Test-4:

1.Create an abstract Shape class with an abstract method area(). Implement subclasses Circle, Rectangle, and Triangle, each overriding the area() method to calculate the area specific to the shape. Additionally, provide overloaded constructors for each subclass to handle different input types for the shape dimensions.

Solution:

```
// Abstract Shape class
abstract class Shape {
// Abstract method for calculating area (to be implemented by subclasses)
public abstract double area();
}
class Circle extends Shape {
private double radius;
public Circle(double radius) {
this.radius = radius;
}
@Override
public double area() {
return Math.PI * radius * radius;
}
}
class Rectangle extends Shape {
private double length;
private double width;
public Rectangle(double length, double width) {
this.length = length;
```

```java
        this.width = width;
    }
    @Override
    public double area() {
        return length * width;
    }
}
class Triangle extends Shape {
    private double base;
    private double height;
    public Triangle(double base, double height) {
        this.base = base;
        this.height = height;
    }
    @Override
    public double area() {
        return 0.5 * base * height;
    }
}
public class Main {
    public static void main(String[] args) {
        Circle circle = new Circle(5.0);
        Rectangle rectangle = new Rectangle(4.0, 3.0);
        Triangle triangle = new Triangle(4.0, 3.0);
        System.out.println("Area of circle: " + circle.area());
        System.out.println("Area of rectangle: " + rectangle.area());
        System.out.println("Area of triangle: " + triangle.area());
    }

}
```

Output:

```
Area of circle: 78.53981633974483

Area of rectangle: 12.0

Area of triangle: 6.0
```