

# SOLID & Design patterns

Khoi Nguyen

# Agenda

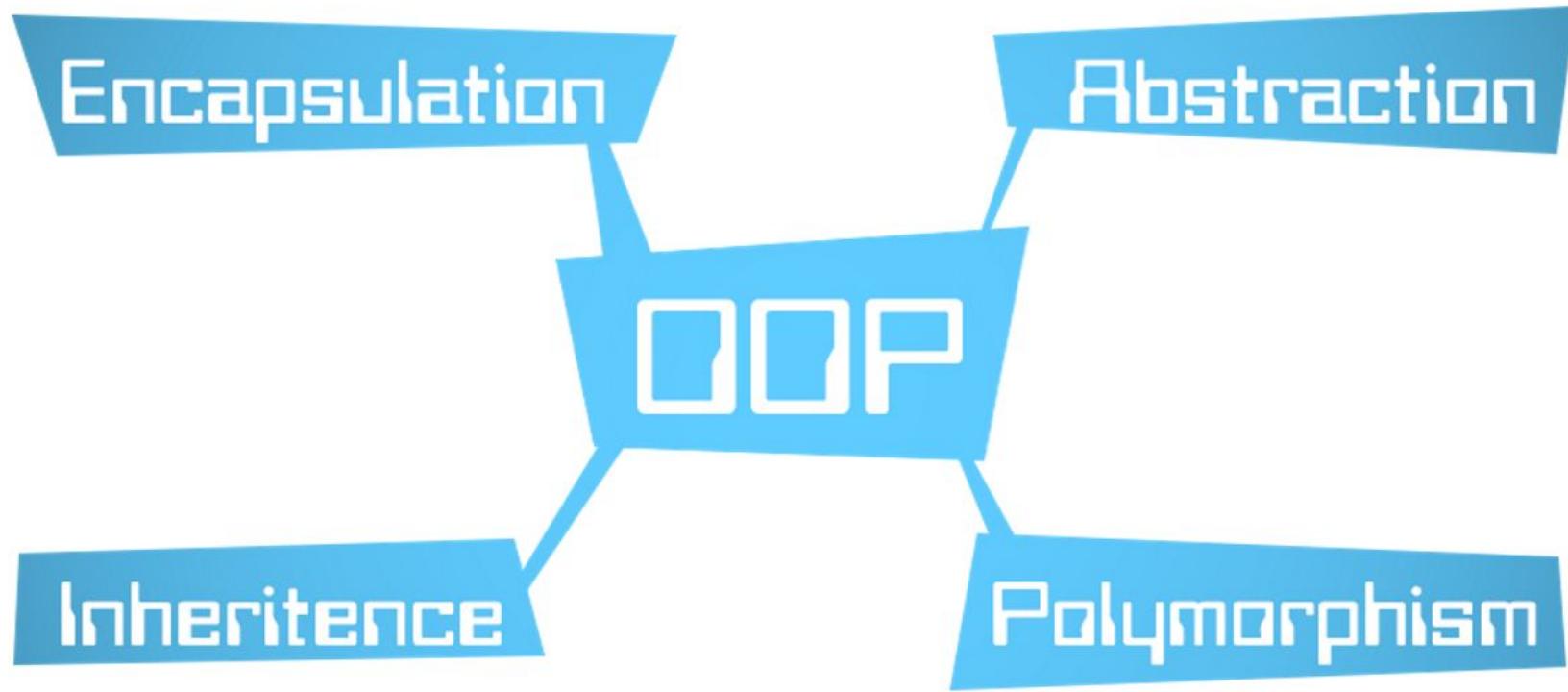
- Intro
- SOLID
- Design patterns
  - Decorator
  - Strategy
  - Visitor
- Q & A

# Intro

- Design patterns không khó hiểu nếu chúng ta nắm vững SOLID.
- Design patterns trở nên quen thuộc, gần gũi.
- Design patterns trở thành công cụ giao tiếp.



