

**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.*

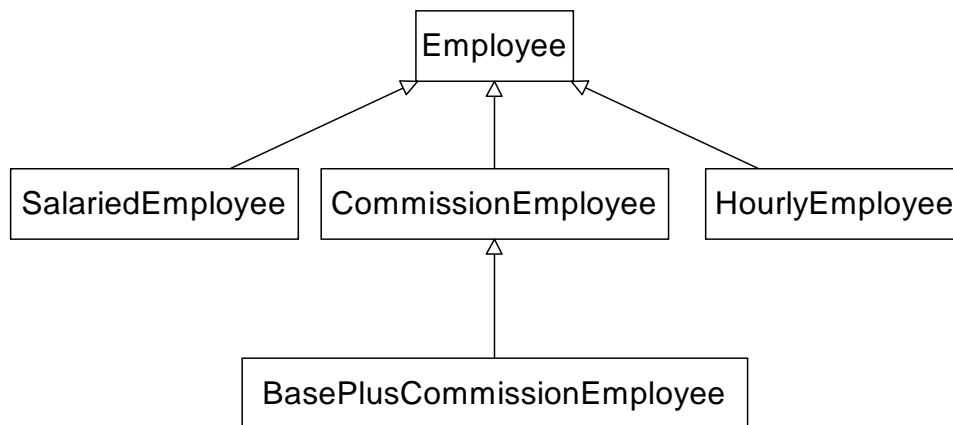


Figure 1. Class hierarchy for the polymorphic employee-payroll application.

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());
    }
}
```



```
public class Employee {
    private String firstName;
    private String lastName;
    private String SSN;

    public Employee (String first, String last, String ssn) {
        firstName = first;
        lastName = last;
        SSN = ssn;
    }

    public String getFirstName(){
        return firstName;
    }

    public void setFirstName (String first) {
        firstName = first;
    }

    public String getLastName(){
        return lastName;
    }

    public void setLastName (String last) {
        lastName = last;
    }

    public String getSSN(){
        return SSN;
    }

    public void setSSN(String number){
        SSN = number;
    }

    public String toString(){
        return getFirstName() + " " + getLastName() + "\nSSN: " + getSSN();
    }
}
```

```
public class SalariedEmployee extends Employee {
    private double weeklySalary;

    public SalariedEmployee (String first, String last, String ssn, double
salary){
        super (first, last, ssn);
        setWeeklySalary(salary);
    }

    public double getWeeklySalary(){
        return weeklySalary;
    }

    public void setWeeklySalary(double salary){
        weeklySalary = salary < 0.0 ? 0.0 : salary;
    }

    public double earnings(){
        return weeklySalary;
    }

    public String toString(){
        return "\nsalaried employee: " + super.toString() + "\nearned: $" +
earnings();
    }
}
```

```
public class HourlyEmployee extends Employee {
    private double wage;
    private double hours;

    public HourlyEmployee (String first, String last, String SSN, double
hourlyWage, double hoursWorked) {
        super (first, last, SSN);
        setWage (hourlyWage);
        setHours (hoursWorked);
    }

    public double getWage(){
        return wage;
    }

    public void setWage (double wageAmount) {
        wage = wageAmount < 0.0 ? 0.0 : wageAmount;
    }

    public double getHours(){
        return hours;
    }

    public void setHours (double hoursWorked){
        hours = (hoursWorked >= 0.0 && hoursWorked <= 168.0) ? hoursWorked :
0.0;
    }

    public double earnings(){
        if (hours <= 40)
            return wage * hours;
        else
            return 40 * wage + (hours - 40) * wage * 1.5;
    }

    public String toString(){
        return "\nhourly employee: " + super.toString() + "\nearned: $ " +
earnings();
    }
}
```

```
public class CommissionEmployee extends Employee{
    private double grossSales;
    private double commissionRate;

    public CommissionEmployee (String first, String last, String SSN, double
grossWeeklySales, double percent){
        super (first, last, SSN);
        setGrossSales (grossWeeklySales);
        setCommissionRate (percent);
    }

    public double getGrossSales(){
        return grossSales;
    }

    public void setGrossSales (double sales){
        grossSales = sales < 0.0 ? 0.0 : sales;
    }

    public double getCommissionRate(){
        return commissionRate;
    }

    public void setCommissionRate (double rate){
        commissionRate = (rate > 0.0 && rate < 1.0) ? rate : 0.0;
    }

    public double earnings(){
        return getCommissionRate() * getGrossSales();
    }

    public String toString(){
        return "\ncommission empolyee: " + super.toString() + "\nearned: $ "
+ earnings();
    }
}
```

```
public class BasePlusCommissionEmployee extends CommissionEmployee {
    private double baseSalary;

    public BasePlusCommissionEmployee (String first, String last, String SSN,
        double grossSalesAmount, double rate, double baseSalaryAmount){
        super (first, last, SSN, grossSalesAmount, rate);
        setBaseSalary (baseSalaryAmount);
    }

    public double getBaseSalary(){
        return baseSalary;
    }

    public void setBaseSalary (double salary){
        baseSalary = salary < 0.0 ? 0.0 : salary;
    }

    public double earnings(){
        return getBaseSalary() + super.earnings();
    }

    public String toString(){
        return "\nbase-salaried commission employee: " + super.getFirstName()
+
        " " + super.getLastName() + "\nsocial security number: " +
super.getSSN() + "\nearned: $" + earnings();
    }
}
```