

بررسی الگوهای طراحی -

Proxy

توسط : محمد حسین زارع

@zare88r: twitter

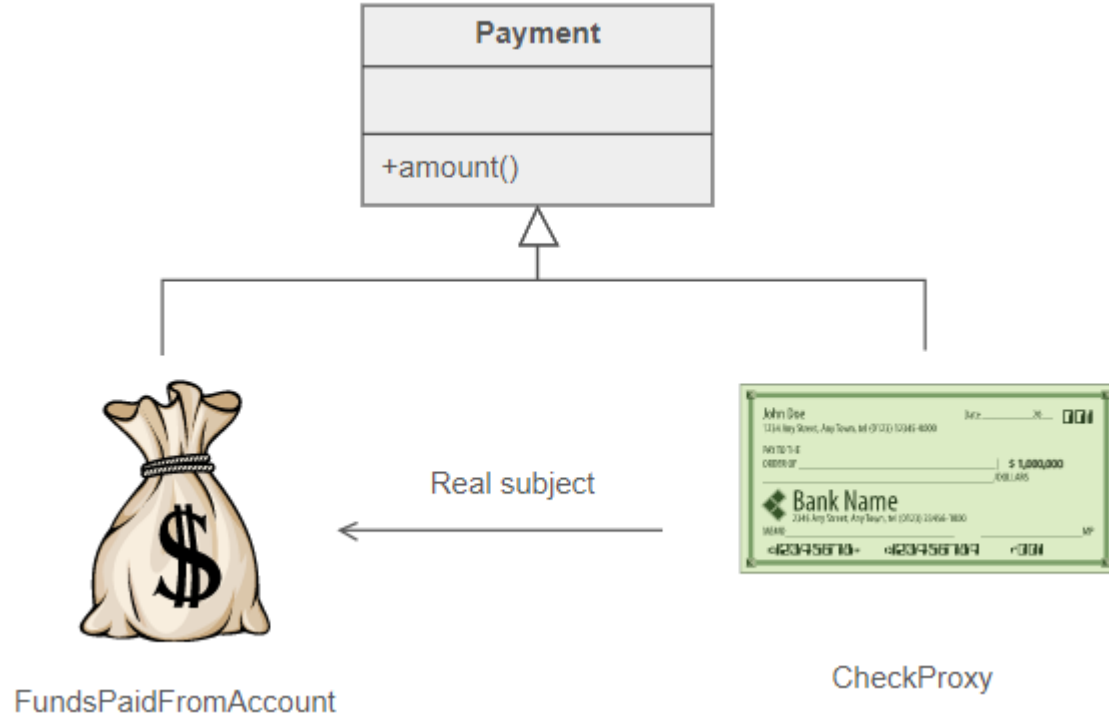
telegram

Join Us : @JavaLandCH

مفاهیم

- امکان wrapping به یک interface
- میتواند به یک شیء functionality اضافه نماید.
- دسترسی مستقیم به شیء اصلی از طریق Proxy
- نمونه ها
- `java.lang.reflect.Proxy`
- `java.rmi.*`

