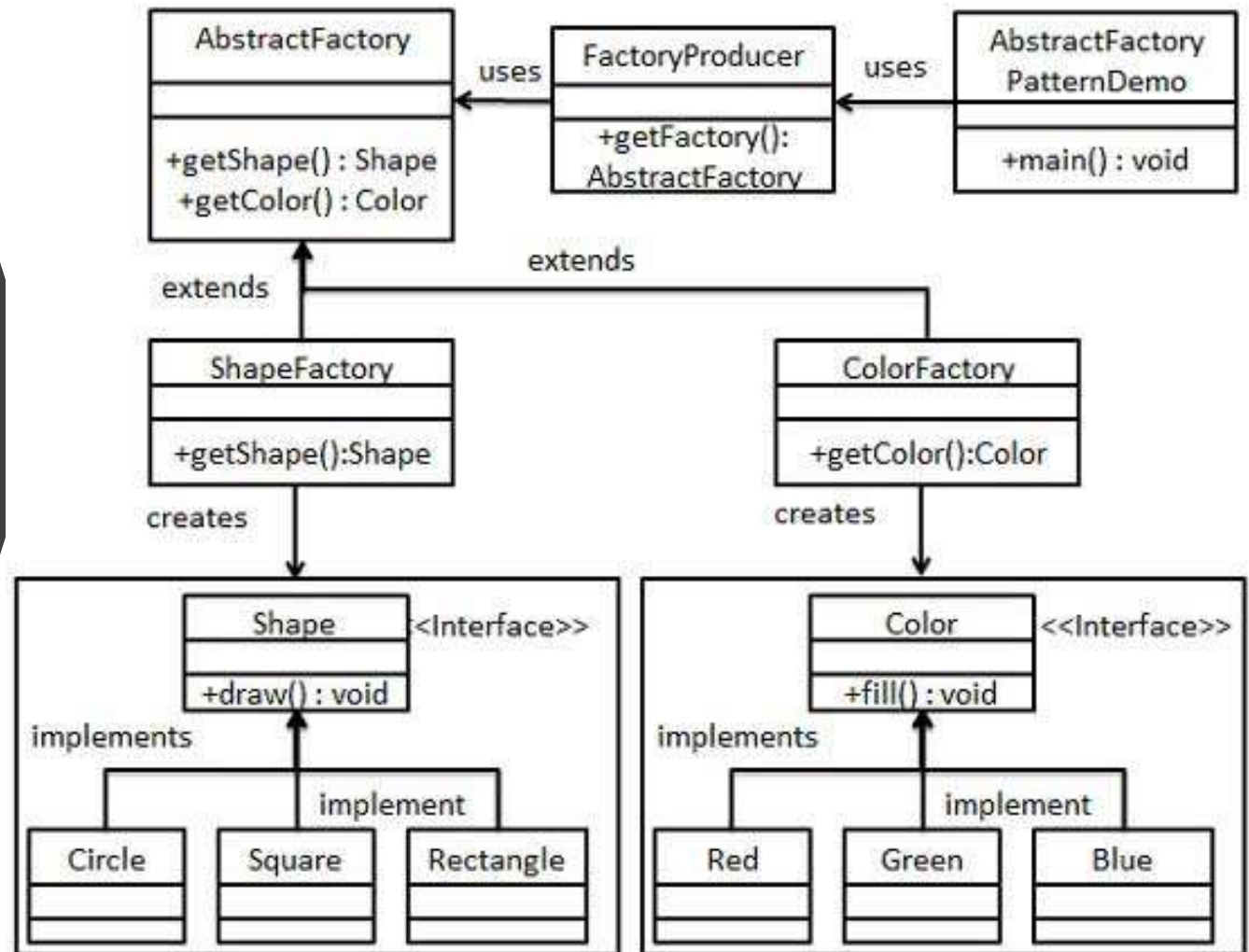


Abstract Factory Design



Tata Automobile

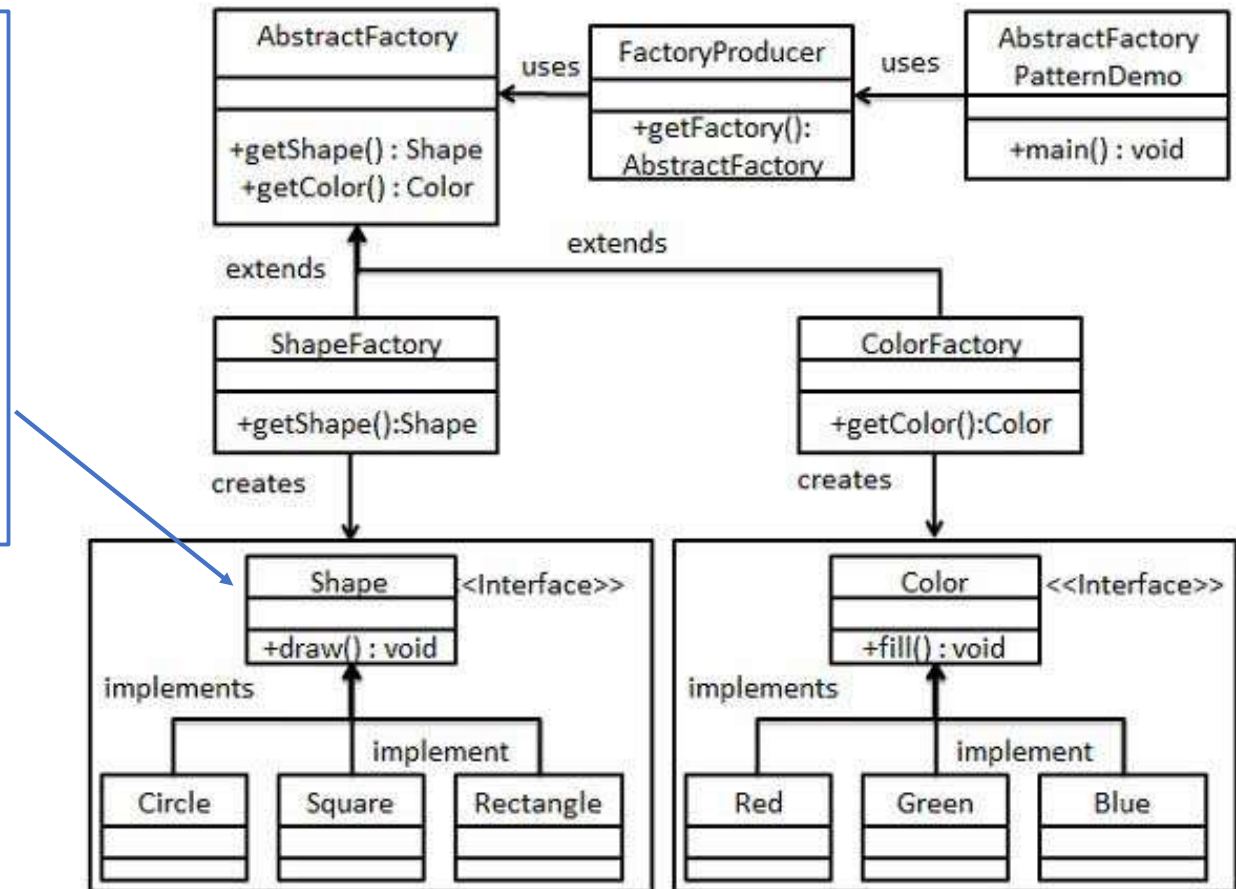
Simple Example



```
package com.adfp;
```

```
public interface Shape
```

```
{  
    void drawShape();  
}
```

