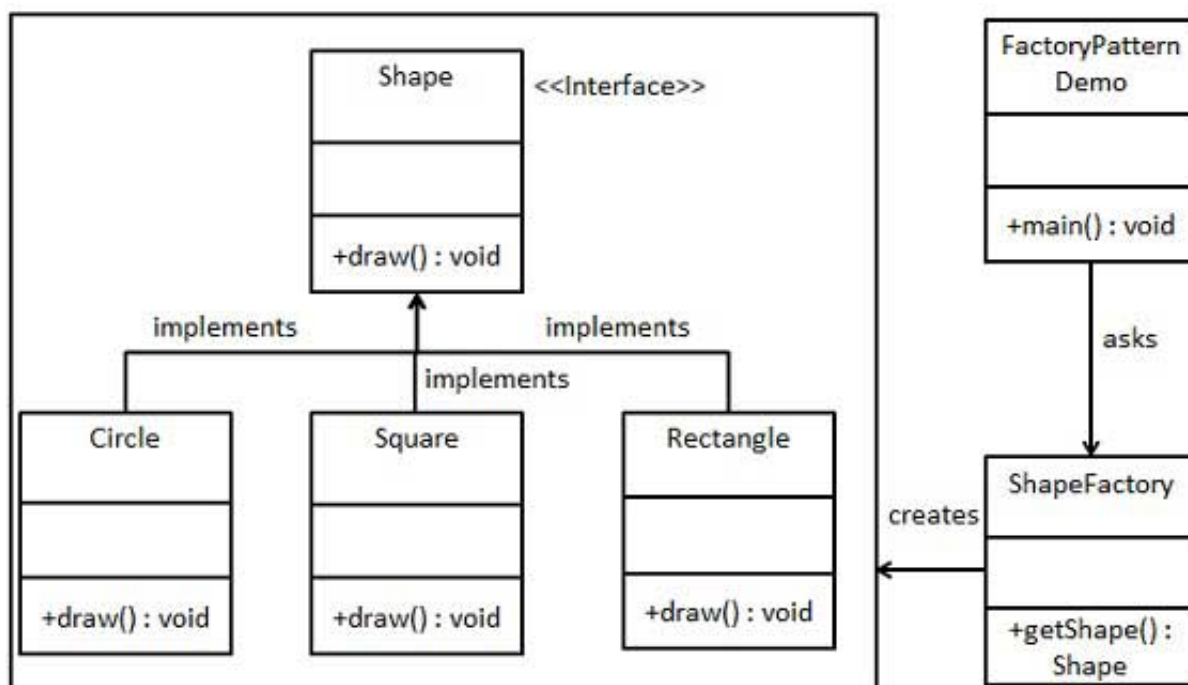# 工厂模式

工厂模式是Java中常用的一种设计模式。这个模式属于创建型模式，提供了一种创建对象的最佳方式。

在工厂模式中，我们可以在不向客户端暴露实现逻辑的情况下创建对象，使用一个通用的对象来创建对象。

实现

我们准备创建一个Shape接口和其具体的实现类。然后定义了一个工厂类 `ShapeFactory` 。`FactoryPatternDemo` ，我们的示例将会使用 `ShapeFactory` 来获得一个 `Shape` 对象。

第一步

创建一个接口。
Shape.java

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

第二步

Rectangle.java

```java
public class Rectangle implements Shape {
    @Override
    public void draw() {
        System.out.println("Inside Rectangle::draw() method.");
    }
}
```

Square.java

```java
public class Square implements Shape {
    @Override
    public void draw() {
        System.out.println("Inside Square::draw() method.");
    }
}
```

Circle.java

```java
public class Circle implements Shape {
    @Override
    public void draw() {
        System.out.println("Inside Circle::draw() method.");
    }
}
```

第三步

```java
ShapeFactory.java

public class ShapeFactory {

    //use getShape method to get object of type shape
    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;
     }
}
```

第四步

FactoryPatternDemo.java

```java
public class FactoryPatternDemo {

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

      //get an object of Circle and call its draw method.
      Shape shape1 = shapeFactory.getShape("CIRCLE");

      //call draw method of Circle
      shape1.draw();

      //get an object of Rectangle and call its draw method.
      Shape shape2 = shapeFactory.getShape("RECTANGLE");

      //call draw method of Rectangle
      shape2.draw();

      //get an object of Square and call its draw method.
      Shape shape3 = shapeFactory.getShape("SQUARE");

      //call draw method of circle
      shape3.draw();
   }
}
```

第五步

验证输出

```
Inside Circle::draw() method.
Inside Rectangle::draw() method.
Inside Square::draw() method.
```