Week 10 - In-Class Exercise for Training and Documentation

# Managing an Outdated Payroll System

## Context

Your company has completed the acquisition of a competitor and among your new assets is their payroll system that is internally developed. The team that built it has generally quit, or retired, and the company had previously outsourced the maintenance of the software with mixed, but generally negative results.

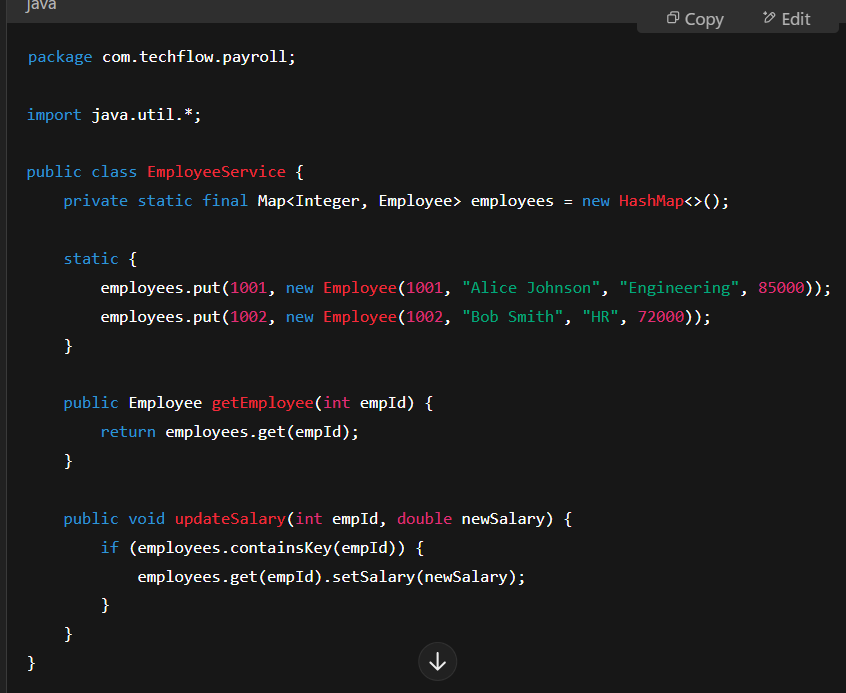
The codebase was not written to the highest possible standards, has inconsistencies and patchwork code to alleviate systemic problems and further still, it needs to be adapted and extended to accommodate new realities. The documentation is partial, lost, corrupted or unclear for end users and entirely non-existent for developers.

Your task is to analyze the code and produce a clear set of documents for users and developers.

# Codebase

## Employee Record Management

Document how the payroll system retrieves and updates employee data.