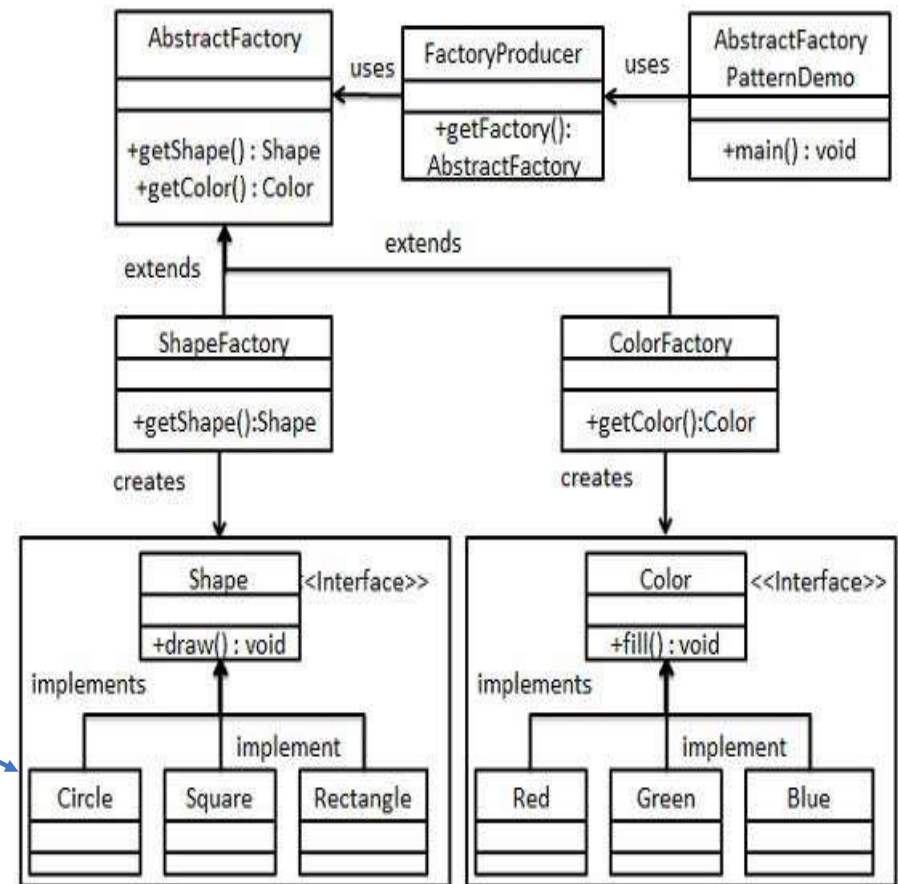


```

package com.adfp;

public class Circle implements Shape
{
    @Override
    public void draw()
    {
        System.out.println("Shape Circle.");
    }
}

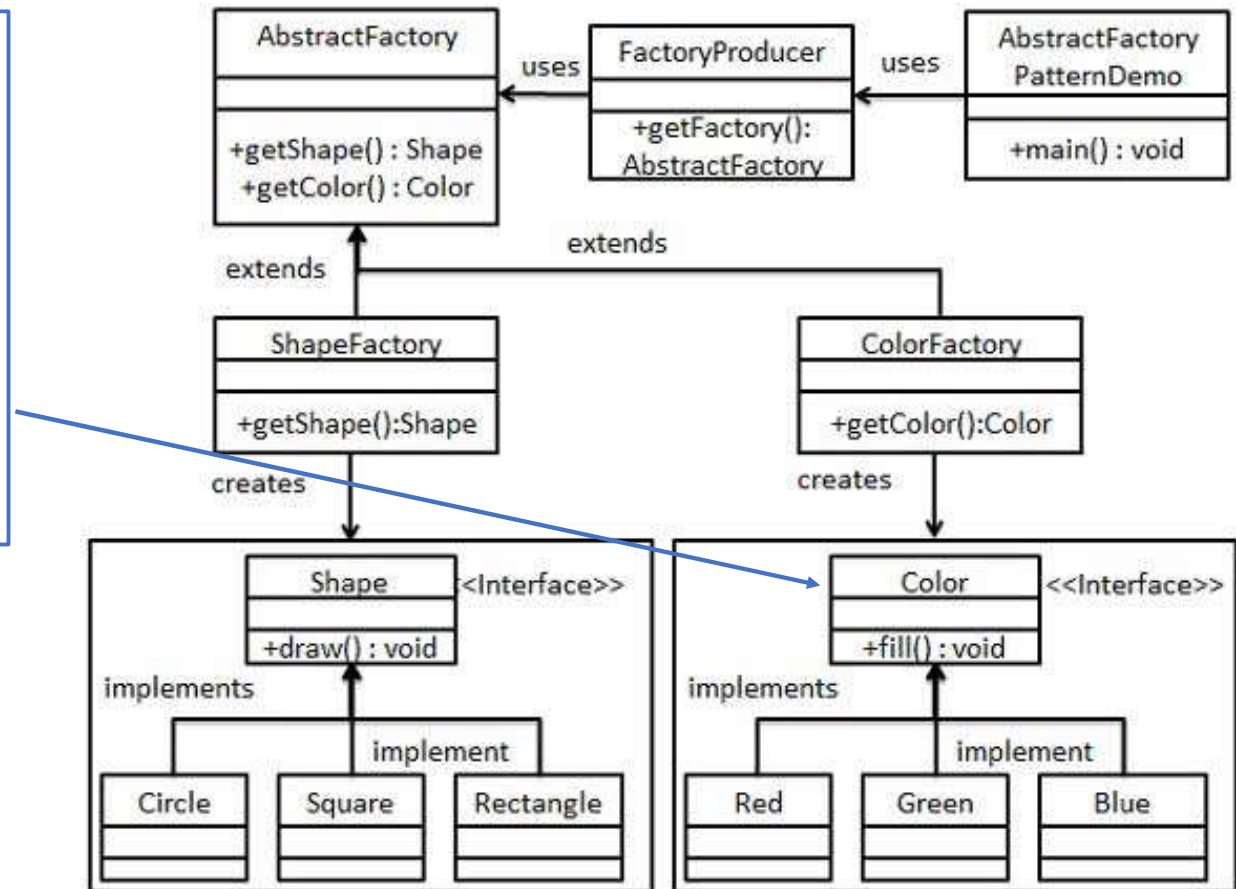
```



```
package com.adfp;
```

```
public interface Color
```

```
{  
    void fill();  
}
```

