FKPayrollDesign

Design Objectives:

To get better Insight of:

1) ***Type Design***
2) ***Object Oriented Paradigm***
3) ***Focus on Behaviour Rather than on States***
4) ***Following basics 4 rules in OOPs in System Design***

Version Control:

Version 0.01 *: Add employee Functionality for Administration*

Database Used: employee

Classes: Main, Administration

Purpose:

Main Class: To provide Basic Choice if the User wants to Enter as an employee And providing required essentials of Admin Authentication.

Administration Class: Provide Choice to Enter an Employee based on Username in the employee database.

Version 0.02 *: Add and Delete employee Functionality for Administration*

Database Used: employee

Classes: Main, Administration

Purpose:

Main Class: To provide Basic Choice if the User wants to Enter as an employee And providing required essentials of Admin Authentication.

Administration Class: Provide Choice to Enter an Employee based on Username in the employee database or delete an employee based on id using different functions.

Version 0.03*: 1) Add and Delete employee Functionality for Administration for both Hourly and Monthly Employee extending Employee Interface*

2) Payment making facility for Administrator
3) Setting HourlyRate, PunchingDailyCard, ModeofPayment for Hourly Employee.
4) Setting MonthlySalary, ModeofPayment for MonthlyEmployee
5) Password Authentication for User Too

Database Used: HourlyEmployee, MonthlyEmployee, HourlyEmployeeDaily, AdminDatabase

Version 0.04:

1. *All features from Version 0.03*
2. *Commission Option enabled for HourlyEmployee to Add in their monthly salary by Punching in their commissions on Daily Basis*

Database Used: HourlyEmployee, MonthlyEmployee, HourlyEmployeeDaily, AdminDatabase, Commission Table

Version 0.05:

1. *All features from previous versions.*
2. *Employees have a field set up for LastPaymentType to check in the type of way which he can withdraw salary and enabling of changing payment type for Employees.*

Database Used: HourlyEmployee, MonthlyEmployee, HourlyEmployeeDaily, AdminDatabase, Commission Table

Version 0.06:

1. *All features from previous versions*
2. *Employees can now be a part of a Union. Union can enter as a diff type from the Main function and can levy various taxes based on employee individually. The cesses will be then removed from next salary of the employee.*

Database Used: HourlyEmployee, MonthlyEmployee, HourlyEmployeeDaily, AdminDatabase, Commission Table, membership

Final Version:

1. *Administrator privilege now allows the HR services for both the Monthly and Hourly Employee based on Present Date. If present day==Friday, Hourly is paid and Date=30, Monthly is paid. HR can make a decision to make a date till which payment can be made*

REMARKS ON DESIGN CHOICES:

1) *Database is stored separately depending on the choice of employee to ensure maximum storage taking in decision to make use of 3nf for all databases. Databases are strong which clear choices to use depending on circumstances.*
2) *Choice of varied classes is made in a way to ensure that One Class behave as a single unit with it's own functionality, open-close principle is applied in aspects that changes in the class required don't affect the whole other classes.*
3) *Variables are used at minimal and follow localization by being local to the functions, hence inaccessible outside.*
4) *Polymorphic nature is supported by implementing interfaces, and supplying in Database name rather than making different functions depending on Database Name.*
5) *Inheritance of the interfaces make viable use of Designing towards extension.*
6) *Abstraction is strictly followed both by usage of abstract type interfaces and making only functions public within the classes.*

Databases used in Final Project:

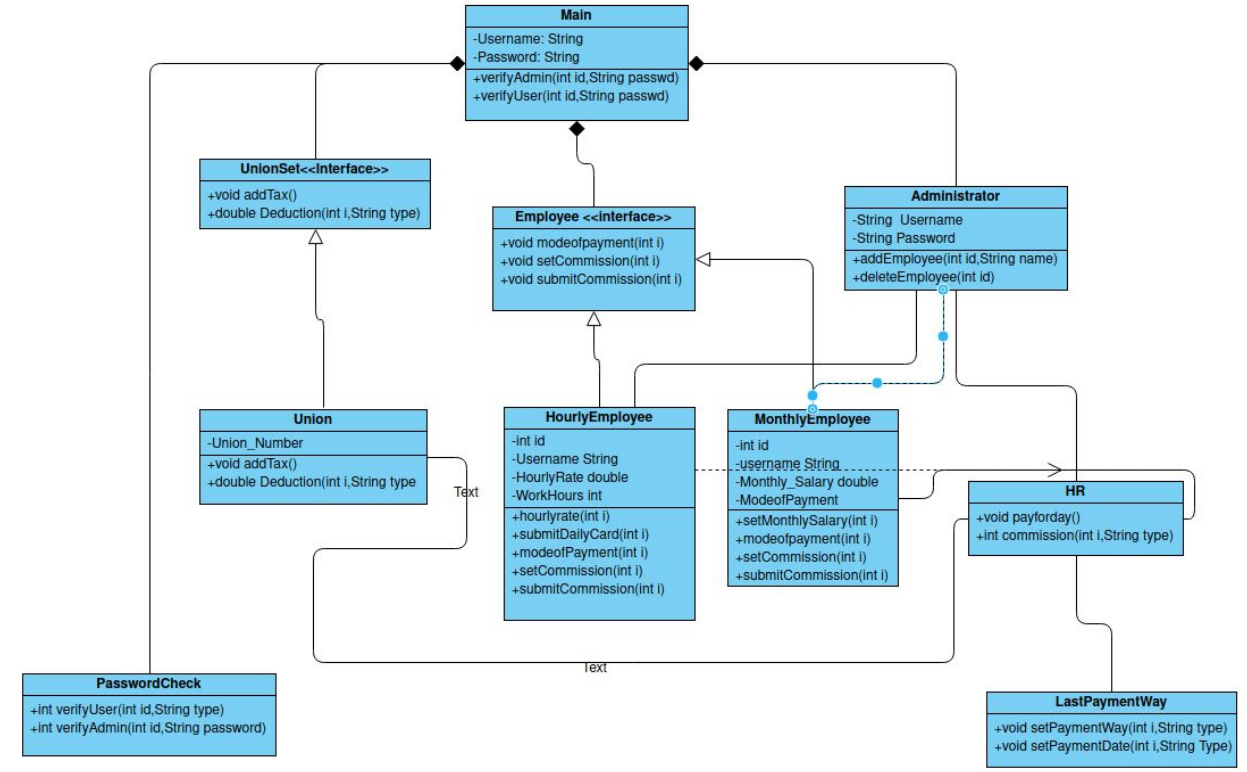
1) *HourlyEmployee*
2) *MonthlyEmployee*
3) *HourlyEmployeeDaily*
4) *AdminDatabase*
5) *Commission Table*
6) *membership*