مفاهيم

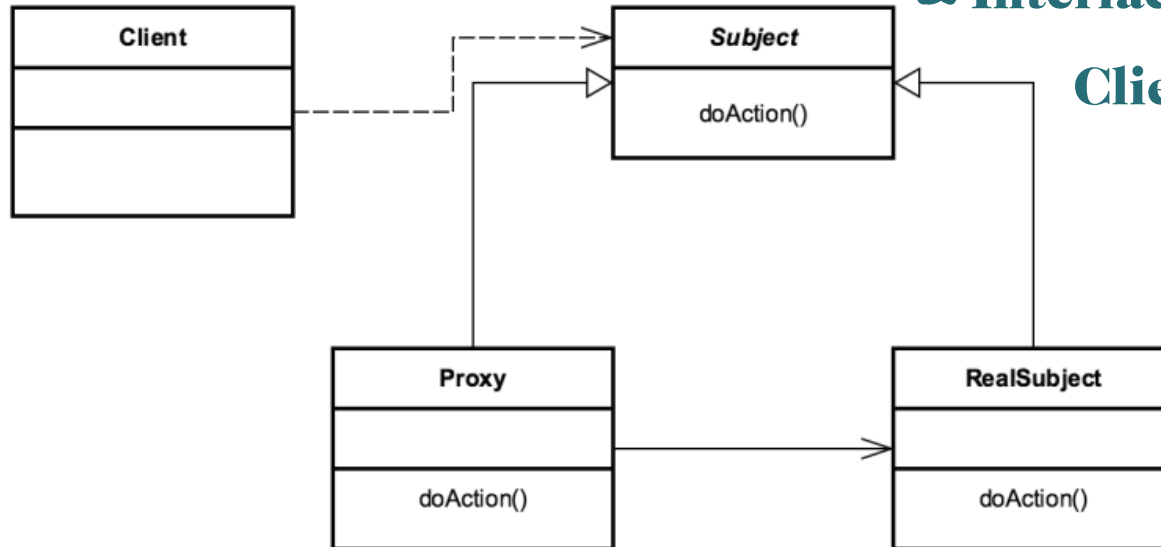


طراحی

▪ بر مبنای Interface

▪ پیاده سازی Interface ها

▪ Client و Proxy



مثال 1

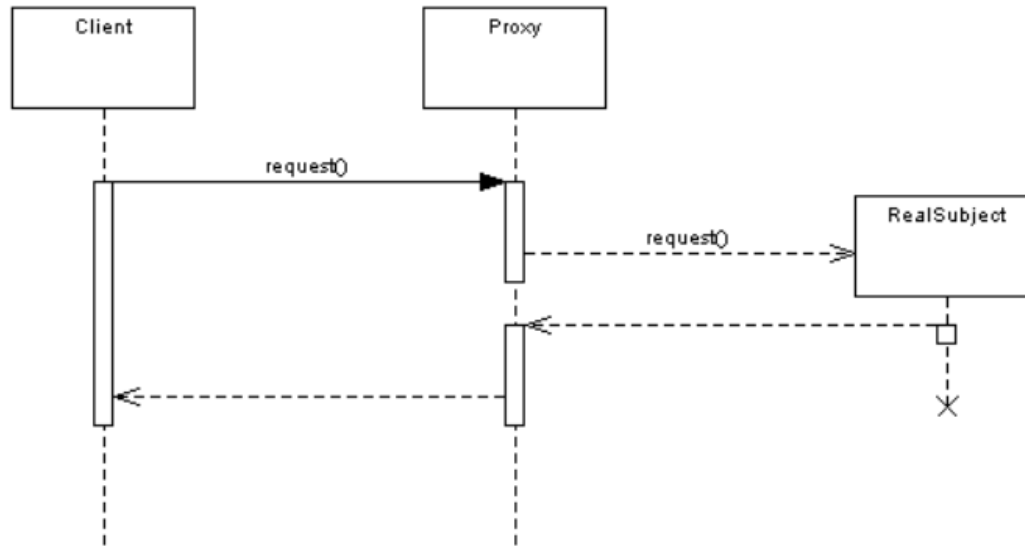
```
public class Test {  
    public int plus(int a, int b) {  
        return a + b;  
    }  
    public static void main(String[] args) throws Exception{  
        Object obj = Class.forName("com.javaland.proxy.Test")  
            .newInstance();  
        Method plus = obj.getClass().getMethod("plus",  
            new Class[]{int.class, int.class});  
        Object result = plus.invoke(obj, 12, 34);  
        System.out.println(result);  
    }  
}
```

تفاوت با Decorator

Proxy

- . میتواند قابلیت اضافه کند ولی هدف اصلی آن نیست
- . ارتباط شیء اصلی با Proxy در زمان کامپایل مشخص میشود
- . هدف اصلی Proxy کنترل کردن شیء اصلی است به طور مثال برای پیاده سازی امنیت ، caching ، lazy objects و ...

Sequence Diagram



یک نمونه

```
public interface Image {  
    void displayImage();  
}
```


یک نمونه

```
public class RealImage implements Image {  
    public RealImage(URL url) {  
        //load up the image  
        loadImage(url);  
    }  
    public void displayImage() {  
        // a method that only the real image has  
    }  
    private void loadImage(URL url) {  
        //do resource intensive operation to load image  
    }  
}
```

یک نمونه

```
public class ProxyImage implements Image {  
    private URL url;  
  
    public ProxyImage(URL url) {  
        this.url = url;  
    }    //this method delegates to the real image  
  
    public void displayImage() {  
        RealImage real = new RealImage(url);  
        real.displayImage();  
    }  
}
```

جمع بندی

- امکان کنترل یک شیء را به یک API میدهد
- در فریم ورک ها برای پیاده سازی امنیت ، سادگی ، صدا زدن از راه دور ، DI یا IOC استفاده میشود
- از لحاظ ساختار به الگوهای دیگر بسیار شبیه است