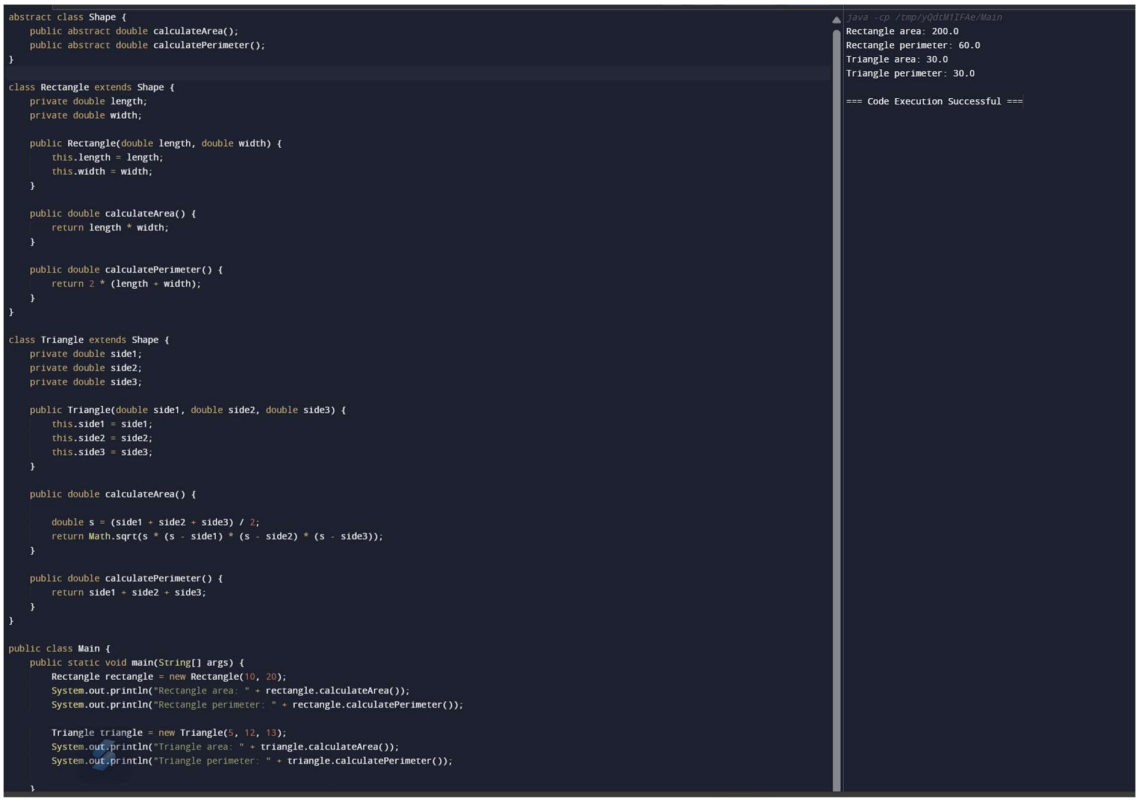
Test 4:-

SET 2

1. Create a base class called Shape with virtual functions area() and perimeter(). Derive two classes Rectangle and Triangle from the base class. Implement the area() and perimeter() functions for each class.



```
abstract class Shape {
    public abstract double calculateArea();
    public abstract double calculatePerimeter();
}

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

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

    public double calculateArea() {
        return length * width;
    }

    public double calculatePerimeter() {
        return 2 * (length + width);
    }
}

class Triangle extends Shape {
    private double side1;
    private double side2;
    private double side3;

    public Triangle(double side1, double side2, double side3) {
        this.side1 = side1;
        this.side2 = side2;
        this.side3 = side3;
    }

    public double calculateArea() {

        double s = (side1 + side2 + side3) / 2;
        return Math.sqrt(s * (s - side1) * (s - side2) * (s - side3));
    }

    public double calculatePerimeter() {
        return side1 + side2 + side3;
    }
}

public class Main {
    public static void main(String[] args) {
        Rectangle rectangle = new Rectangle(10, 20);
        System.out.println("Rectangle area: " + rectangle.calculateArea());
        System.out.println("Rectangle perimeter: " + rectangle.calculatePerimeter());

        Triangle triangle = new Triangle(5, 12, 13);
        System.out.println("Triangle area: " + triangle.calculateArea());
        System.out.println("Triangle perimeter: " + triangle.calculatePerimeter());

    }
```

```
/java --cp /tmp/yQdcWlIFAe/Main
Rectangle area: 200.0
Rectangle perimeter: 60.0
Triangle area: 30.0
Triangle perimeter: 30.0

=== Code Execution Successful ===
```

2. Create a base class called Animal with a virtual function move(). Derive two classes Bird and Fish from the base class. Implement the move() function for each class.

```java
abstract class Animal {
    abstract void move();
}

class Bird extends Animal {
    void move() {
        System.out.println("The bird flies in the sky.");
    }
}

class Fish extends Animal {
    void move() {
        System.out.println("The fish swims in the water.");
    }
}

public class Main {
    public static void main(String[] args) {
        Bird bird = new Bird();
        bird.move();

        Fish fish = new Fish();
        fish.move();
    }
}
```

```
java -cp /tmp/NusqfkyU4B/Main
The bird flies in the sky.
The fish swims in the water.

=== Code Execution Successful
```

3. Create a base class called Person with a virtual function greet(). Derive two classes Student and Teacher from the base class. Implement the greet() function for each class.

```java
abstract class person {
    abstract void greet();
}

class student extends person {
    void greet() {
        System.out.println("good morning teacher");
    }
}

class teacher extends person {
    void greet() {
        System.out.println("good morning students thank uou");
    }
}

public class Main {
    public static void main(String[] args) {
     student s = new student();
        s.greet();

        teacher t = new teacher();
        t.greet();
    }
}
```

```
java -cp /tmp/roldjaLiHV/Main
good morning teacher
good morning students thank uou

=== Code Execution Successful ===
```