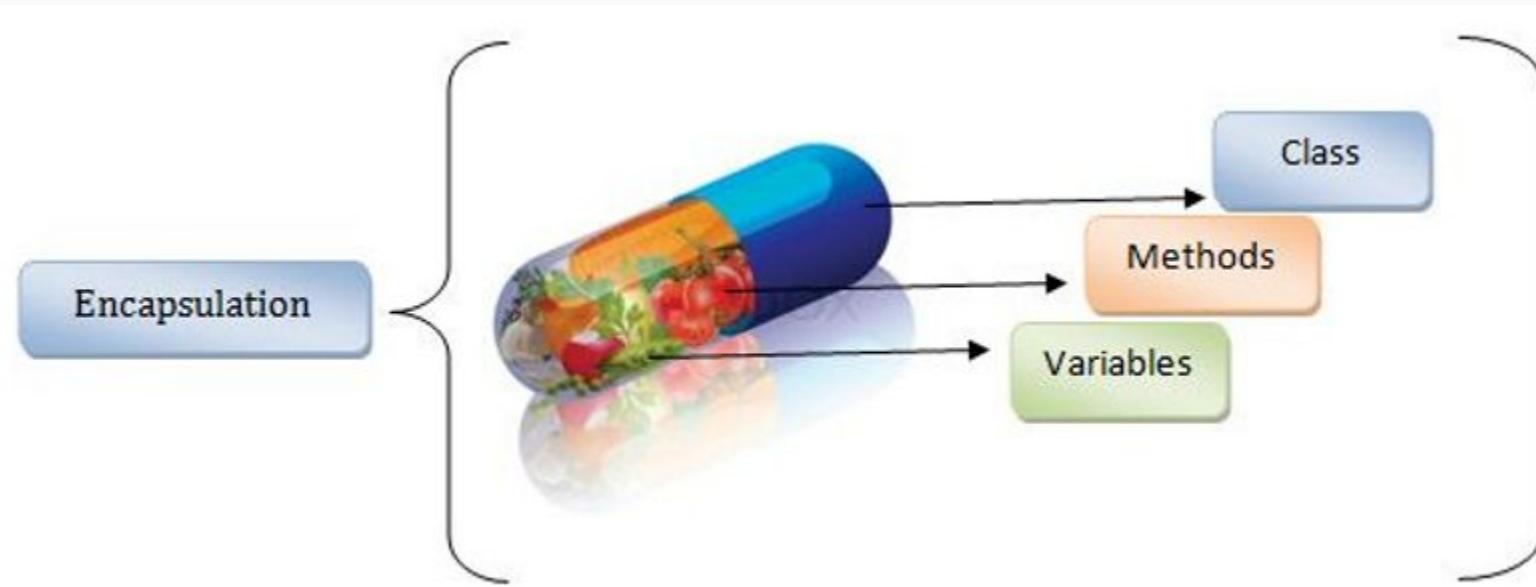
Xả stress tại [xem.vn](http://xem.vn)



