```java
import java.io.*;
import java.util.*;

abstract class Shape {
    protected int dim1, dim2;

    public Shape(int dim1, int dim2) {
        if (dim1 <= 0 || dim2 <= 0) {
            throw new IllegalArgumentException("Dimensions must be positive 
numbers!");
        }
        this.dim1 = dim1;
        this.dim2 = dim2;
    }

    public abstract double area();

    public final void displayInfo(PrintWriter log) {
        String msg = this.getClass().getSimpleName() + " Area: " + area();
        System.out.println(msg);
        log.println(msg);
    }
}

interface Colorable {
    String colorDescription();
}

class Rectangle extends Shape implements Colorable {
    private String color;
    public Rectangle(int dim1, int dim2, String color) {
        super(dim1, dim2);
        this.color = color;
    }
    public double area() { return dim1 * dim2; }
    public String colorDescription() { return "The Rectangle is colored " + color; 
}
}

class Triangle extends Shape implements Colorable {
    private String color;
    public Triangle(int dim1, int dim2, String color) {
        super(dim1, dim2);
        this.color = color;
    }
    public double area() { return 0.5 * dim1 * dim2; }
    public String colorDescription() { return "The Triangle is colored " + color; }
}

class Circle extends Shape implements Colorable {
    private String color;
    public Circle(int radius, String color) {
        super(radius, radius);
        this.color = color;
    }
    public double area() { return Math.PI * dim1 * dim1; }
    public String colorDescription() { return "The Circle is colored " + color; }
}

public class ShapeDemo {
    public static void main(String[] args) {
        try (Scanner sc = new Scanner(System.in);
             PrintWriter log = new PrintWriter(new FileWriter("ShapeDemo.log",
true))) {

            Random rand = new Random();
            String[] colors = {"Red", "Blue", "Green", "Yellow"};

            Shape[] shapes = new Shape[3];
            try {
                System.out.print("Enter Rectangle dimensions: ");
                int r1 = sc.nextInt(), r2 = sc.nextInt();
                log.println("Rectangle input: " + r1 + " " + r2);
                shapes[0] = new Rectangle(r1, r2,
colors[rand.nextInt(colors.length)]);

                System.out.print("Enter Triangle dimensions: ");
                int t1 = sc.nextInt(), t2 = sc.nextInt();
                log.println("Triangle input: " + t1 + " " + t2);
                shapes[1] = new Triangle(t1, t2,
colors[rand.nextInt(colors.length)]);

                System.out.print("Enter Circle radius: ");
                int c1 = sc.nextInt();
                log.println("Circle input: " + c1);
                shapes[2] = new Circle(c1, colors[rand.nextInt(colors.length)]);

                for (Shape s : shapes) {
                    s.displayInfo(log);
                    String colorMsg = ((Colorable) s).colorDescription();
                    System.out.println(colorMsg);
                    log.println(colorMsg);
                }
            } catch (IllegalArgumentException e) {
                System.out.println("Invalid input: " + e.getMessage());
                log.println("Invalid input: " + e.getMessage());
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```