**Question : Implementing the Factory Method Pattern**

**CODE:**

package dnsolutionsproblem1;

interface Shape {

void draw();

}

class Circle implements Shape {

public void draw() {

System.*out*.println("This will return Circle");

}

}

class Rectangle implements Shape {

public void draw() {

System.*out*.println("This will return Rectangle");

}

}

class Square implements Shape {

public void draw() {

System.*out*.println("This will return Square");

}

}

class ShapeFactory {

public Shape getShape(String shapeType) {

if (shapeType == null) {

return null;

}

if (shapeType.equalsIgnoreCase("CIRCLE")) {

return new Circle();

} else if (shapeType.equalsIgnoreCase("RECTANGLE")) {

return new Rectangle();

} else if (shapeType.equalsIgnoreCase("SQUARE")) {

return new Square();

}

return null;

}

}

public class FactoryPatternDemo {

public static void main(String[] args) {

ShapeFactory shapeFactory = new ShapeFactory();

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

shape1.draw();

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

shape2.draw();

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

shape3.draw();

}

}

**Output screenshot**