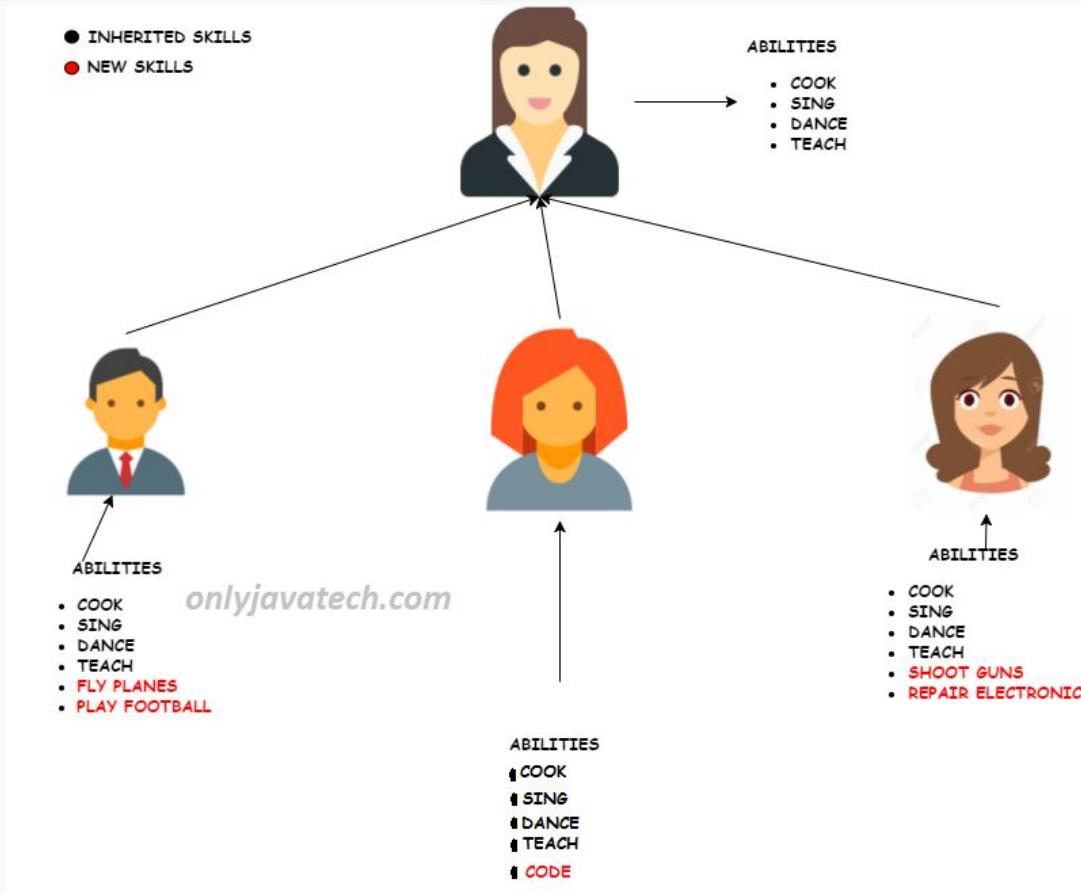
## OOP > Abstraction



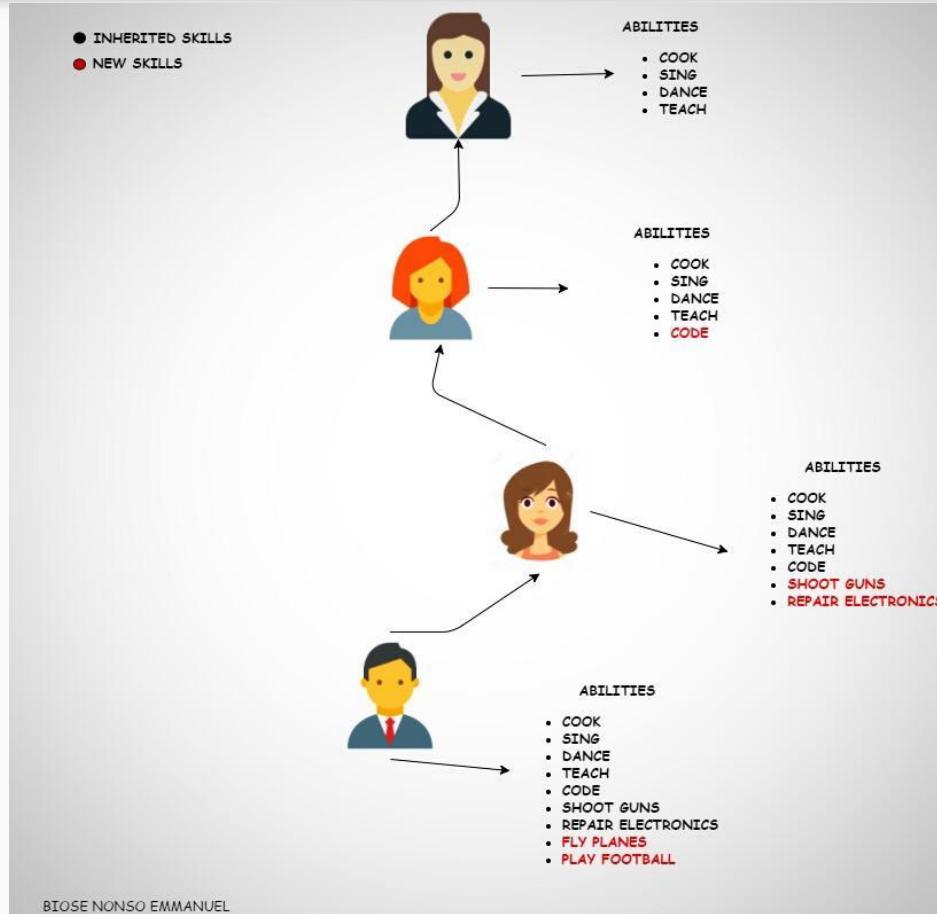
## OOP > Encapsulation



# OOP > Inheritance



# OOP > Inheritance



# OOP > Polymorphism

edureka!



