

Task 2: Factory Method

Create a ShapeFactory class that encapsulates the object creation logic of different Shape objects like Circle, Square, and Rectangle.

```
package com.Day28;
public class ShapeFactoryDemo {
    interface Shape {
        void draw();
    }
    static class Circle implements Shape {
        @Override
        public void draw() {
            System.out.println("Drawing a Circle");
        }
    }
    static class Square implements Shape {
        @Override
        public void draw() {
            System.out.println("Drawing a Square");
        }
    }
    static class Rectangle implements Shape {
        @Override
        public void draw() {
            System.out.println("Drawing a Rectangle");
        }
    }
    static class ShapeFactory {
        Shape getShape(String shapeType) {
            if (shapeType == null) {
                return null;
            }
            if (shapeType.equalsIgnoreCase("CIRCLE")) {
                return new Circle();
            } else if (shapeType.equalsIgnoreCase("SQUARE")) {
                return new Square();
            } else if (shapeType.equalsIgnoreCase("RECTANGLE")) {
                return new Rectangle();
            }
            return null;
        }
    }
    public static void main(String[] args) {
        ShapeFactory shapeFactory = new ShapeFactory();
        Shape shape1 = shapeFactory.getShape("CIRCLE");
        shape1.draw(); // Output: Drawing a Circle
        Shape shape2 = shapeFactory.getShape("SQUARE");
        shape2.draw(); // Output: Drawing a Square
        Shape shape3 = shapeFactory.getShape("RECTANGLE");
        shape3.draw(); // Output: Drawing a Rectangle
    }
}
```