**FKPayRollDesign**

**Structure**:

There are three types of employees:

1. Hourly - Hourly Salary
2. Flat Salary - Monthly Salary
3. Commissioned - Monthly Salary + Commissions from Sales

The base class is Employee class. There are 3 child classes of it (HourlyEmp, SalriedEmp, CommEmp)

Employee class implements **‘pay’ interface** which contain 3 methods which are crucial methods for every employee which calculating payroll for a day

**isPayDay** 🡪 Asks each employee if today is the paydate of that employee

**calcPay** 🡪 then return the amount that should be credited. It includes all extra charges(Union charges if any, sales commission for commissioned employee, etc)

**unionDeduction** 🡪 Since any employee can be a part of union so it is included in the interface. Upon executing, a given amount is approved to be debited from the employee’s next salary.

There are also class dependent methods like **setTimeHour** for hourly employees, **setSales** for commissioned employees, etc. They serve as mediator between the functions described below and any employee object. They are the one which actually penetrate into employee and set the required fields. They are similar to getters are setters.

**What we can do and how it is done?**

*We can do any of the following functions on any day. Union Deduction is also implemented differently. It was supposed to be done automatically on a weekly basis, but I kept it to be entirely dependent on user when to run it.*

Add employee 🡪 Application prompts user to enter the type of employee and credentials of the same.

After that that entry is added into the database.

Delete Employee 🡪 Application takes the id from the user, checks if the employee with that id exists. If yes, then that entry is deleted from the database.

Post a time card 🡪 Inputs taken are id and hours. If the request is valid(i.e. there’s a hourly employee corresponding to that id) then hour for the day are added in hours field of that employee.

Post a sales receipt 🡪 Inputs taken ae id and sales amount. If the request is valid(i.e. there’s a commissioned employee corresponding to that id) then sales field of that employee is updated. Appropriate balance corresponding to that sale((commission rate/100 )\* sales) is also added at that time.

Post a Union Card 🡪 Union deduction amount is taken as input which is followed by an iteration on the the database. A deduction is completed from the account of employees which are part of union.

**Run the payroll for a day** 🡪 The most important!

This is completed through an Iteration o n the employee list in Database

1. Starts Iteration
2. Ask each employee if today is his/her payday (This function is a part of interface implemented by Employee class).
3. If today is the payday for any employee then the amount to be paid is retrieved from the employee itself through calcPay method(also oart of interface)
4. Payable employees are divided into 3 categories accrding to their chosen method of payment. An entry is made in one out of 3 list. The entry contains employee id and the amount to be paid.
5. Once the iteration is complete, The data in the 3 lists corresponding to each method of payment are written in a separate file using file functions of java.

* There are some other functions like View all employees and advance to next date without doing anything this day. These are Self explanatory
* For the testing purpose of the evaluator, Employee database is initialized with 6 Employee objects, 2 from reach hourly , salaried and commissioned classes.

Simplified UML Class Diagram(All relations are not showed so that it can be shown in image)

(Actual UML Class diagram is in the repo as a .ucls file )