# S. O. L. I. D.

**S**ingle  
Responsibility

**O**pen-closed  
Principle

**L**iskov  
substitution  
Principle

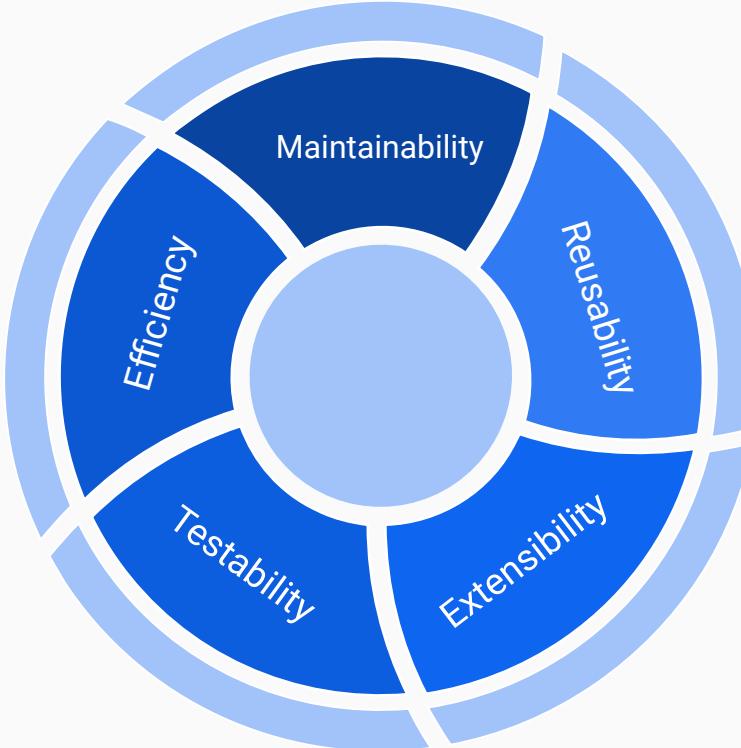
**I**nterface  
Segregation  
Principle

**D**ependancy  
Inversion  
Principle

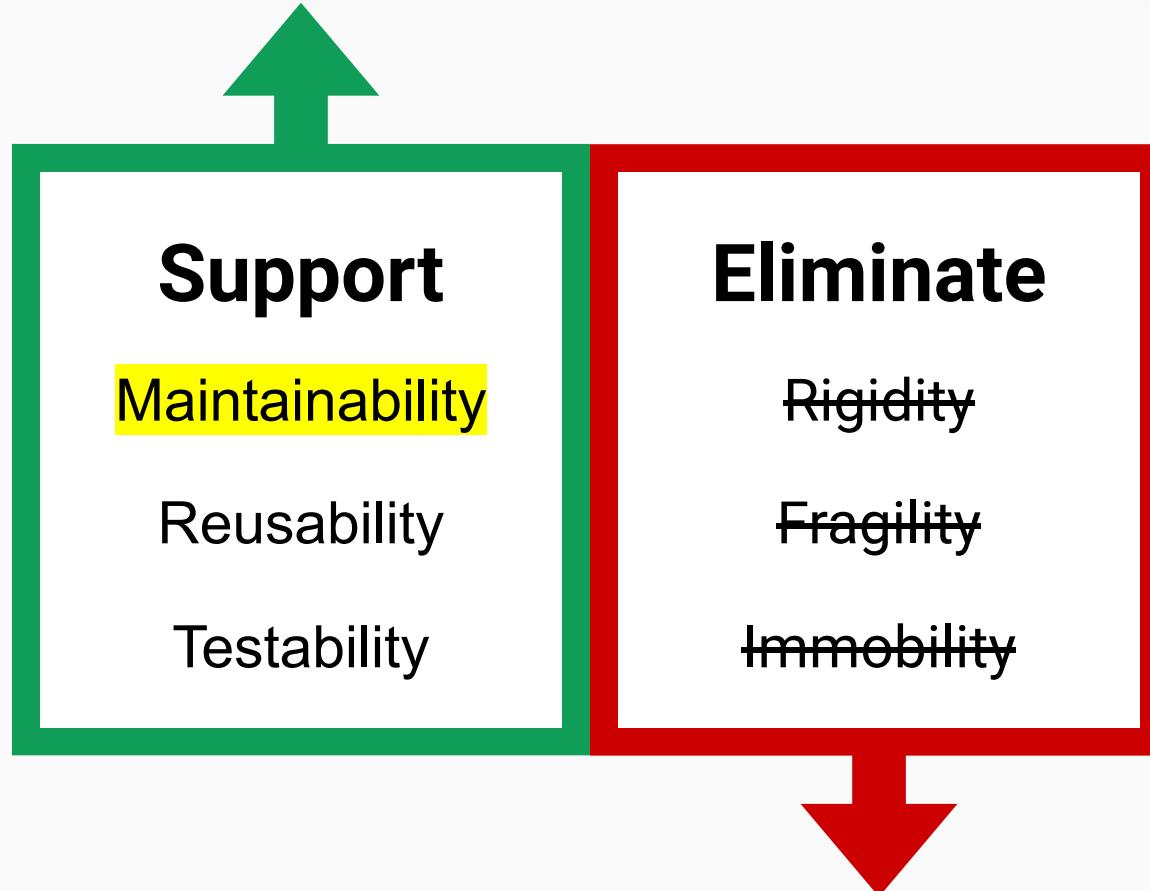
## SOLID > Bad Design



# SOLID > Good Design



## SOLID > Single Responsibility



