# OOP Exam – 13.07.2024

# Employee Management System

**Project Overview:**

This project involves developing a console-based application in Java, focusing on Object-Oriented Programming (OOP) principles. The application will serve as an Employee Management System, enabling users to manage employee data for a company.

**Key Objectives:**

* Implement OOP concepts like encapsulation, inheritance, polymorphism, and abstraction.
* Create a user-friendly console interface.
* Ensure data persistence by reading and writing employee data to/from a CSV file**.**

**Employee Data Management:**

* Add Employee: Input employee details (ID, name, department, role, salary) and add them to the system.
* Edit Employee: Modify existing employee details.
* Fire Employee: Flag that the employee doesn’t work at the company anymore.
* List Employees: Display a list of all active employees with their details.

**Data Persistence:**

* Read Data: Initialize the system with employee data from a CSV file.
* Write Data: Save current employee data to a CSV file upon exit.
* Search Functionality: Search for employees by name, ID, or department.

**Technical Requirements:**

* Employee and other relevant classes encapsulating respective properties and methods.
* Define interfaces for common functionalities.
* Use abstract classes where appropriate.

**File Handling:**

* Implement CSV file operations for data persistence (**DO NOT USE LIBRARIES FOR READING CSV FILES**).

**Exception Handling:**

* Robust error and exception management for user inputs and file operations.

**Data Structures:**

* Use appropriate collections (like Lists, Maps) to manage and organize data.

**Documentation:**

* Comprehensive documentation of classes, methods, and logic flow.
* Source Code: Well-organized and commented Java source files.

**Evaluation Criteria:**

* Adherence to OOP principles.
* Code quality and organization.
* Functionality and correctness.
* Error handling and validations.
* Documentation and ease of use.

**Each criterion gives you 20% of total points.**

**Example Usage of the Application:**

| **import java.util.Scanner;**  **public class EmployeeManagementApp {**  **public static void main(String[] args) {**  **// implement fileReader/fileWriter to handle saving into csv/json**  **Service service = new EmployeeService(reader, writer);**  **Manager manager = new EmployeeManager(service);**  **System.out.println("Welcome to Employee Management System");**  **displayCommands();**  **boolean isRunning = true;**  **while (isRunning) {**  **manager.execute(command);**  **// Add Employee**  **// 1, Peter Peterson, IT, Junior Java Developer, 1400.50**  **// Add Employee**  **// 2, Ivan Ivanson, IT, Junior Front-End Developer, 1400.00**  **// Edit 1**  **// Peter Peterson, IT, Java Developer, 2500.00**  **// List employees**  **// Search Department Marketing**  **// Search Id 1**  **// Fire 1**  **// Search Name Ivan**  **// Save & Exit**  **}**  **}**  **}** |
| --- |
| **example csv**  **Id,Name,StartDate,EndDate,Department,Role,Salary**  **55,Joseph Josephson,1980-05-20,null,IT,Junior Developer,2000.15**  **...** |

**Good luck and have fun!**