

بررسی الگوهای طراحی

Decorator

توسط : محمد حسین زارع

@HoseinZare67: twitter

telegram

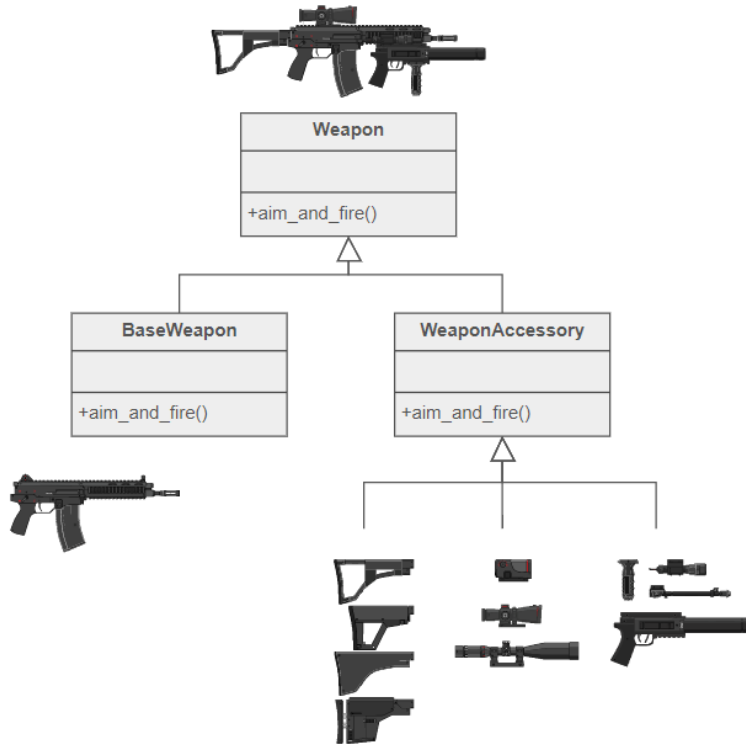
Join Us : @JavaLandCH

<http://javaland.blog.ir/>

مفاهیم

- به اسم wrapper شناخته میشود
- اضافه کردن رفتار بدون تأثیر روی اصل پیاده سازی
- **Single Responsibility Principle**
- بهره بری از composition
- نمونه ها
- **java.io.InputStream**
- **java.util.Collections#checkedList**
- **UI Components**

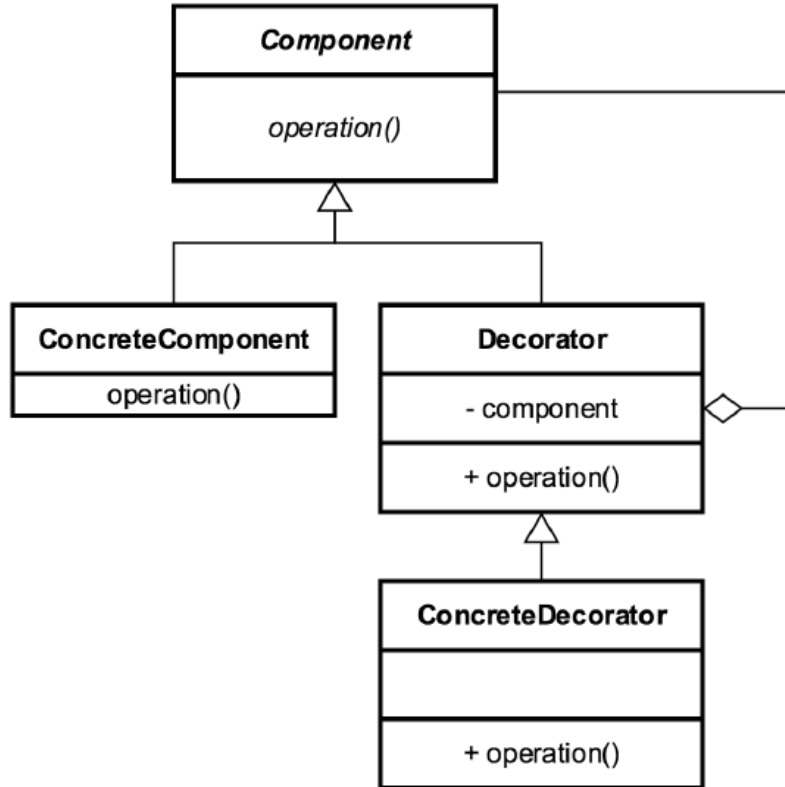
مفاهيم



طراحی

- به کارگیری ترکیب و ارث بری با هم
- جایگزینی برای subclassing
- Constructor ها همیشه به یک شیء از ساختار ارث بری
احتیاج دارند