## SOLID > Single Responsibility

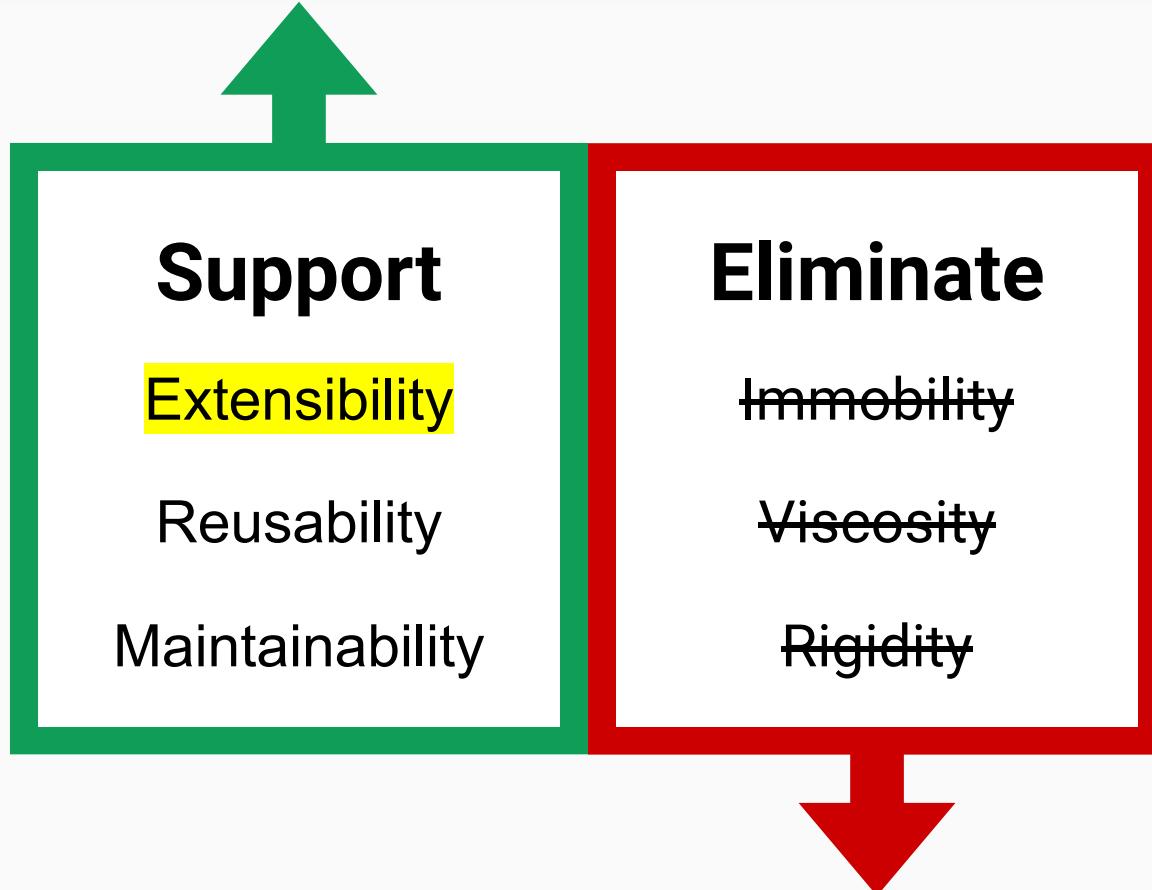
“

A class should have one, and only one, reason to change.



### SINGLE RESPONSIBILITY PRINCIPLE

Just Because You Can, Doesn't Mean You Should



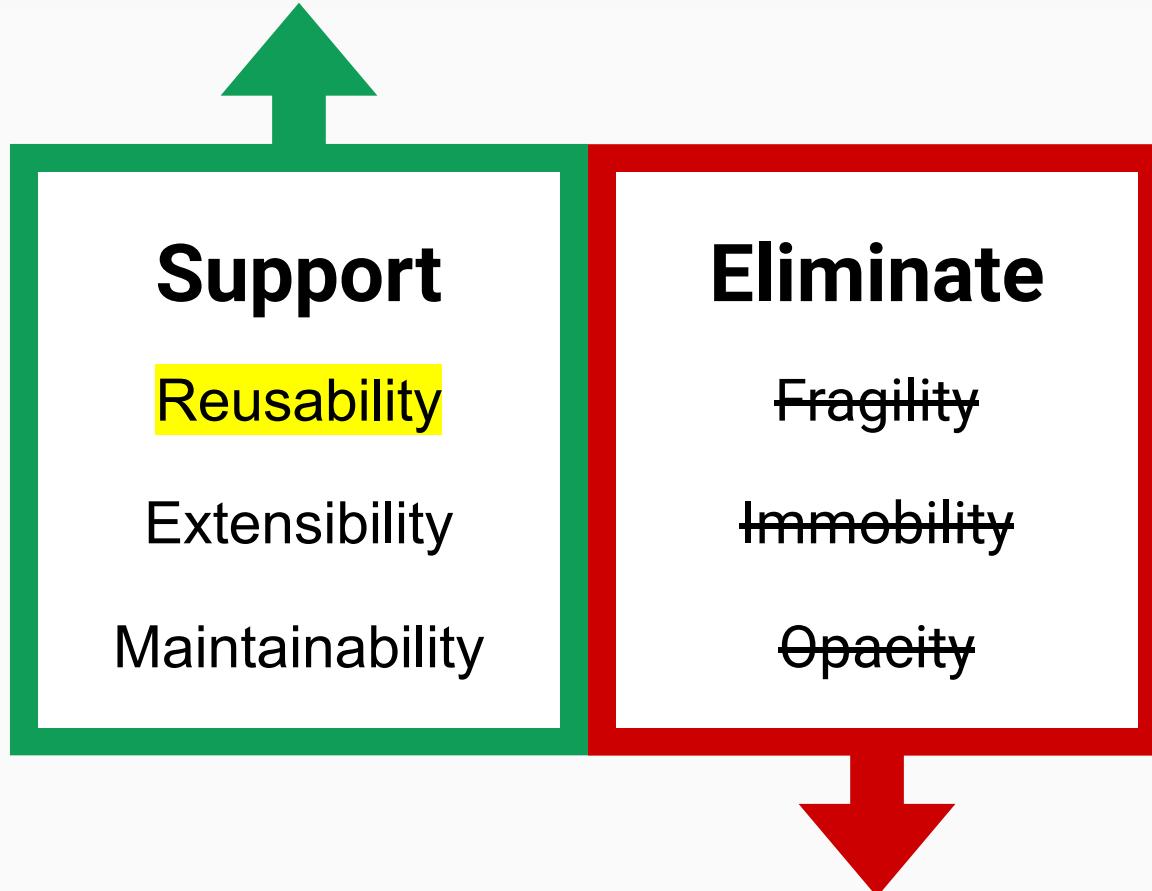
# SOLID > Open-Closed

“

You should be able to extend a classes behavior, without modifying it.