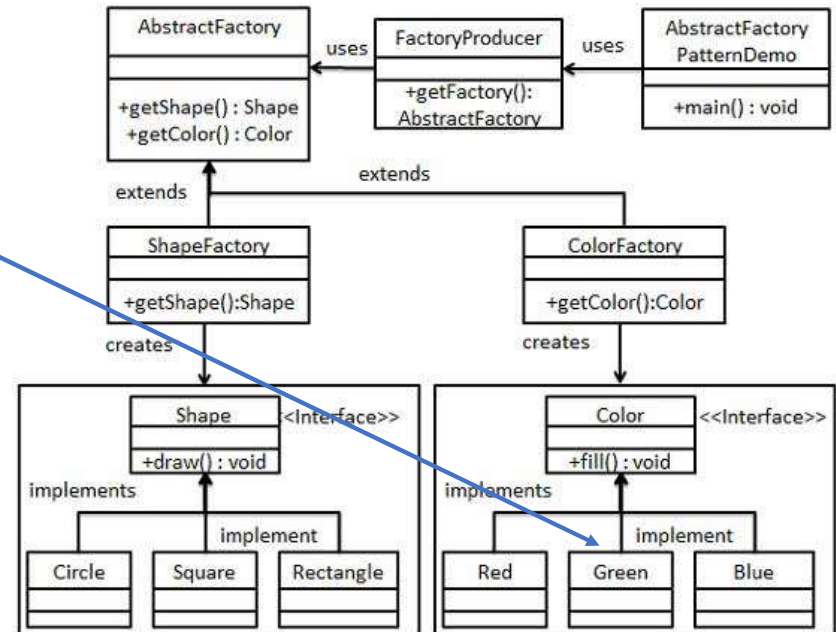


```

package com.adfp;

public class Green implements Color {
    @Override public void fill()
    {
        System.out.println("Fill Green Color.");
    }
}

