

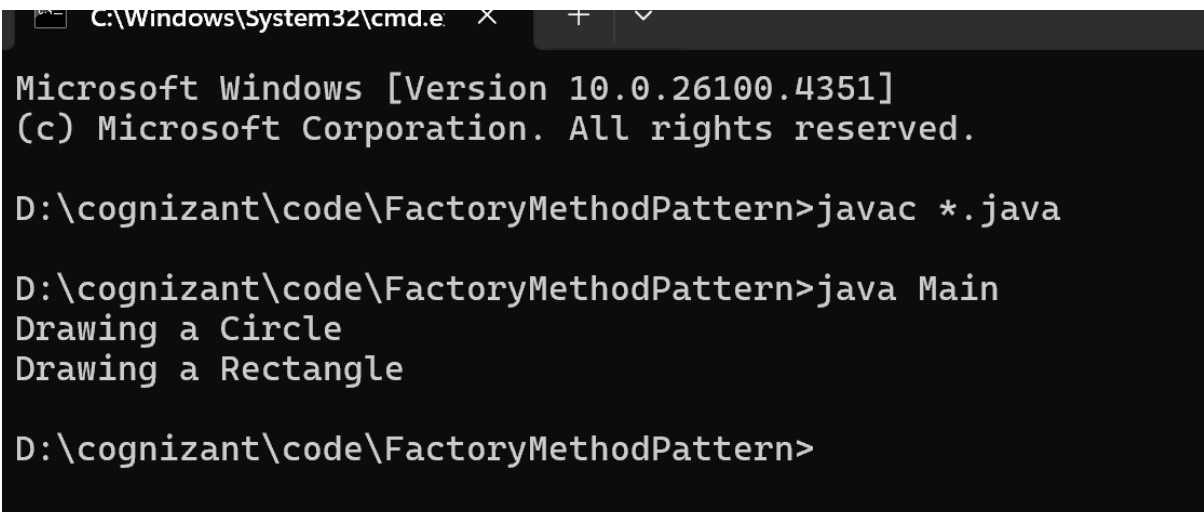
2.Implementation of Factory Method Pattern:

Code:

```
public interface Shape {  
    void draw();  
}  
  
public class Circle implements Shape {  
    public void draw() {  
        System.out.println("Drawing a Circle");  
    }  
}  
  
public class Rectangle implements Shape {  
    public void draw() {  
        System.out.println("Drawing a Rectangle");  
    }  
}  
  
public class ShapeFactory {  
    public Shape createShape(String shapeType) {  
        if (shapeType == null) {  
            return null;  
        }  
        if (shapeType.equalsIgnoreCase("CIRCLE")) {  
            return new Circle();  
        } else if (shapeType.equalsIgnoreCase("RECTANGLE")) {  
            return new Rectangle();  
        }  
        return null;  
    }  
}  
  
public class Main {
```

```
public static void main(String[] args) {  
    ShapeFactory factory = new ShapeFactory();  
  
    Shape shape1 = factory.createShape("CIRCLE");  
    shape1.draw();  
  
    Shape shape2 = factory.createShape("RECTANGLE");  
    shape2.draw();  
}  
}
```

Output:



```
C:\Windows\System32\cmd.e  x  +  v  
Microsoft Windows [Version 10.0.26100.4351]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\cognizant\code\FactoryMethodPattern>javac *.java  
  
D:\cognizant\code\FactoryMethodPattern>java Main  
Drawing a Circle  
Drawing a Rectangle  
  
D:\cognizant\code\FactoryMethodPattern>
```