FK Payroll Design

Design Objectives and Approaches

The system is designed in 3 parts :-

1) UX :- This is the menu or input or user-interface block. It manages all the user interaction and sends approcrite messages/signals to the other blocks. Users interact only at this layer.

2)Data Base:- A database interactor singleton class manages all the interactions with the database by firing appropriate queries using the sqlQueryExecuter class.

3)The System :- It contain the Employee and other associated classes. It manages the main payroll calculation task of the system.

All these three layers are completely independent of each other and one can be replaced at any-point of time without affecting the other.

Design Choices and Your Preferences

While designing the Payment Scheme and Type of Payment, I had 2 choices weather to keep them as enums and write the required functions to perform separated by if else in employe only or to work with composite design pattern . I chose composite design pattern for payment scheme since it had complex credit calculation function and used enum for mode of payment sing it was just one line functions.

While designing the database I had a choice weather to keep the account details and address of the employee with the employee only or create separate tables and link using foreign key, since they were both optional. I decided to keep account details in a separate table and address in the same table, since I though most employee might prefer not to give account details and many low end workers might not even be having account.
In unions case, I created a separate table to keep all union details and then created links in yet another

separate table to prevent null values in the employee table.

I also created a Controller/Configuration class using a singleton design pattern. This class contain global configuration data like date of month for paying employee or day of month. So that if you want to start giving money on Mondays instead of fridays then it could be easily done.

The Controller class also contained the safe input taking methods.

The Ux design was also done using inheritance. I created an abstract menu class containing the general menu pattern and then most of the menu classes were derived from it and instantiated.

Extra effort is given to keep each layer independent of each or so that if database changes or ux changes from cli to gui , other layers will still work perfectly.

Assumptions

1. The payroll function will be called every day.
2. Every month the payment will be given on a particular prdecided date. The date can be changed from 1 to 28.
3. Union fees is same for all employees belonging to a union.
4. Union fees is amount per week.
5. Commission is in percent.