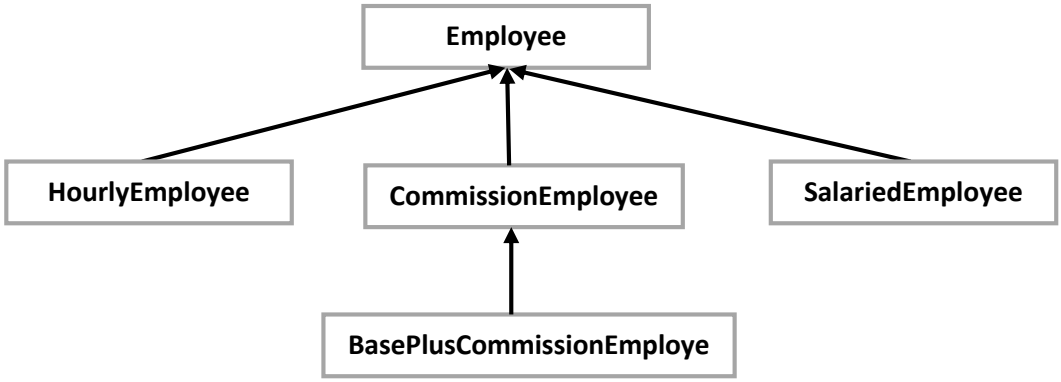
JAVA PRACTICAL_7

1. A company pays its employees on a weekly basis. The employees are of four types: Salaried employees are paid a fixed weekly salary regardless of the number of hours worked, hourly employees are paid by the hour and receive overtime pay (i.e., 1.5 times their hourly salary rate) for all hours worked in excess of 40 hours, commission employees are paid a percentage of their sales and base-salaried-commission-employees receive a base salary plus a percentage of their sales. For the current pay period, the company has decided to reward salaried-commission employees by adding 10% to their base salaries. The company wants you to write an application that performs its payroll calculations polymorphically. Use abstract class Employee to represent the general concept of an employee. The classes that extend Employee are SalariedEmployee, CommissionEmployee and Hourly-Employee. Class BasePlusCommissionEmployee which extends CommissionEmployee represents the last employee type. The figure below illustrated the hierarchy.



2. Implement the Shape hierarchy shown in Figure below. Each Two-DimensionalShape should contain method getArea to calculate the area of the two-dimensional shape. Each ThreeDimensionalShape should have methods getArea and getVolume to calculate the surface area and volume, respectively, of the three-dimensional shape. Create a program that uses an array of Shape references to objects of each concrete class in the hierarchy. The program should print a text description of the object to which each array element refers. Also, in the loop that processes all the shapes in the array, determine whether each shape is a TwoDimensionalShape or a ThreeDimensionalShape. If it's a TwoDimensionalShape, display its area. If it's a ThreeDimensionalShape, display its area and volume.