

DATE:29.12.23

DAY4- JAVA- QUIZ-2

1. Create a Java class named Calculator with two methods:

i) multiply method that takes two integers and returns their product.

ii) multiply method overload that takes three doubles and returns their product.

```
import java.util.Scanner;

class Calculator {
    int multiply(int a, int b) {
        return a * b;
    }

    double multiply(double a, double b, double c) {
        return a * b * c;
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter an integer for result1 calculation: ");
        int inputInt1 = scanner.nextInt();

        System.out.print("Enter another integer for result1 calculation: ");
        int inputInt2 = scanner.nextInt();

        System.out.print("Enter a double for result2 calculation: ");
        double inputDouble1 = scanner.nextDouble();

        System.out.print("Enter another double for result2 calculation: ");
        double inputDouble2 = scanner.nextDouble();

        System.out.print("Enter one more double for result2 calculation: ");
        double inputDouble3 = scanner.nextDouble();

        Calculator calculator = new Calculator();

        int result1 = calculator.multiply(inputInt1, inputInt2);
        double result2 = calculator.multiply(inputDouble1, inputDouble2, inputDouble3);

        System.out.println("Result 1: " + result1);
        System.out.println("Result 2: " + result2);
    }
}
```

2. Create a class hierarchy representing different types of employees in a company. Design a base class `Employee` with fields for the employee's name, employee ID, and a method named `calculateSalary` that returns the basic salary. Implement two subclasses: `Manager` and `Developer`.

```
class Employee {
    String name;
    int employeeID;

    Employee(String name, int employeeID) {
        this.name = name;
        this.employeeID = employeeID;
    }

    double calculateSalary() {
        return 50000;
    }
}

class Manager extends Employee {
    double bonusPercentage;

    Manager(String name, int employeeID, double bonusPercentage) {
        super(name, employeeID);
        this.bonusPercentage = bonusPercentage;
    }

    @Override
    double calculateSalary() {
        return super.calculateSalary() * (1 + bonusPercentage / 100);
    }
}

class Developer extends Employee {
    String programmingLanguage;

    Developer(String name, int employeeID, String programmingLanguage) {
        super(name, employeeID);
        this.programmingLanguage = programmingLanguage;
    }

    @Override
    double calculateSalary() {
        return super.calculateSalary() + 1000;
    }
}

public class Main {
    public static void main(String[] args) {
        Manager manager = new Manager("John Doe", 101, 10);
    }
}
```

```

        Developer developer = new Developer("Jane Smith", 102, "Java");

        System.out.println("Manager Salary: $" + manager.calculateSalary());
        System.out.println("Developer Salary: $" + developer.calculateSalary());
    }
}

```

3. Implement a class hierarchy with a base class Vehicle and two derived classes Car and Motorcycle

Write a program to create instances of car and motorcycle, call the calculateSpeed method on each and determine the vehicle with the highest effective speed

```

class Vehicle {
    int speed;

    Vehicle(int speed) {
        this.speed = speed;
    }

    int calculateSpeed() {
        return speed;
    }
}

class Car extends Vehicle {
    int numberOfPassengers;

    Car(int speed, int numberOfPassengers) {
        super(speed);
        this.numberOfPassengers = numberOfPassengers;
    }

    @Override
    int calculateSpeed() {
        return super.calculateSpeed() * numberOfPassengers;
    }
}

class Motorcycle extends Vehicle {
    int numberOfWheels;

    Motorcycle(int speed, int numberOfWheels) {
        super(speed);
        this.numberOfWheels = numberOfWheels;
    }

    @Override
    int calculateSpeed() {
        return super.calculateSpeed() * numberOfWheels;
    }
}

```

```
}
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        Car car = new Car(60, 4);
```

```
        Motorcycle motorcycle = new Motorcycle(80, 2);
```

```
        int carSpeed = car.calculateSpeed();
```

```
        int motorcycleSpeed = motorcycle.calculateSpeed();
```

```
        System.out.println("Car Speed: " + carSpeed);
```

```
        System.out.println("Motorcycle Speed: " + motorcycleSpeed);
```

```
        String fastestVehicle = (carSpeed > motorcycleSpeed) ? "Car" : "Motorcycle";
```

```
        System.out.println("The fastest vehicle is: " + fastestVehicle);
```

```
    }
```

```
}
```

