Problem 1: Create a Java class Calculator that provides different ways to perform addition. Include three methods: The first method takes two integer numbers, the second method takes three integer numbers, and the third method takes two double numbers. In the main method, create an object of Calculator class.

Calculator.java

```java
public class Calculator {
    public int add(int a, int b){
        return a+b;
    }
    public int add (int a, int b, int c){
        return a+b+c;
    }
    public double add(double a, double b){
        return a+b;
    }

    Run | Debug
    public static void main(String[] args){
        Calculator cal = new Calculator();
        System.out.println("Sum of 2 Integers: "+ cal.add(a:5, b:10));
        System.out.println("Sum of 3 Integers: "+ cal.add(a:15, b:15));
        System.out.println("Sum of 2 Doubles: "+ cal.add(a:5.5, b:11.5));
    }
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\@CODES\Academic\Java\Lab\Lab Report 6>  & 'C:\Program Files\Eclipse Adoptium\jdk-17.0.13
lsInExceptionMessages' '-cp' 'C:\Users\sobha\AppData\Roaming\Code\User\workspaceStorage\fdc97c
\Lab Report 6_177e2dd3\bin' 'Calculator'
Sum of 2 Integers: 15
Sum of 3 Integers: 30
Sum of 2 Doubles: 17.0
PS E:\@CODES\Academic\Java\Lab\Lab Report 6>
```

Problem 2: Create a Java class Shape that provides different ways to calculate the area. Include three methods: the first method takes one parameter (side length) to calculate the area of a square, the second method takes two parameters (length and width) to calculate the area of a rectangle, and the third method takes one decimal parameter (radius) to calculate the area of a circle. In the main method, create an object of Shape class.

Sape.java 1

E: > @CODES > Academic > Java > Lab > Lab Report 6 > Sape.java > Shape

```java
class Shape {
    public double area() {
        return 0;
    }
}
class Circle extends Shape {
    private double radius;

    public Circle(double radius) {
        this.radius = radius;
    }

    public double area(){
        return 3.1416 * radius * radius;
    }
}

class Rectangle extends Shape {
    private double length;
    private double width;

public Rectangle (double length, double width) {
    this.length = length;
    this.width = width;
}
    public double area() {
        return length * width;
    }
}

  public class Sape{
      Run | Debug
      public static void main(String[] args) {
          Shape circle = new Circle(radius:5);
          Shape rectangle = new Rectangle(length:4,width:6);

          System.out.println("Area of Circle: " +circle.area());
          System.out.println("Area of Rectangle: " +rectangle.area());
      }
  }

```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\sobha>  & 'C:\Program Files\Eclipse Adoptium\jdk-17.0.13.11-hotspot\bin\jav
Local\Temp\vscodesws_1640c\jdt_ws\jdt.ls-java-project\bin' 'Sape'
Area of Circle: 78.54
Area of Rectangle: 24.0
PS C:\Users\sobha>
```

Problem 3: Write a Java program to define a class Employee with instance variables name and id, along with a method calculateSalary(). Create two subclasses, Worker and Supervisor, each having additional instance variables baseSalary and bonus. In both subclasses, override the calculateSalary() method to compute and return the salary.

Company.java

```java
class Employee {
    protected String name;
    protected int id;

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

    public double calculateSalary() {
        return 0;
    }
}

class Worker extends Employee {
    private double baseSalary;
    private double bonus;

    public Worker(String name, int id, double baseSalary, double bonus) {
        super(name, id);
        this.baseSalary = baseSalary;
        this.bonus = bonus;
    }

    public double calculateSalary() {
        return baseSalary + bonus;
    }
}

class Supervisor extends Employee {
    private double baseSalary;
    private double bonus;
```

Company.java

```java
    public Supervisor(String name, int id, double baseSalary, double bonus) {
        super(name, id);
        this.baseSalary = baseSalary;
        this.bonus = bonus;
    }

    public double calculateSalary() {
        return baseSalary + bonus;
    }
}

public class Company {
    Run | Debug
    public static void main(String[] args) {
        Employee worker = new Worker(name:"Worker", id:1001, baseSalary:15000, bonus:5000);
        Employee supervisor = new Supervisor(name:"Supervisor", id:2001, baseSalary:30000, bonus:10000);

        System.out.println("Worker salary: " + worker.calculateSalary());
        System.out.println("Supervisor salary: " + supervisor.calculateSalary());
    }
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Run: Com

```
6_177e2dd3\bin' 'Company'
Worker salary: 20000.0
Supervisor salary: 40000.0
PS E:\@CODES\Academic\Java\Lab\Lab Report 6>
```

Problem 4: Write a Java program to define a class Vehicle with a method speedUp(). Create two subclasses: Car and Bicycle, each having an instance variable currentSpeed. In both subclasses, override the speedUp() method to increase the vehicle's speed differently.

Road.java

Road.java

```java
class Vehicle {
    public void speedUp() {
    }
}

class Car extends Vehicle {
    private int currentSpeed;

    public Car(int speed) {
        this.currentSpeed = speed;
    }

    public void speedUp() {
        currentSpeed += 20;
        System.out.println("Car speed: " + currentSpeed);
    }
}

class Bicycle extends Vehicle {
    private int currentSpeed;

    public Bicycle(int speed) {
        this.currentSpeed = speed;
    }

    public void speedUp() {
        currentSpeed += 5;
        System.out.println("Bicycle speed: " + currentSpeed);
    }
}
```

Road.java

```
    public void speedUp() {
        currentSpeed += 5;
        System.out.println("Bicycle speed: " + currentSpeed);
    }
}

public class Road {
    Run | Debug
    public static void main(String[] args) {
        Vehicle car = new Car(speed:60);
        Vehicle bicycle = new Bicycle(speed:10);

        car.speedUp();
        bicycle.speedUp();
    }
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\@CODES\Academic\Java\Lab\Lab Report 6> ^C
PS E:\@CODES\Academic\Java\Lab\Lab Report 6>
PS E:\@CODES\Academic\Java\Lab\Lab Report 6>  e:; cd 'e:\@CODES\Academic\Java\Lab\Lab Report 6
7.0.13.11-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\sobh
fdc97cdf1538aba08b1220769e47925d\redhat.java\jdt_ws\Lab Report 6_177e2dd3\bin' 'Road'
Car speed: 80
Bicycle speed: 15
PS E:\@CODES\Academic\Java\Lab\Lab Report 6>
```