طراحی



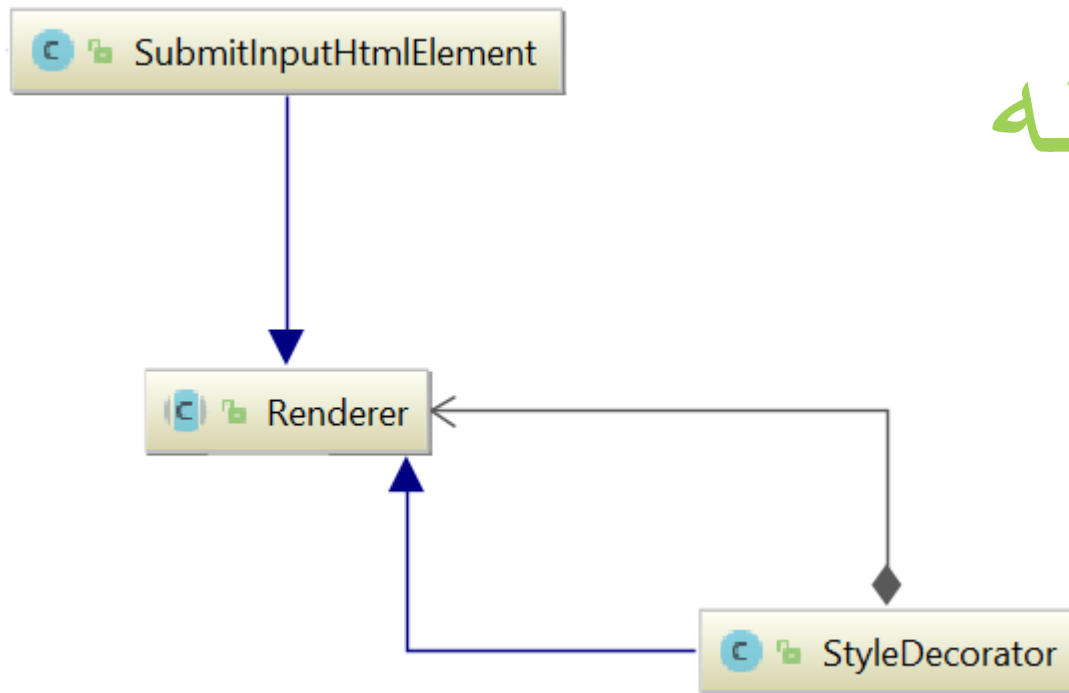
مثال 1

```
File file = new File("./output.txt");  
file.createNewFile();
```

```
OutputStream oStream = new FileOutputStream(file);  
DataOutputStream doStream = new DataOutputStream(oStream);
```

```
doStream.writeLong(14L);  
doStream.writeFloat(12.5F);
```

یک نمونه



یک نمونه

```
public abstract class Renderer {  
  
    private Attribute[] attributes;  
    public Renderer(Attribute... attributes) {  
        this.attributes = attributes;  
    }  
    public Renderer() {  
    }  
    public void applyAttributes(Attribute... attributes) {  
        this.attributes = attributes;  
    }  
    public abstract String write();  
}
```


یک نمونه

```
public class SubmitInputElement extends Renderer {  
  
    @Override  
    public String write() {  
        return "<input type='submit' "  
            + writeAttributes(getAttributes()) +  
            ">";  
    }  
}
```

یک نمونه

```
public class StyleDecorator extends Renderer {  
    private Renderer renderer;  
  
    public StyleDecorator(Renderer renderer) {  
        this.renderer = renderer;  
    }  
    @Override  
    public String write() {  
String styles = writeAttributesAsStyle(getAttributes());  
        if (styles != null)  
            styles = "style=\\"" + styles + "\"";  
        return "<div class='form-input' " + styles + ">"  
            + renderer.write() + "</div>";  
    }  
}
```

جمع بندی

- ساختار کلاس اصلی دست نمیخورد
- یک راه مناسب برای اضافه کردن رفتار جدید به یک سیستم
- ممکن است با ارث بری اشتباه گرفته شود
- استفاده از آن برای کلاینت ها کمی پیچیده است