

بررسی الگوهای طراحی -

Visitor

توسط : محمد حسین زارع

@zare88r : twitter

telegram

Join Us : @JavaLandCH

مفاهیم

- جدا کردن الگوریتم از آبنجکت
- اضافه کردن ویژگی های جدید
- استفاده از Open/Close Principle
- نمونه ها :
- `java.lang.model.element.Element`
- `java.lang.model.element.ElementVisitor`

مفاهيم



Cab company dispatcher

Object structure is
list of Customers



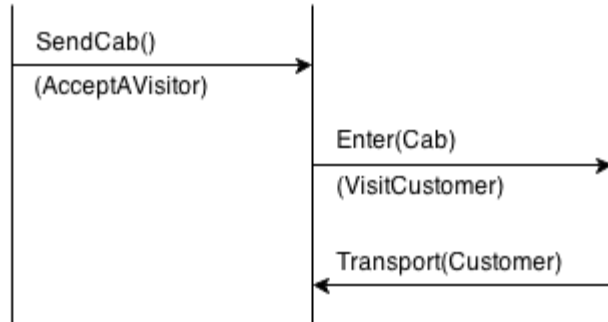
Customer

Concrete element of
Customers list

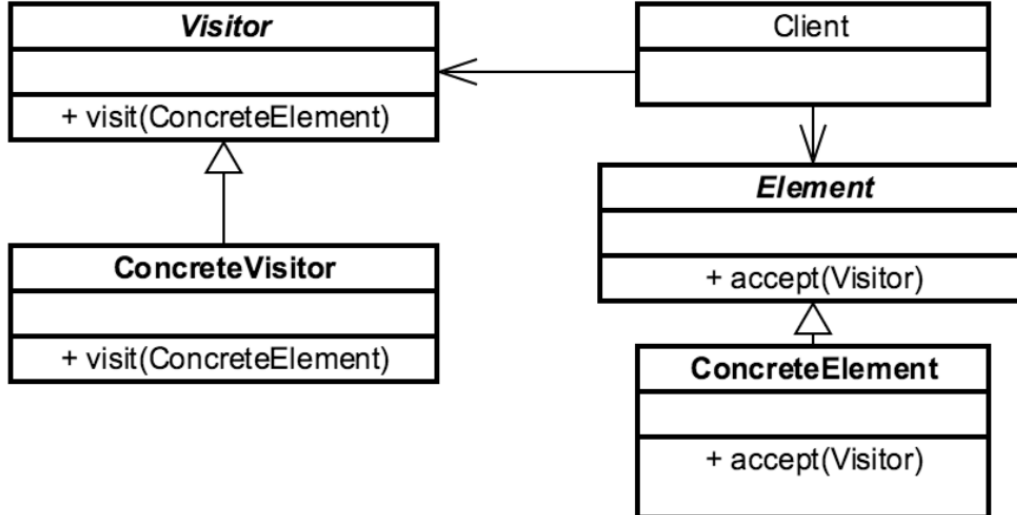


Taxi

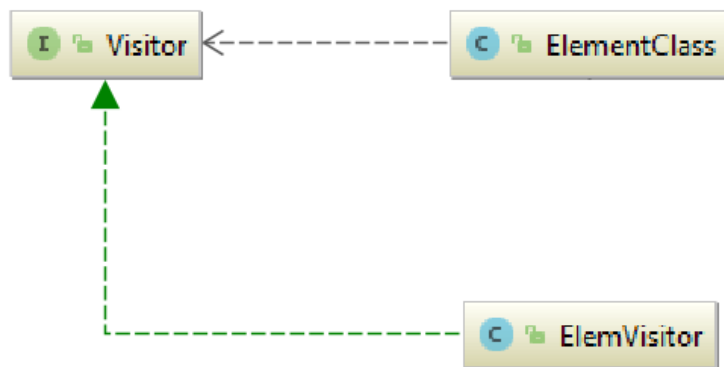
Visitor



- **Client**
- **Element (Concrete Element)**
- **Visitor (Concrete Visitor)**



یک نمونه



یک نمونه

```
public class ElementClass {  
    private String value;  
  
    public ElementClass(String value) {  
        this.value = value;  
    }  
  
    public String getValue() {  
        return value;  
    }  
  
    public void setValue(String value) {  
        this.value = value;  
    }  
  
    public void accept(Visitor visitor) {  
        visitor.visit(this);  
    }  
}
```

یک نمونه

```
public interface Visitor {  
    void visit(ElementClass elementClass);  
}
```

```
public class ElemVisitor implements Visitor{  
  
    public void visit(ElementClass elementClass) {  
        elementClass.setValue(elementClass.getValue() + "ADDED VALUE ");  
    }  
}
```

جمع بندی

- برای زمانی که امکانی جهت **adapting** نیاز داریم
- بیرون آوردن تغییرات
- کم کردن پیچیدگی