***REPORT***

**1- Problem Description**

* Each item details include: ID, name, address, salary, allowance
* The text file, named employees.txt contains some initial item details in the following line-by-line format:
  + Line for the ID of employee
  + Line for the name of employee
  + Line for the address of employee
  + Line for the salary of employee
  + Line for the allowance of employee
* Write a Java program having a simple menu that allows users managing a list of items. Functions are supported:
  + Adding new employee
  + Update an employee
  + Remove an employee
  + Print list of employee
  + Print list of employee in range
  + Save Item
  + Quit

**More requirements**

Format of the code of employee: E000, in which 0s are digits

Name of item is uppercase.

The printed list have order of ascending ID.

**2- Analysis**

From the problem description, the following use-cases are identified:

System

User

**3- Design**

**Class Design**

From the main concepts in the problem, following classes are identified:

|  |  |
| --- | --- |
| Concept | Class |
| Employee (ID, name, address , salary, allowance) | Employee (ID, name, address , salary, allowance) |
| List of Employee | EmpList, a collection of employee |
| Menu | Menu, a collection of strings |
| Program | EmpManage |

**EmpManager: Vector<Employee>**

**Class details**

+ EmpManager()

+ addFile(String fName): void

+ saveFile(String fName) : void

+addEmp : void

+ remove(): void

+ find (String fID): int

+ update(): void

+print: void

+printRange : void

+ sort(): void

+ Text(): void

+ PrintText(): void

**Menu: Vector<String>**

+ addMenu(String s): void

+ getChoice(): int

**Employee**

- ID: String

- name: String

- address : String

- salary: double

-allowance : double

+Employee(String, String, String, double, double)

+ getters/ setters

+ output()

**Main**

+ main(String[]): void

( +: public member, -: private member)

**User Interface:**

Menu of the program will be seen as:

1. Add new item
2. Delete an item
3. Update an item
4. Print the list
5. Print list in range
6. Save to file
7. Quit

**4- Implementation**

1. ***Initial data of the program ( file employees.txt)***

Please explore the software structure

1. ***Software***

Please run the program

**5- Testing**

|  |  |
| --- | --- |
| Test | Result (passed/ not passed) |
| Format of the code of an item: E000 | Passed |
| Name of items are uppercase | Passed |
| List is printed as an ascending order of ID | Passed |
| List is printed as an ascending order of ID in the input range | Passed |