Simple Payroll Report

Last Updated: March 2, 2016

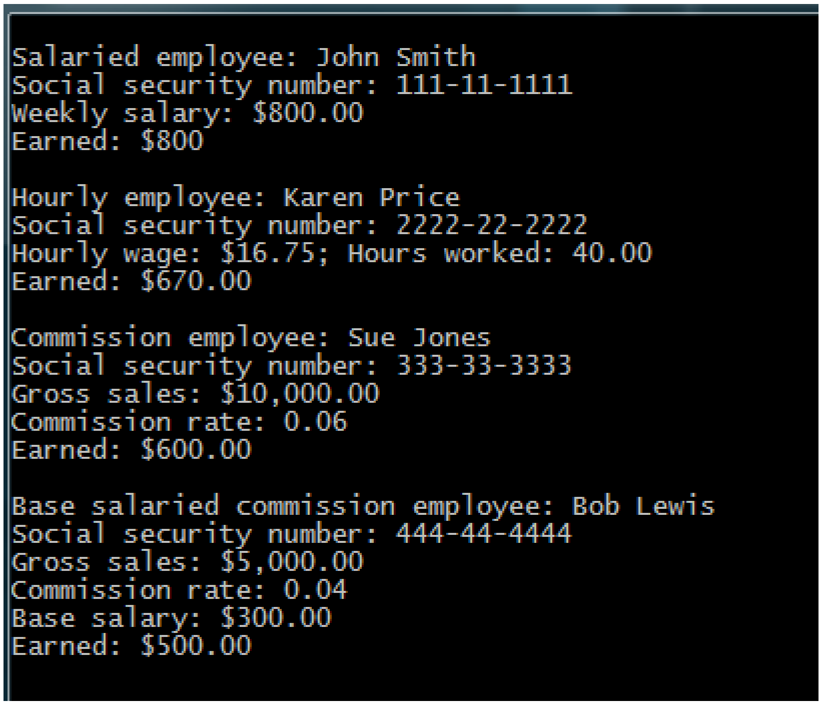
Question

Your product manager wants you to write an application that performs payroll calculations for various employee types polymorphically.

According to the biz req document, the employees are getting paid on a weekly basis. There are four types of employees: **Salaried employees** are paid a fixed weekly salary regardless of the number of hours worked. **Hourly employees** are paid by the hour and receive "time-and-a-half" overtime pay for all hours worked in excess of 40 hours. **Commission employees** are paid a percentage of their sales, and finally **salary based commission employees** receive a base salary plus a percentage of their sales.

Sample Output

- Please make sure to get all the necessary input from the user and make sure you are handling the inputs properly and gracefully – i.e. You should not let the app crash if garbage input is passed in
- Also please make sure that you are getting exactly the same earnings result for each polymorphic types described and please make sure to use your encapsulation, inheritance, and polymorphism knowledge
- Please also provide a simple inheritance diagram to communicate your design
- Lastly, make sure that you **<u>provide all the unit tests</u>** that you could come up with



```
Salaried employee: John Smith
Social security number: 111-11-1111
Weekly salary: $800.00
Earned: $800

Hourly employee: Karen Price
Social security number: 2222-22-2222
Hourly wage: $16.75; Hours worked: 40.00
Earned: $670.00

Commission employee: Sue Jones
Social security number: 333-33-3333
Gross sales: $10,000.00
Commission rate: 0.06
Earned: $600.00

Base salaried commission employee: Bob Lewis
Social security number: 444-44-4444
Gross sales: $5,000.00
Commission rate: 0.04
Base salary: $300.00
Earned: $500.00
```

Note – Your print statements should exactly match. Please pay attention to the sample output window above for the text and format that is expected from your output.

Developer Notes

- You are not given any implementation and/or guidance for this assignment. Read the specs and check out the output and determine your implementation and test cases yourself
- Please pay attention to the deadlines. Your commit timestamp should not be later than the hard deadline
- Before committing your changes please make sure
  - Your solution files build and run fine
  - Please **provide unit tests** yourself and make sure your unit tests pass
  - Please provide your **UML design doc** that communicates your design at a high level
- Once you committed your changes, please test it one last time to make sure everything is ok
- Please send the github link to the location of your solution to me and to your TA

Deadlines

- For this assignment both deadline and hard-deadline is **3/16/2015 @ 4pm**. Unfortunately assignments sent over after this deadline will not receive any credit

***NOTE

This assignment will count as two assignments. Please make every effort to complete it.