

```

// Use dynamic method dispatch.
class TwoDShape {
    private double width;
    private double height;
    private String name;

    // A default constructor.
    TwoDShape() {
        width = height = 0.0;
        name = "none";
    }

    // Parameterized constructor.
    TwoDShape(double w, double h, String n) {
        width = w;
        height = h;
        name = n;
    }

    // Construct object with equal width and height.
    TwoDShape(double x, String n) {
        width = height = x;
        name = n;
    }

    // Construct an object from an object.
    TwoDShape(TwoDShape ob) {
        width = ob.width;
        height = ob.height;
        name = ob.name;
    }

    // Accessor methods for width and height.
    double getWidth() { return width; }
    double getHeight() { return height; }
    void setWidth(double w) { width = w; }
    void setHeight(double h) { height = h; }

    String getName() { return name; }

    void showDim() {
        System.out.println("Width and height are " +
            width + " and " + height);
    }

    double area() {
        System.out.println("area() must be overridden");
        return 0.0;
    }
}

```

```
// A subclass of TwoDShape for triangles.
class Triangle extends TwoDShape {
    private String style;

    // A default constructor.
    Triangle() {
        super();
        style = "none";
    }

    // Constructor for Triangle.
    Triangle(String s, double w, double h) {
        super(w, h, "triangle");

        style = s;
    }

    // One argument constructor.
    Triangle(double x) {
        super(x, "triangle"); // call superclass constructor

        style = "filled";
    }

    // Construct an object from an object.
    Triangle(Triangle ob) {
        super(ob); // pass object to TwoDShape constructor
        style = ob.style;
    }

    // Override area() for Triangle.
    double area() {
        return getWidth() * getHeight() / 2;
    }

    void showStyle() {
        System.out.println("Triangle is " + style);
    }
}
```

```
// A subclass of TwoDShape for rectangles.
class Rectangle extends TwoDShape {
    // A default constructor.
    Rectangle() {
        super();
    }

    // Constructor for Rectangle.
    Rectangle(double w, double h) {
        super(w, h, "rectangle"); // call superclass constructor
    }

    // Construct a square.
    Rectangle(double x) {
        super(x, "rectangle"); // call superclass constructor
    }

    // Construct an object from an object.
    Rectangle(Rectangle ob) {
        super(ob); // pass object to TwoDShape constructor
    }

    boolean isSquare() {
        if(getWidth() == getHeight()) return true;
        return false;
    }

    // Override area() for Rectangle.
    double area() {
        return getWidth() * getHeight();
    }
}
```

```
class DynShapes {
    public static void main(String args[]) {
        TwoDShape shapes[] = new TwoDShape[5];

        shapes[0] = new Triangle("outlined", 8.0, 12.0);
        shapes[1] = new Rectangle(10);
        shapes[2] = new Rectangle(10, 4);
        shapes[3] = new Triangle(7.0);
        shapes[4] = new TwoDShape(10, 20, "generic");

        for(int i=0; i < shapes.length; i++) {
            System.out.println("object is " + shapes[i].getName());
            System.out.println("Area is " + shapes[i].area());

            System.out.println();
        }
    }
}
```