Program:1

```
public class Calculator {
   public int multiply(int a, int b) {
      return a * b;
   }
   public double multiply(double x, double y, double z) {
      return x * y * z;
   }
   public static void main(String[] args) {
      Calculator calculator = new Calculator();
      int result1 = calculator.multiply(5, 3);
      System.out.println("Multiplication result (integers): " + result1);
      double result2 = calculator.multiply(2.5, 3.0, 1.5);
      System.out.println("Multiplication result (doubles): " + result2);
   }
}
```

Program:2

```
class Employee {
   private String name;
   private int employeeID;
   private static final double BASIC_SALARY = 50000;
   public Employee(String name, int employeeID) {
      this.name = name;
      this.employeeID = employeeID;
   }
   public double calculateSalary() {
      return BASIC_SALARY;
   }
   public String toString() {
      return "Employee ID: " + employeeID + ", Name: " + name + ", Basic Salary: $" +
BASIC_SALARY;
```

```
    }
}
class Manager extends Employee {
    private double bonusPercentage;
    public Manager(String name, int employeeID, double bonusPercentage) {
        super(name, employeeID);
        this.bonusPercentage = bonusPercentage;
    }

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

    @Override
    public String toString() {
        return super.toString() + ", Bonus Percentage: " + bonusPercentage + "%";
    }
}

class Developer extends Employee {
    private String programmingLanguage;
    public Developer(String name, int employeeID, String programmingLanguage) {
        super(name, employeeID);
        this.programmingLanguage = programmingLanguage;
    }

    @Override
    public double calculateSalary() {
        return super.calculateSalary() + 10000;
    }

    @Override
    public String toString() {
        return super.toString() + ", Programming Language: " + programmingLanguage;
    }
}

public class Main {
    public static void main(String[] args) {
        Manager manager = new Manager("John Doe", 101, 15);
        Developer developer = new Developer("David", 102, "Java");
        System.out.println("Manager Details:");
        System.out.println(manager);
        System.out.println("Calculated Salary: $" + manager.calculateSalary());
        System.out.println();

        System.out.println("Developer Details:");
        System.out.println(developer);
        System.out.println("Calculated Salary: $" + developer.calculateSalary());
    }
}
```

Program:3

```
class Vehicle {
    protected double speed;
    public Vehicle(double speed) {
        this.speed = speed;
    }
    public double calculateSpeed() {
        return speed;
    }
}
class Car extends Vehicle {
    private int passengers;
    public Car(double speed, int passengers) {
        super(speed);
        this.passengers = passengers;
    }
    @Override
    public double calculateSpeed() {
        return super.calculateSpeed() * passengers;
    }
}
class Motorcycle extends Vehicle {
    private int wheels;
    public Motorcycle(double speed, int wheels) {
        super(speed);
        this.wheels = wheels;
    }
    @Override
    public double calculateSpeed() {
        return super.calculateSpeed() * wheels;
    }
}
public class Main {
    public static void main(String[] args) {
        Car car = new Car(60, 4);
        Motorcycle motorcycle = new Motorcycle(80, 2);
        System.out.println("Car Details:");
        System.out.println("Speed: " + car.calculateSpeed());
        System.out.println();
        System.out.println("Motorcycle Details:");
        System.out.println("Speed: " + motorcycle.calculateSpeed());
        System.out.println();

        if (car.calculateSpeed() > motorcycle.calculateSpeed()) {
            System.out.println("Car has the highest effective speed.");
        } else if (car.calculateSpeed() < motorcycle.calculateSpeed()) {
            System.out.println("Motorcycle has the highest effective speed.");
        } else {
            System.out.println("Both vehicles have the same effective speed.");
        }
    }
}
```