

LAB

Tuesday 7/22/2008

Now we use inheritance to perform payroll calculation based on the type of an employee. Consider the following problem:

*A company pays its employees on a weekly basis. The company has four types of employees: salaried employees, who are paid a fixed weekly salary regardless of the number of hours worked; hour employees, who are paid by the hour (for example, $wage * hours$) and receive overtime pay (for example, $(hours - 40) * wage * 1.5$) and the maximum hours worked is 168 hours; commission employees, who are paid a percentage of their sales; and salaried-commission employees, who receive a base salary plus a percentage of their sales. The company wants to implement a Java application that performs its payroll calculations.*

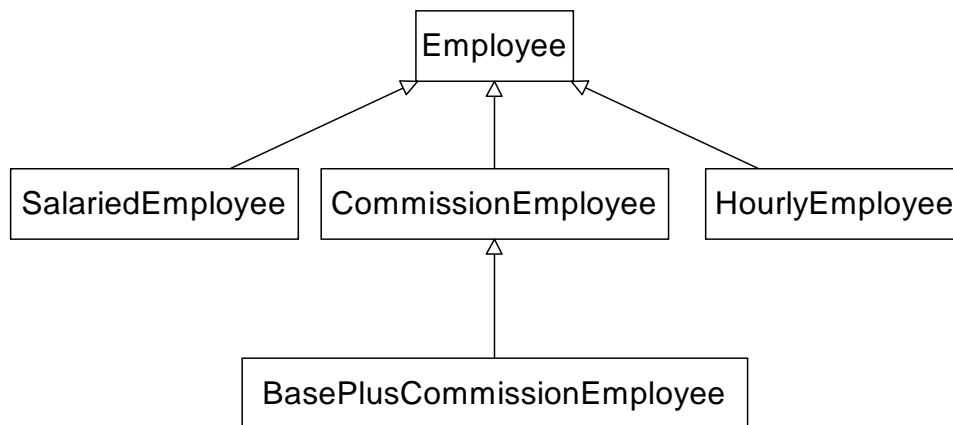


Figure 1. Class hierarchy for the polymorphic employee-payroll application.

Now, before implementing the program, expand Figure 1 to make a complete UML class diagram that includes all necessary variables and methods based on the description above.

Note that the Employee class is described with a first name, a last name and a social security number and their associated methods. For security, all data members for the Employee class are declared as private.

Once the UML class diagram is done, implement the code.

Below is the driver class, called `PayrollSystem`.

```
public class PayrollSystem{
    public static void main (String[] args){
        SalariedEmployee e1 = new SalariedEmployee("George", "Bush", "111-22-
3333",100000);
        System.out.println (e1.toString());

        HourlyEmployee e2 = new HourlyEmployee("Bill", "Clinton", "222-33-
4444", 16.75, 40);
        System.out.println (e2.toString());

        CommissionEmployee e3 = new CommissionEmployee("John", "MaCain", "333-
44-5555", 10000.0, 0.06);
        System.out.println (e3.toString());

        BasePlusCommissionEmployee e4 = new
BasePlusCommissionEmployee("Barack", "Obama", "444-55-6666", 5000.0, 0.04,
300);
        System.out.println (e4.toString());
    }
}
```