Implement the concept of Class, sub class(minimum 3),  Data members (minimum 7 ) , Methods ( minimum 2 )  , Access Specifier ( minimum 2 ) ,  Default Constructor, Method Overloading (minimum 1 method), Constructor overloading (minimum of 2) in **EMPLOYEE PAYROLL MANAGEMENT** domain.

class Employee {

private int empID;// 1

private String name;// 2

private String Dept;// 3

protected double basicSal;// 4

// default constructor

public Employee() {

empID = 0;

name = "";

Dept = "";

basicSal = 0.0;

}

// constructor overloading-1

public Employee(int empID, String name, String Dept, double basicSal) {

this.empID = empID;

this.name = name;

this.Dept = Dept;

this.basicSal = basicSal;

}

// constructor overloading-2

public Employee(String name, String Dept, double basicSal) {

this.empID = 0;

this.name = name;

this.Dept = Dept;

this.basicSal = basicSal;

}

// method-1

public void dispDetails() {

System.out.println("Employee ID: " + empID);

System.out.println("Name: " + name);

System.out.println("Dept: " + Dept);

System.out.println("Basic Salary: " + basicSal);

System.out.println("Total Salary: " + this.calculateSalary());

System.out.println();

}

// method-2

public double calculateSalary() {

return basicSal;// 5

}

}

// subclass-1

class Manager extends Employee {

private double bonus;// 6

public Manager(int empID, String name, String Dept, double basicSal, double bonus) {

super(empID, name, Dept, basicSal);

this.bonus = bonus;

}

// method overloading

public double calculateSalary() {

return basicSal + bonus;

}

}

// subclass-2

class Developer extends Employee {

private double overtimePay;// 7

public Developer(int empID, String name, String Dept, double basicSal, double overtimePay) {

super(empID, name, Dept, basicSal);

this.overtimePay = overtimePay;

}

// method overloading

public double calculateSalary() {

return basicSal + overtimePay;

}

}

// subclass-3

class partTime extends Employee {

public partTime(String name, String Dept, double basicSal) {

super(0, name, Dept, basicSal);

}

}

public class lab1 {

public static void main(String[] args) {

// creating an ordinary employee

Employee employee1 = new Employee(1, "employee1", "HR", 50000.0);

// creating a manager

Manager manager1 = new Manager(2, "manager", "Finance", 60000.0, 5000.0);

// creating a developer

Developer developer1 = new Developer(3, "developer", "IT", 55000.0, 10000.0);

partTime partTime1 = new partTime("PartTimer", "IT", 15000);

employee1.dispDetails();

manager1.dispDetails();

developer1.dispDetails();

partTime1.dispDetails();

}

}