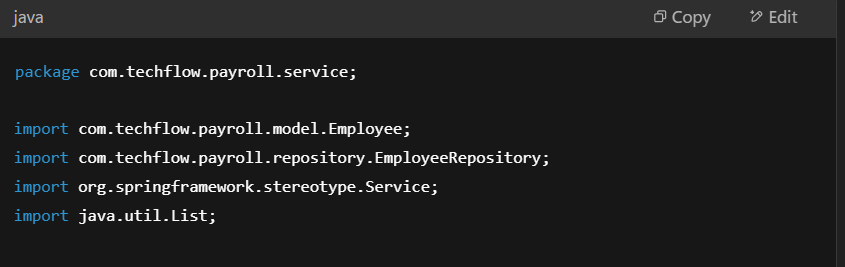


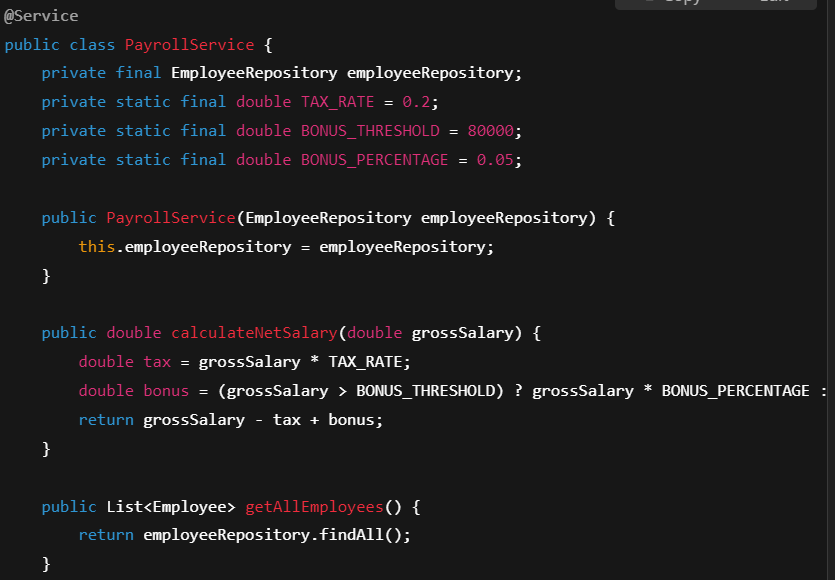
### Related Questions

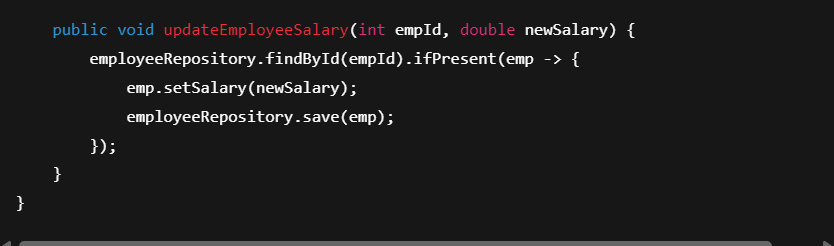
1. What does this codebase do?
2. How does salary updating work?
3. What happens if an invalid employee id is used?
4. There is a requirement to add new employees. How do we handle it? Hint: think about OOP architecture.

## Payroll Calculation

Document how the system calculates net salary after tax deductions.





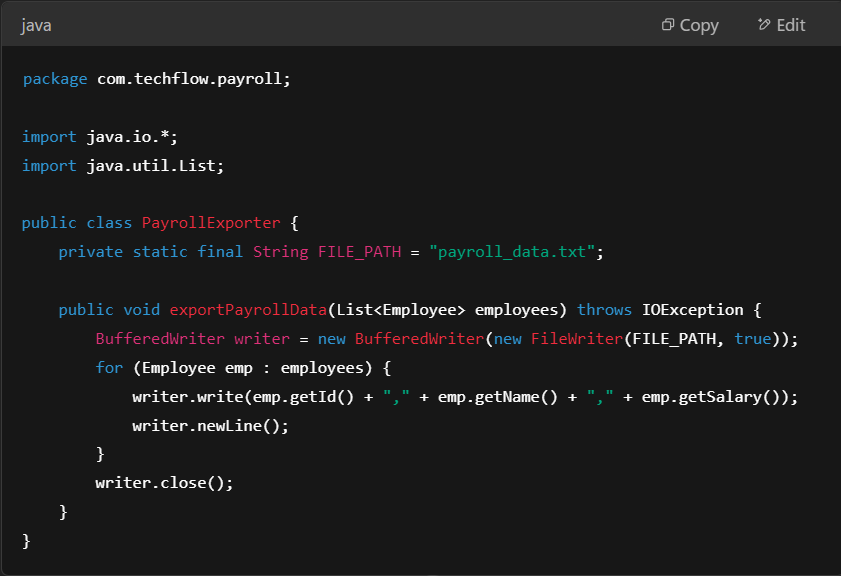


### Related Questions

1. How does the tax calculation work?
2. What happens if an employee earns above $80,000?
3. There is a need to extend this to include health insurance deductions.
4. There is another requirement that this mechanism must reflect a progressive tax structure.

## File-Based Payroll Data Export

Document how the system saves payroll data to a file for accounting purposes.

