

Assignment 2 - Employee Hierarchy

In Chapter 9, we studied an inheritance hierarchy in which class `BasePlusCommissionEmployee` inherited from `CommissionEmployee`. However, not all employees are `CommissionedEmployees`. For this assignment, create a more general `Employee` class that factors out the attributes and behavior that are common to all employees. Also, you will add two new types of employees, a `SalariedEmployee`, who gets paid a fixed salary, and an `HourlyEmployee` who gets paid an hourly wage plus time-and-a-half (1.5 times their hourly wage for hours worked over 40 hours). Create the proper `Employee` class hierarchy to reflect good object-oriented design.

The `SalariedEmployee` will need the additional instance variable of salary, and the `HourlyEmployee` will need the additional instance variables of `hourlyWage` and `hoursWorked`. Add setters and getters for these classes as appropriate. These classes will also need the `earnings` method and an override for the `toString()` method. The `SalariedEmployee` class will need a `setSalary` method to set the current salary for the employee. The `HourlyEmployee` class will need set methods for the `hourlyWage` and `hoursWorked` variables also. Salary and `hourlyWage` should be checked to make sure they are greater than zero and `hoursWorked` should be checked to ensure that it is between 0 and 168 (the number of hours in a week).

Your code in the subclasses should call methods in the super classes whenever possible to reduce the amount of code in the subclasses and utilize the code already developed in the super classes as in the code demonstrated in Figures 9.10 and 9.11 in the book.

Use the following code in your main function to test your classes:

```
CommissionEmployee employee1 = new CommissionEmployee("Fred", "Jones", "111-11-1111", 2000.0, .05);
BasePlusCommissionEmployee employee2 = new BasePlusCommissionEmployee("Sue", "Smith", "222-22-2222", 3000.0, .05, 300);
SalariedEmployee employee3 = new SalariedEmployee("Sha", "Yang", "333-33-3333", 1150.0);
HourlyEmployee employee4 = new HourlyEmployee("Ian", "Tanning", "444-44-4444", 15.0, 50);
HourlyEmployee employee5 = new HourlyEmployee("Angela", "Domchek", "555-55-5555", 20.0, 40);
System.out.printf("%s%s%s%s%s", employee1, employee2, employee3, employee4, employee5);
```

The output from your program should look like the following:

run:

```
Commissioned Employee: Fred Jones with ssn: 111-11-1111
  Gross Sales: 2000.00
  Commission Rate: 0.05
  Earnings: $100.00
Base Salary Plus Commissioned Employee: Sue Smith with ssn: 222-22-2222
  Gross Sales: 3000.00
  Commission Rate: 0.05
  with Base Salary of: $300.00
  Earnings: $450.00
Salaried Employee: Sha Yang with ssn: 333-33-3333
  Salary: 1150.00
  Earnings: $1150.00
Hourly Employee: Ian Tanning with ssn: 444-44-4444
  Hourly Wage: 15.00
  Hours Worked: 50.00
  Earnings: $825.00
Hourly Employee: Angela Domchek with ssn: 555-55-5555
  Hourly Wage: 20.00
  Hours Worked: 40.00
  Earnings: $800.00
```