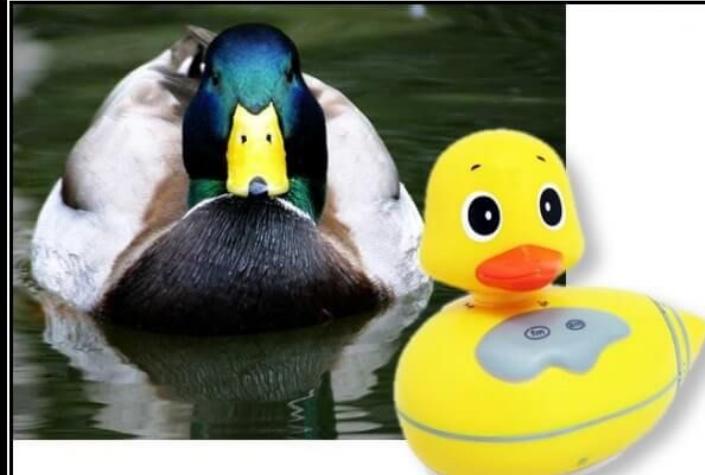
## SOLID > Liskov substitution



## SOLID > Liskov substitution

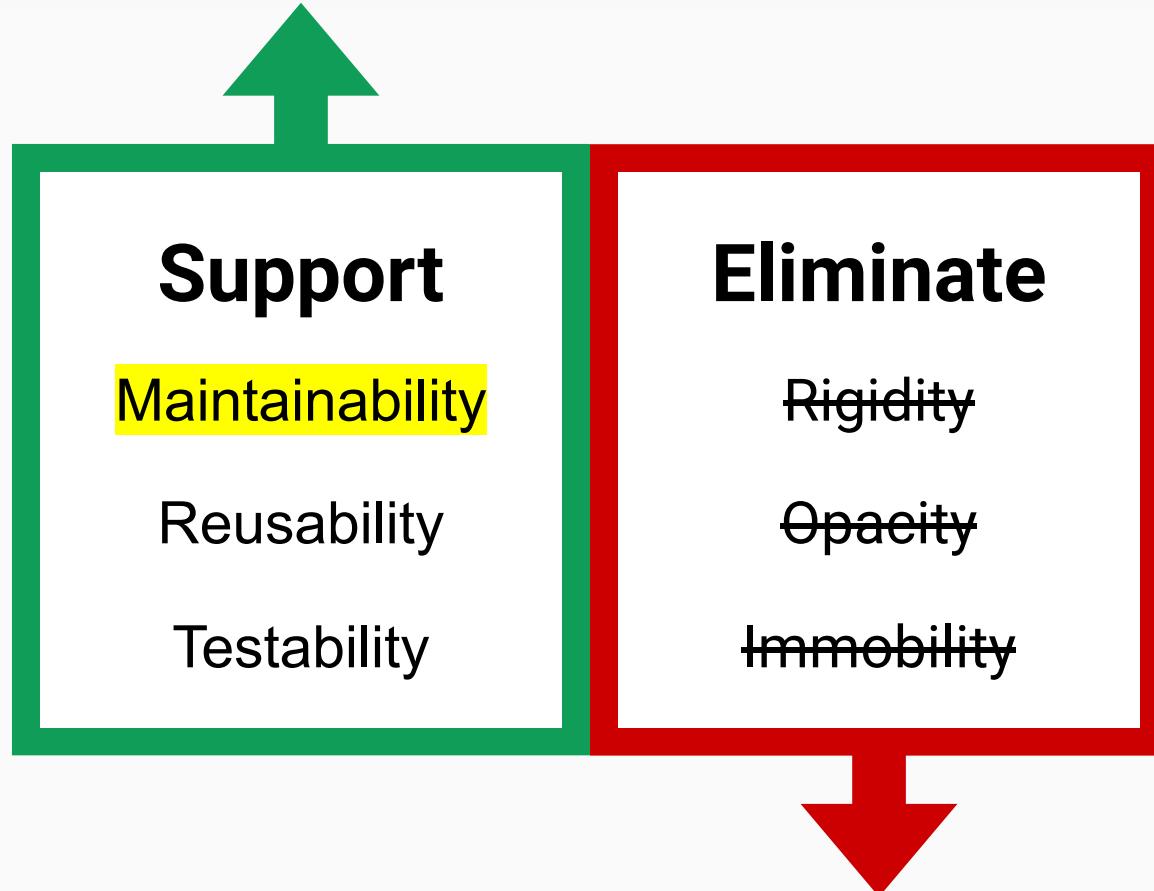
“

Derived classes must be substitutable for their base classes.