```

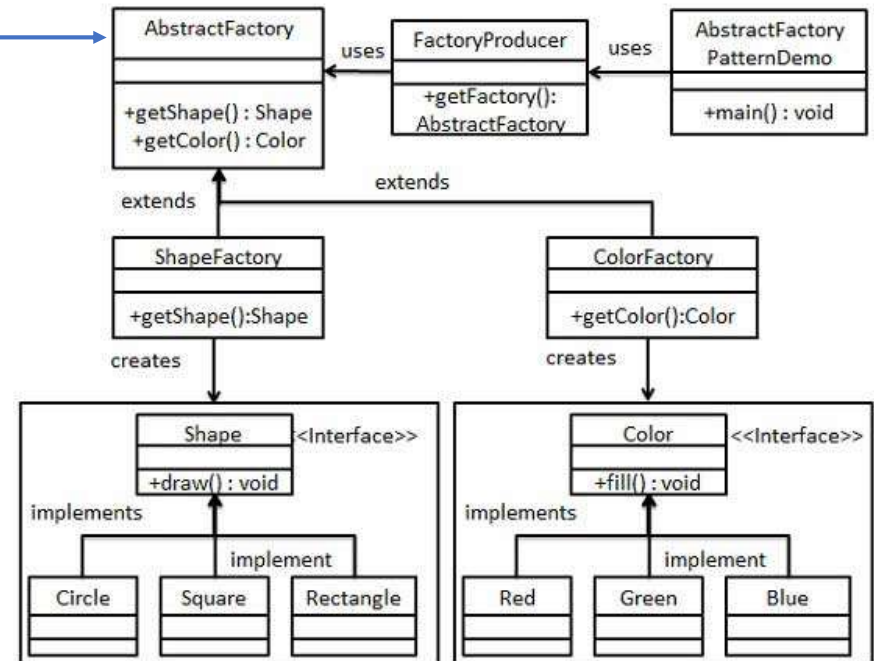



```

package com.adfp;

public abstract class AbstractFactory
{
    abstract Color getColor(String color);
    abstract Shape getShape(String shape);
}

```

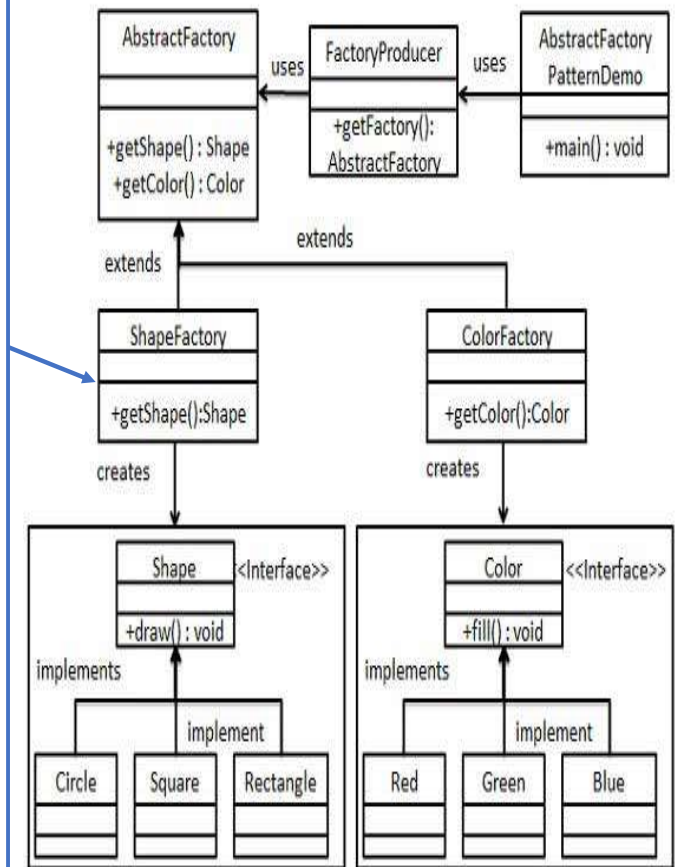



```

package com.adfp;

public class ShapeFactory extends AbstractFactory
{ //getShape method returns object of input type shape @Override
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; }
@Override
Color getColor(String color)
{ return null; } }