Assumptions in Final Project:

1) *User Authentication through Databases*
2) *Admin Authentication through a separate AdminDatabase*

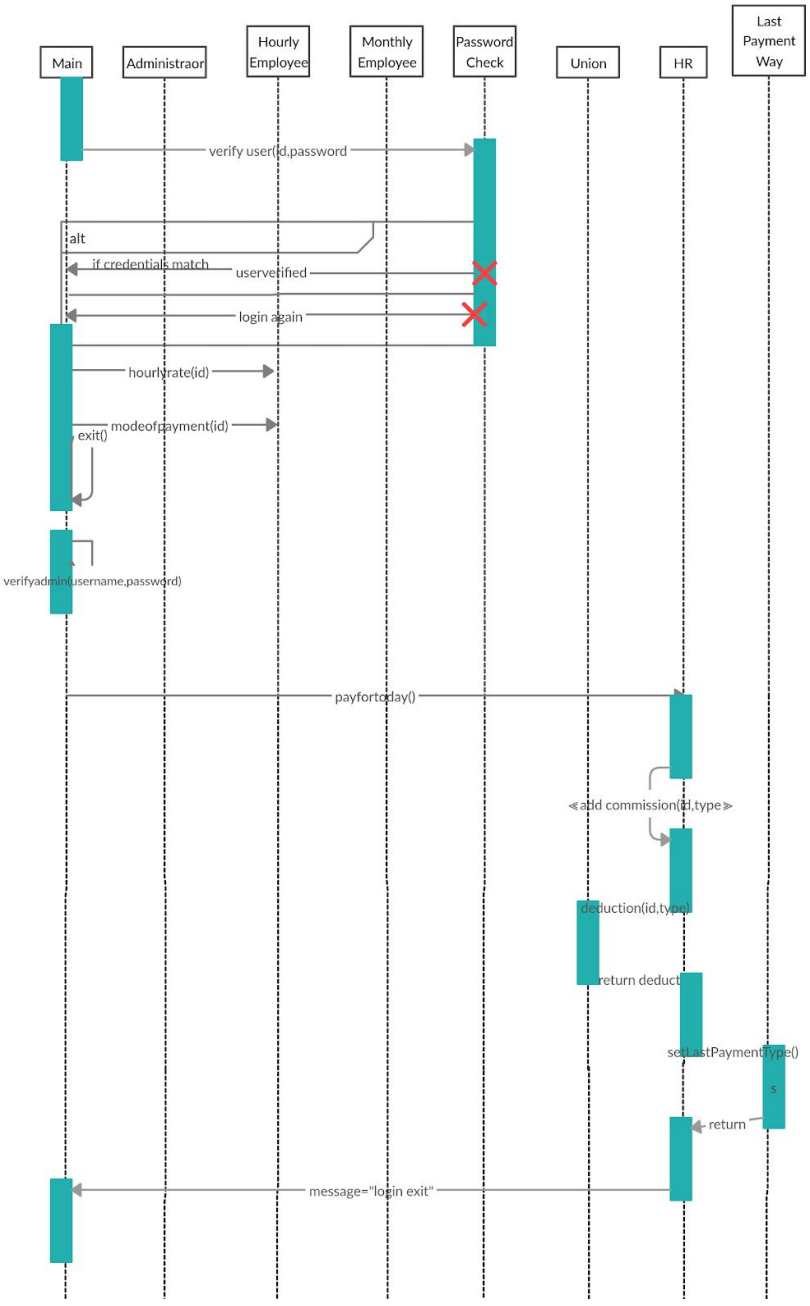
Major Usage of: JDBC, Date libraries in SQL and Java

Class Diagram:



The following Class Diagram is as Abstract Type and shows Major Dependencies.

Sequence Diagram:



Description: An employee logins, his verifying credentials are send to Password Check, where password is checked, upon successful return, can update his hourly rate , mode of payment, etc. He then logout of system. The admin enters, with his User Id,Password which upon being succesful, accredits salary of employees.

Remarks: Could not update properly because of absence of knowledge of few tools.