The system automates salary calculation and provides a simple mechanism for managing employee records. It improves accuracy, reduces manual effort, and ensures proper data storage.

My project serves as a foundation for developing more advanced payroll management systems in the future.

GitHub Link: https://github.com/aa6875-sketch/JAVA-Project-Employee-Payroll-System-