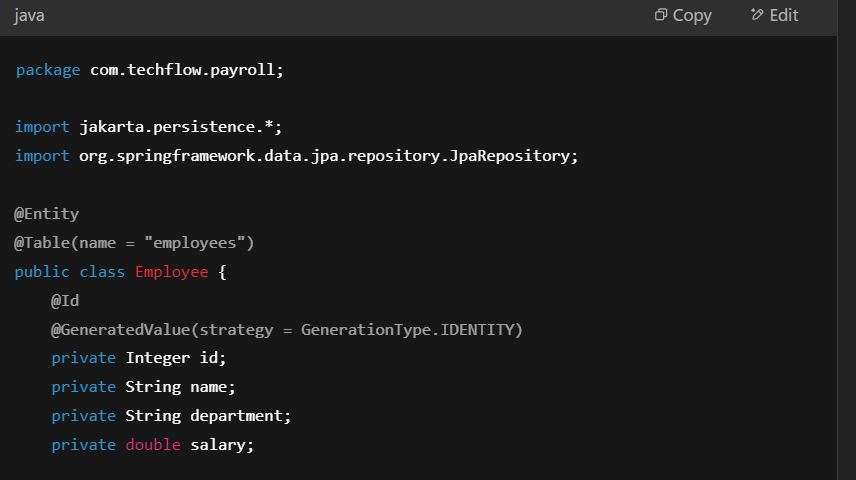


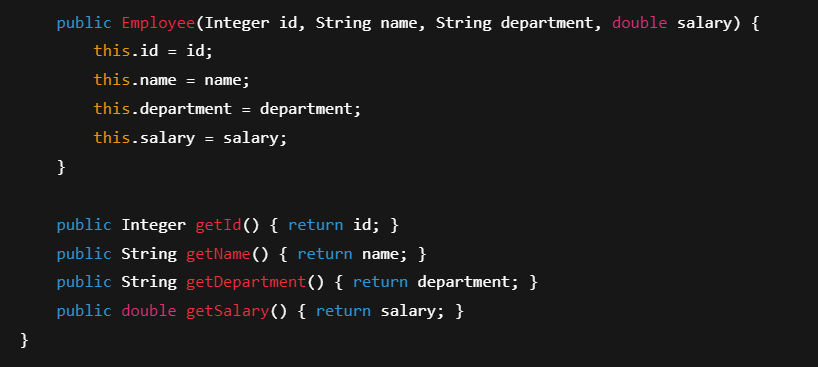
### Related Questions

1. Where is the payroll data stored?
2. Hat happens if a file is missing?
3. How would you modify this to encrypt payroll files?
4. There is a requirement to put this project on a cloud service. How would we need to extend it?

