CSCD 211
Lab 7

You are a programmer in the IT department of an important law firm. Your job is to create a program that will report gross salary amounts and other compensation.

There are three types of employees in your firm:
- Programmers
- Lawyers
- Accountants

Your computer-based solution will use inheritance to reflect the 'general-to-specific' nature of your employee hierarchy.

Employee Class - Abstract
Data Members
- Private final double BASE
- Private String Name
- Protected double Salary

Methods - Required
- EVC that takes the name the basePayrate(sets the constant) and the additionalPayRate(sets the salaray BASE + additonalPayRate)
- getMethods
    - getSalary
    - getName
    - getBaseSalary
    - getType – this.getClass().Figure it out
- setMethods
    - setName
    - setSalary
- toString
- report - abstract
- compareTo – ensure an Employee is passed in – sorts by type – must call getType and then by salary – must call getSalary

Lawyer Class

Data Members
- Private int stockOptions

Methods - Required
- EVC that takes the name the basePayrate, the additionalPayRate, and the number of stock options
- getStockOptions
- setStockOptions – adds or subtracts from the current stockOptions
- report – see sample output for what should be reported
- toString – The name of the class: and then the base class toString

Programmer Class

Data Members
- Private boolean busPass

Methods - Required
- EVC that takes the name the basePayrate, the additionalPayRate, and true/false
- getBusPass
- setBusPass
- report – see sample output for what should be reported
- toString – The name of the class: and then the base class toString

Accountant Class

Data Members
- Private double parkingStipend

Methods - Required
- EVC that takes the name the basePayrate, the additionalPayRate, and the parkingStipend
- getParkingStipend
- setParkingStipend – adds or subtracts from the current parkingStipend
- report – see sample output for what should be reported
- toString – The name of the class: and then the base class toString

NOTES:
- I have provided CSCD211Lab7.java – you may not change this file
- I am providing a basic Javadoc – you must check all preconditions and throw the necessary exceptions

TO TURN IN
A zip file that contains:
- All Java files/folders needed to compile and run your program
- An output run named cscd211Lab7out.txt

Zip is named last name first letter of first name lab7.zip (Example: steinerslab7.zip)