CS260 Object-Oriented Prog & Design Dr. Chetan Jaiswal

Homework - Inheritance

Instructions:

1. Readable answers.
2. Your report must either be typed or clearly written. I will not grade unreadable homework.
3. Please do not email me your homework, rather upload scanned/typed copy on blackboard.
4. Show your work for maximum credit.

Total Points: 10

1. (2 points) Many programs written with inheritance could be written with composition instead, and vice versa. Rewrite class BasePlusCommissionEmployee of the CommissionEmployee–BasePlusCommissionEmployee hierarchy to use composition rather than inheritance. (The related programs are attached)

2. (2 points) Based on the Reptile class create another class as Lizard with private data members as int lizardLength, String lizardLocation. (lizardLocation values could be tropical, dry, cold etc.). Add a 4 argument constructor in this class to initialize values of brainSize, eggSize, lizardLength & lizardLocation. Override the toString method in this class and make it return all the 4 data members as String. Create a main method inside this class and create an object by initializing values for member data. Print this object using toString() method. Output shall look like:

brain size = 0.9

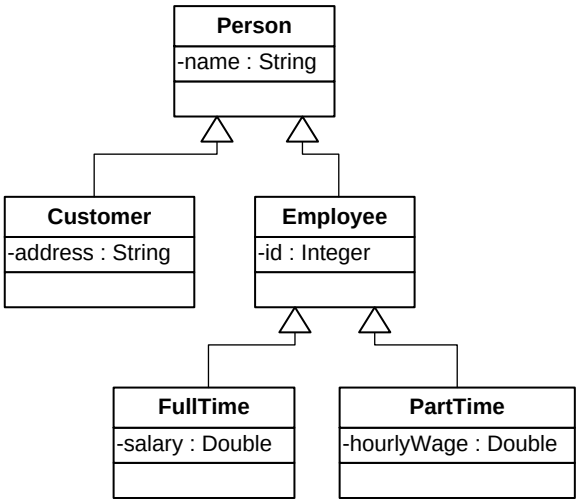
egg size = 4.0

lizardLength = 10

lizardLocation = tropical

3. (2 points) Create your own class “Base” (with 2 member data int num, String str) to demonstrate that it is not possible to create the object of an abstract class. Then extend this class into “Derived” class and provide the definition of the abstract methods from Base class. Now try to create the object of Derived and show that it is now possible. Take print screen to show the outputs.

4. (4 points) Implement:



Methods:

Person: one argument constructor, getName()

Customer: two argument constructor, toString(), getAddress()

Employee: two argument constructor, getID()

FullTime: three argument constructor, getSalary(), toString()

PartTime: three argument constructor, getHourlyWage(), toString()

Steps:

Create an array of Employee as Employee e [] = new Employee[2];

Initialize e[0] with the object of PartTime & e[1] with the object of FullTime.

Print both the objects using toString().

Create an array of Person as Person p [] = new Person[3];

Initialize each element with object of Customer, FullTime & PartTime.

Print all the objects using toString().