### LISKOV SUBSTITUTION PRINCIPLE

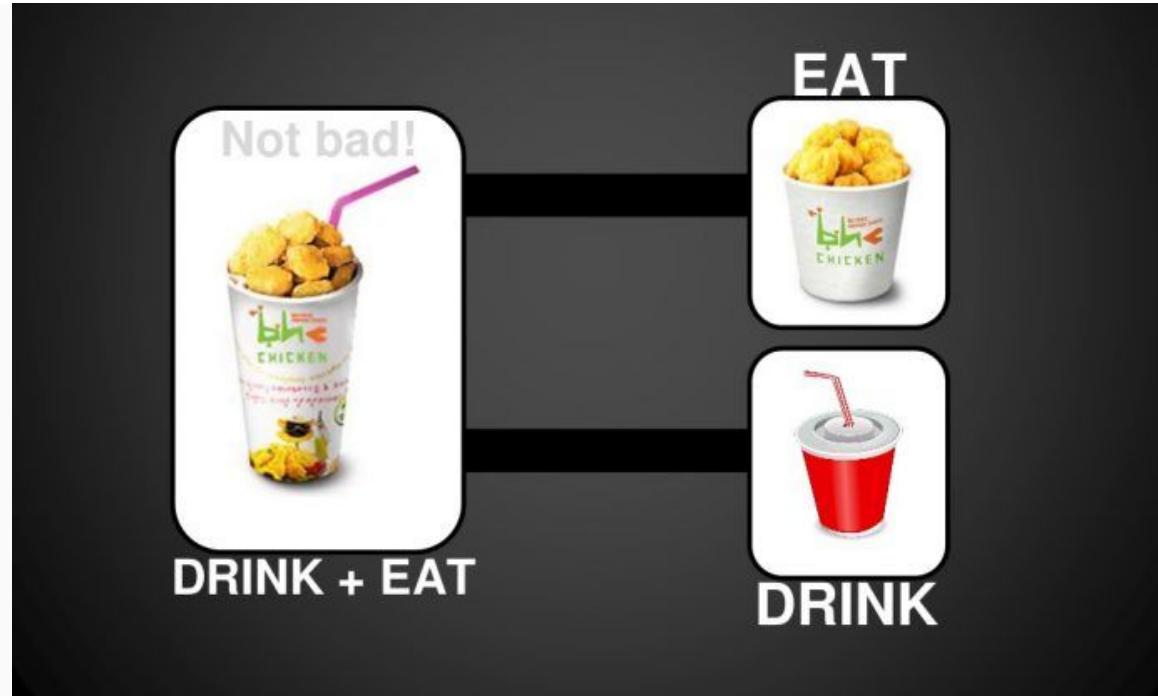
If It Looks Like A Duck, Quacks Like A Duck, But Needs Batteries - You  
Probably Have The Wrong Abstraction

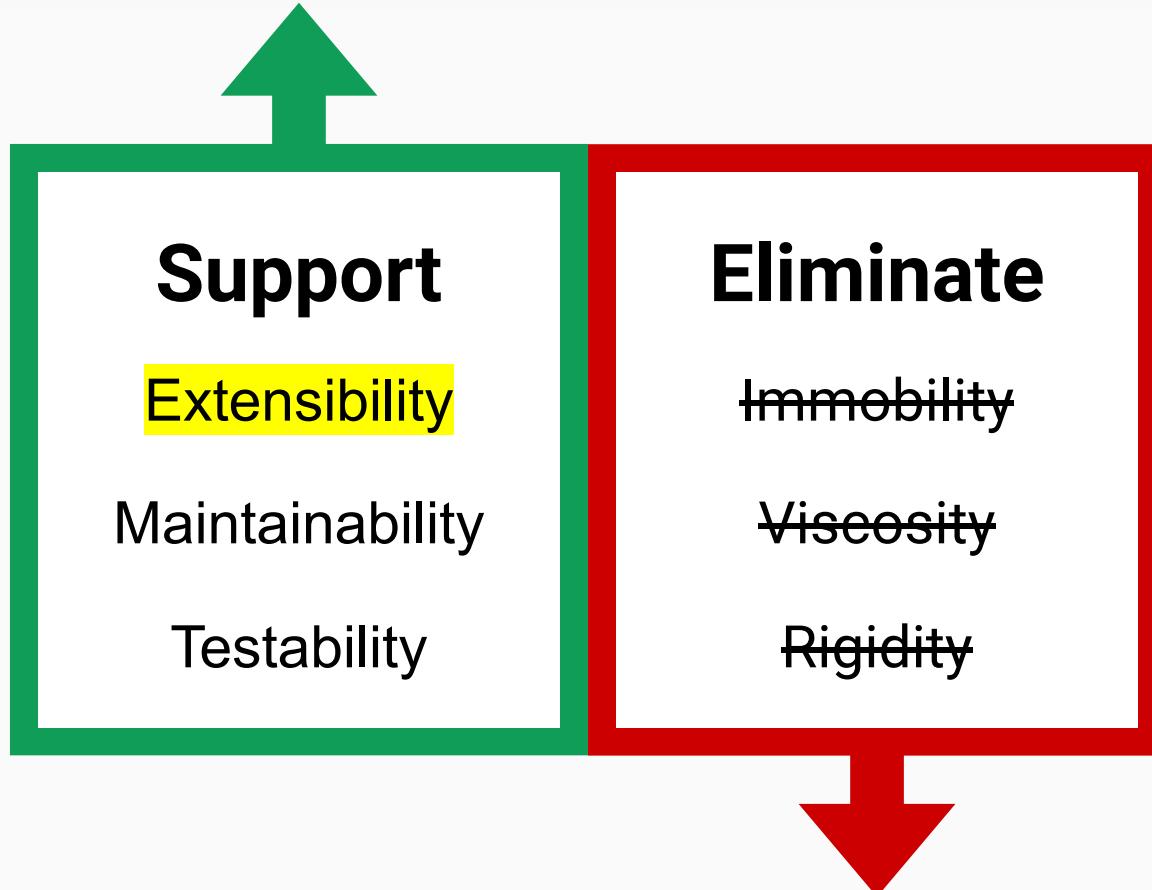


## SOLID > Interface Segregation

“

Make fine grained interfaces that are client specific.