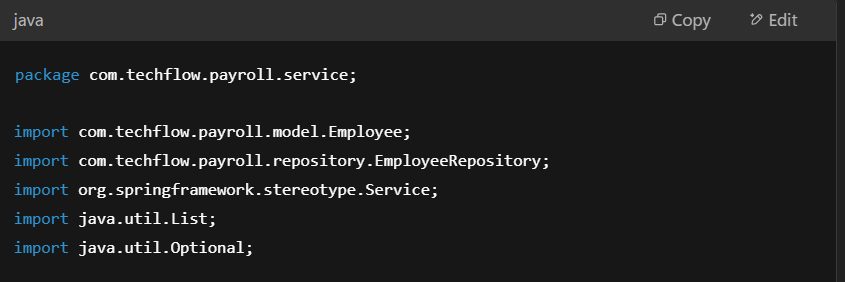
## Database Persistence

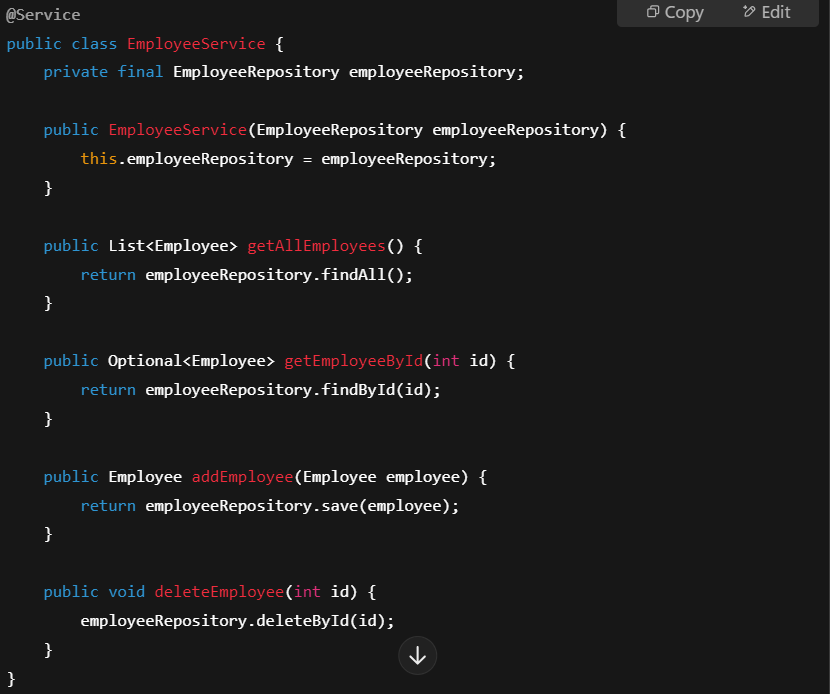
Document how employee data is saved to the database.



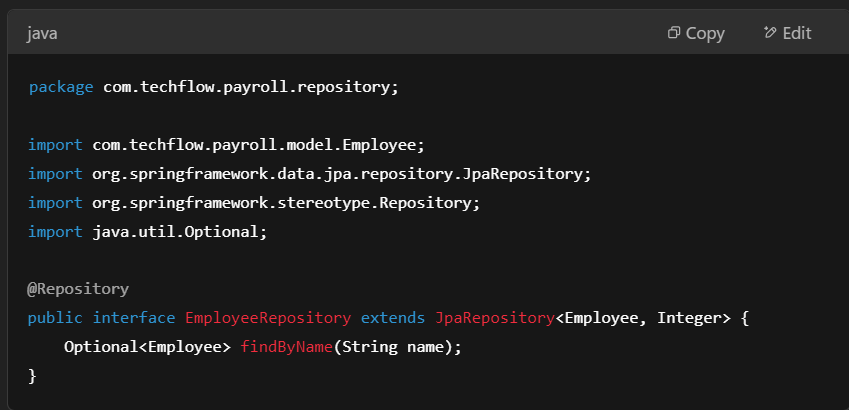


### Interim Service File





### Employee Repository Layer



## Unexpected Managerial Request

📩 **Subject:** Urgent Payroll System Enhancements for Q2  
📢 **From:** CTO - TechFlow Solutions  
📢 **To:** Development Team

Implementing Overtime Pay Calculation

We need the following requirements met:

### Overtime

1. Employees working more than 40 hours per week must receive 1.5x their hourly rate in overtime hours.
2. Overtime calculations should be included in the net salary calculations.
3. Example
   1. If an employee has a base salary of $1000 per week and works 5 extra hours, the formula should be:  
        
      Overtime Pay = extra hours \* (hourly rate \* 1.5)

### Encryption

1. The payroll data file must be encrypted before being saved to disk.
2. Use AES-256 encryption for security compliance.
3. A decryption function should allow authorized users to read the payroll file.

### Bank Account Information

1. Each employee must have a bank account stored securely.
2. Update the database schema and repository layer to include this field.
3. Ensure that the bank account number is properly validated.

### Payroll Summary Report

1. Create a new file export feature to generate a monthly report with the following columns: Employee ID | Employee Name | Net Salary | Overtime Pay | Bank Account
2. Allow filtering by department, such as the Finance Department.
3. The file should be saved in a CSV format.

### External System Access

1. The system must expose a REST API endpoint under “/api/payroll/process” that allows an external system to request payroll processing for all employees.
2. It should return the following:  
   