

## Selected files

### 5 printable files

2A\Circle.java  
2A\Rectangle.java  
2A\Shape.java  
2A\Square.java  
2A\TestShape.java

#### 2A\Circle.java

```
1 public class Circle extends Shape {
2     // Instance variable
3     private double radius;
4
5     // Constructors
6     public Circle() {
7         this.radius = 1.0;
8     }
9
10    public Circle(double radius) {
11        this.radius = radius;
12    }
13
14    public Circle(double radius, String color, boolean filled) {
15        super(color, filled);
16        this.radius = radius;
17    }
18
19    // Getter and setter for radius
20    public double getRadius() {
21        return radius;
22    }
23
24    public void setRadius(double radius) {
25        this.radius = radius;
26    }
27
28    // Methods
29    public double getArea() {
30        return Math.PI * radius * radius;
31    }
32
33    public double getPerimeter() {
34        return 2 * Math.PI * radius;
35    }
36
37    // Override toString method
38    @Override
39    public String toString() {
40        return "Circle[" + super.toString() + ", radius=" + radius + "];"
41    }
42 }
```

#### 2A\Rectangle.java

```
1 public class Rectangle extends Shape {
2     // Instance variables
3     private double width;
```

```
4     private double length;
5
6     // Constructors
7     public Rectangle() {
8         this.width = 1.0;
9         this.length = 1.0;
10    }
11
12    public Rectangle(double width, double length) {
13        this.width = width;
14        this.length = length;
15    }
16
17    public Rectangle(double width, double length, String color, boolean filled) {
18        super(color, filled);
19        this.width = width;
20        this.length = length;
21    }
22
23    // Getter and setter for width and length
24    public double getWidth() {
25        return width;
26    }
27
28    public void setWidth(double width) {
29        this.width = width;
30    }
31
32    public double getLength() {
33        return length;
34    }
35
36    public void setLength(double length) {
37        this.length = length;
38    }
39
40    // Methods
41    public double getArea() {
42        return width * length;
43    }
44
45    public double getPerimeter() {
46        return 2 * (width + length);
47    }
48
49    // Override toString method
50    @Override
51    public String toString() {
52        return "Rectangle[" + super.toString() + ", width=" + width + ", length=" + length +
53        "]\n";
54    }
```

## 2A\Shape.java

```
1 public class Shape {
2     // Instance variables
3     private String color;
4     private boolean filled;
```

```
5
6 // Constructors
7 public Shape() {
8     this.color = "green";
9     this.filled = true;
10 }
11
12 public Shape(String color, boolean filled) {
13     this.color = color;
14     this.filled = filled;
15 }
16
17 // Getters and setters
18 public String getColor() {
19     return color;
20 }
21
22 public void setColor(String color) {
23     this.color = color;
24 }
25
26 public boolean isFilled() {
27     return filled;
28 }
29
30 public void setFilled(boolean filled) {
31     this.filled = filled;
32 }
33
34 // toString method
35 @Override
36 public String toString() {
37     return "Shape[color=" + color + ", filled=" + filled + "]";
38 }
39 }
```

## 2A\Square.java

```
1 public class Square extends Rectangle {
2     // Constructors
3     public Square() {
4         super();
5     }
6
7     public Square(double side) {
8         super(side, side);
9     }
10
11     public Square(double side, String color, boolean filled) {
12         super(side, side, color, filled);
13     }
14
15     // Override toString method
16     @Override
17     public String toString() {
18         return "Square[" + super.toString() + "]";
19     }
20
21     // Override setWidth and setLength to maintain square geometry
```

```
22     @Override
23     public void setWidth(double side) {
24         super.setWidth(side);
25         super.setLength(side);
26     }
27
28     @Override
29     public void setLength(double side) {
30         super.setWidth(side);
31         super.setLength(side);
32     }
33 }
```

## 2A\TestShape.java

```
1 public class TestShape {
2     public static void main(String[] args) {
3         // Test Shape class
4         Shape shape1 = new Shape();
5         System.out.println(shape1);
6
7         Shape shape2 = new Shape("blue", false);
8         System.out.println(shape2);
9
10        // Test Circle class
11        Circle circle1 = new Circle();
12        System.out.println(circle1);
13        System.out.println("Area: " + circle1.getArea());
14        System.out.println("Perimeter: " + circle1.getPerimeter());
15
16        Circle circle2 = new Circle(3.5, "red", true);
17        System.out.println(circle2);
18        System.out.println("Area: " + circle2.getArea());
19        System.out.println("Perimeter: " + circle2.getPerimeter());
20
21        // Test Rectangle class
22        Rectangle rectangle1 = new Rectangle();
23        System.out.println(rectangle1);
24        System.out.println("Area: " + rectangle1.getArea());
25        System.out.println("Perimeter: " + rectangle1.getPerimeter());
26
27        Rectangle rectangle2 = new Rectangle(2.5, 4.0, "yellow", false);
28        System.out.println(rectangle2);
29        System.out.println("Area: " + rectangle2.getArea());
30        System.out.println("Perimeter: " + rectangle2.getPerimeter());
31
32        // Test Square class
33        Square square1 = new Square();
34        System.out.println(square1);
35        System.out.println("Area: " + square1.getArea());
36        System.out.println("Perimeter: " + square1.getPerimeter());
37
38        Square square2 = new Square(5.0, "purple", true);
39        System.out.println(square2);
40        System.out.println("Area: " + square2.getArea());
41        System.out.println("Perimeter: " + square2.getPerimeter());
42    }
43 }
44
```