Dado los siguientes pedazos de código implementa un diagrama representante de este uso del Factory Method

```java
public interface Shape {
    void draw();
}

public class Rectangle implements Shape {

    @Override
    public void draw() {
        System.out.println("Es un rectángulo");
    }
}

public class Square implements Shape {

    @Override
    public void draw() {
        System.out.println("Es un cuadrado");
    }
}

public class Circle implements Shape {

    @Override
    public void draw() {
        System.out.println("Es un círculo");
    }
}

public class ShapeFactory {

    public Shape getShape(String shapeType){
        if(shapeType == null){
            return null;
        }
        if(shapeType.equalsIgnoreCase("CIRCULO")){
            return new Circle();

        } else if(shapeType.equalsIgnoreCase("RECTANGULO")){
```

```java
            return new Rectangle();

        } else if(shapeType.equalsIgnoreCase("CUADRADO")){
            return new Square();
        }

        return null;
    }
}

public class FactoryPatternDemo {

    public static void main(String[] args) {
        ShapeFactory shapeFactory = new ShapeFactory();

        Shape shape1 = shapeFactory.getShape("CIRCULO");

        shape1.draw();

        Shape shape2 = shapeFactory.getShape("RECTANGULO");

        shape2.draw();

        Shape shape3 = shapeFactory.getShape("CUADRADO");

        shape3.draw();
    }
}
```