```

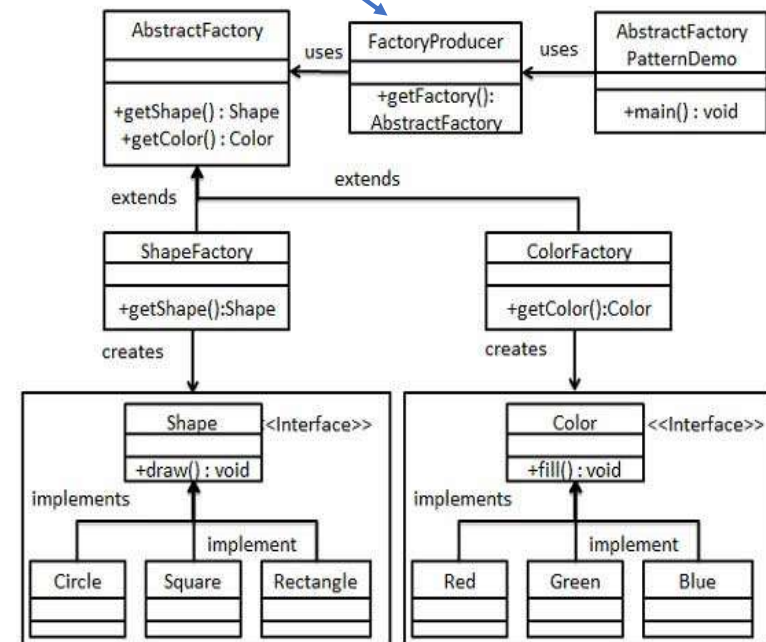


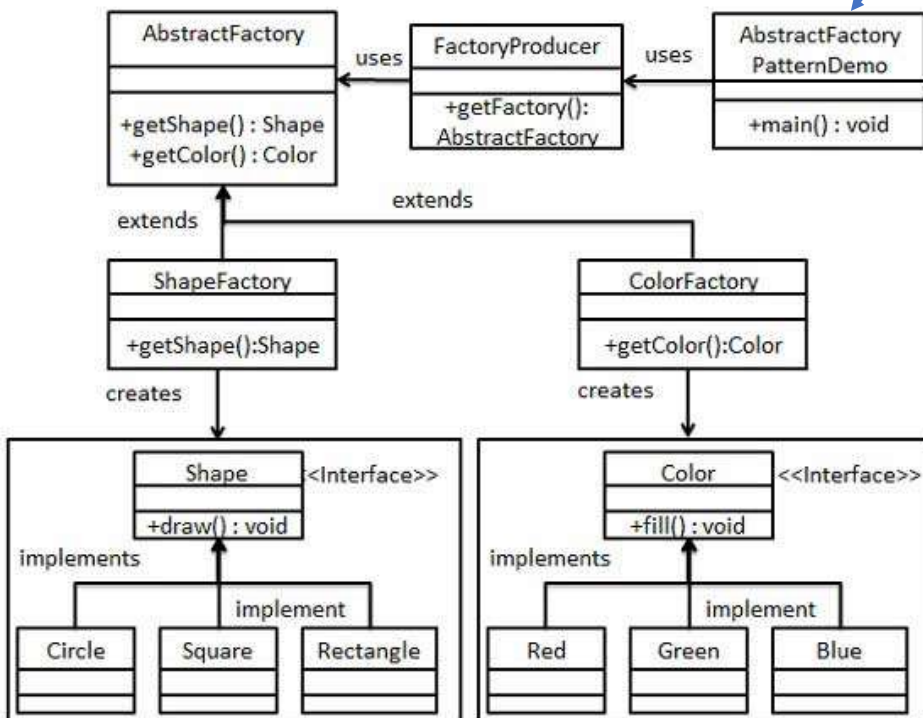
```

package com.adfp;
public class FactoryProducer {
    public static AbstractFactory getFactory(String choice)
    { if(choice.equalsIgnoreCase("Shape"))
    {
        return new ShapeFactory();
    }

    else if(choice.equalsIgnoreCase("Color"))
    {
        return new ColorFactory();
    }
    return null;
    }
}

```





```

package com.adfp;

public class AbstractFactoryPatternTest
{
    public static void main(String args[])
    {
        //get color factory
        AbstractFactory shapeFactory =
            FactoryProducer.getFactory("Shape");

        //Get Circle class object
        Shape shape1 = shapeFactory.getShape("Circle");

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

        /get color factory
        AbstractFactory colorFactory =
            FactoryProducer.getFactory("Color");

        //get an object of Color Red
        Color color1 =
            colorFactory.getColor("Red");

        /call fillColor method of Red
        color1.fillColor();
    }
}

```