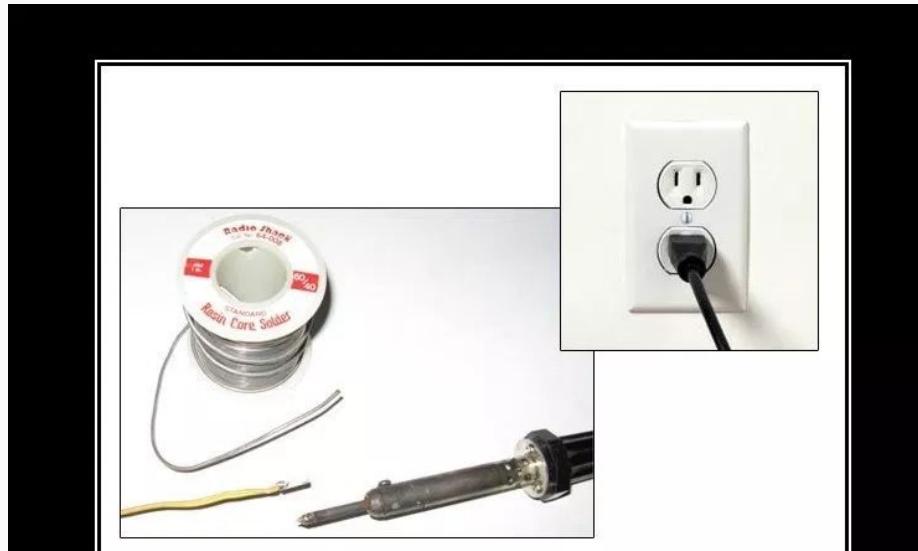




## SOLID > Dependency Inversion

“

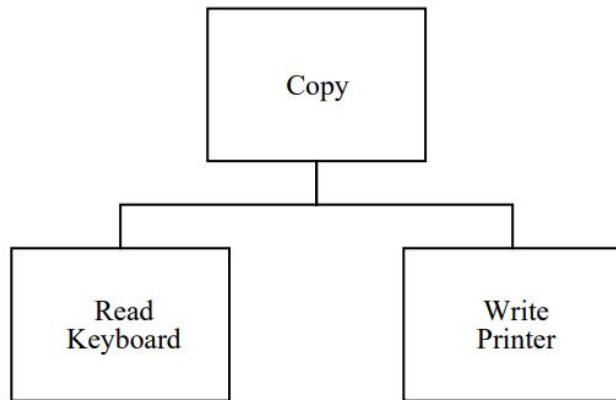
Depend on abstractions, not on concretions.



### DEPENDENCY INVERSION PRINCIPLE

Would You Solder A Lamp Directly To The Electrical Wiring In A Wall?

## SOLID > Dependency Inversion



### **Listing 1. The Copy Program**

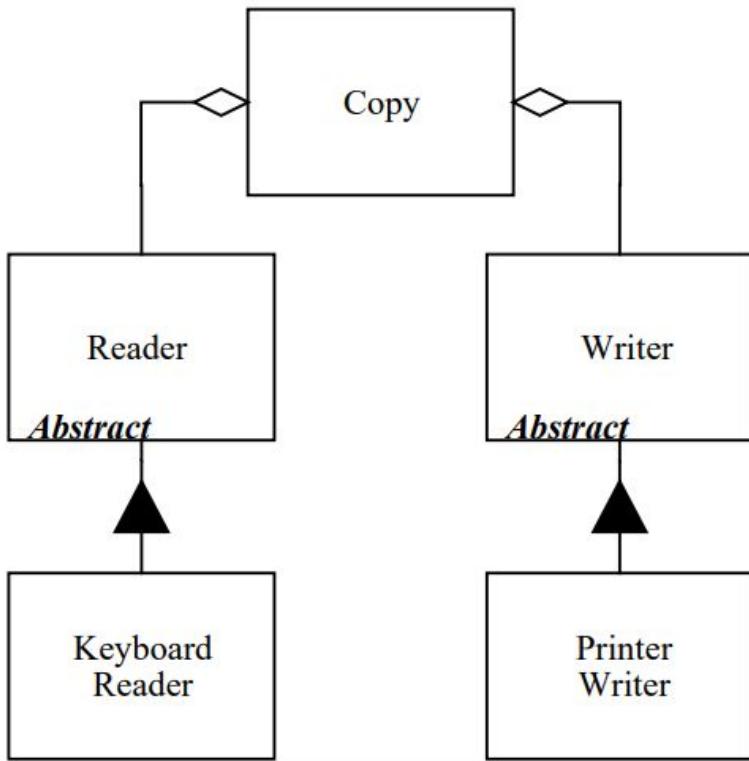
```
void Copy()
{
    int c;
    while ((c = ReadKeyboard()) != EOF)
        WritePrinter(c);
}
```

### **Listing 2. The “Enhanced” Copy Program**

```
enum OutputDevice {printer, disk};  
void Copy(outputDevice dev)  
{  
    int c;  
    while ((c = ReadKeyboard()) != EOF)  
        if (dev == printer)  
            WritePrinter(c);  
        else  
            WriteDisk(c);  
}
```

## SOLID > Dependency Inversion

**Figure 2: The OO Copy Program**



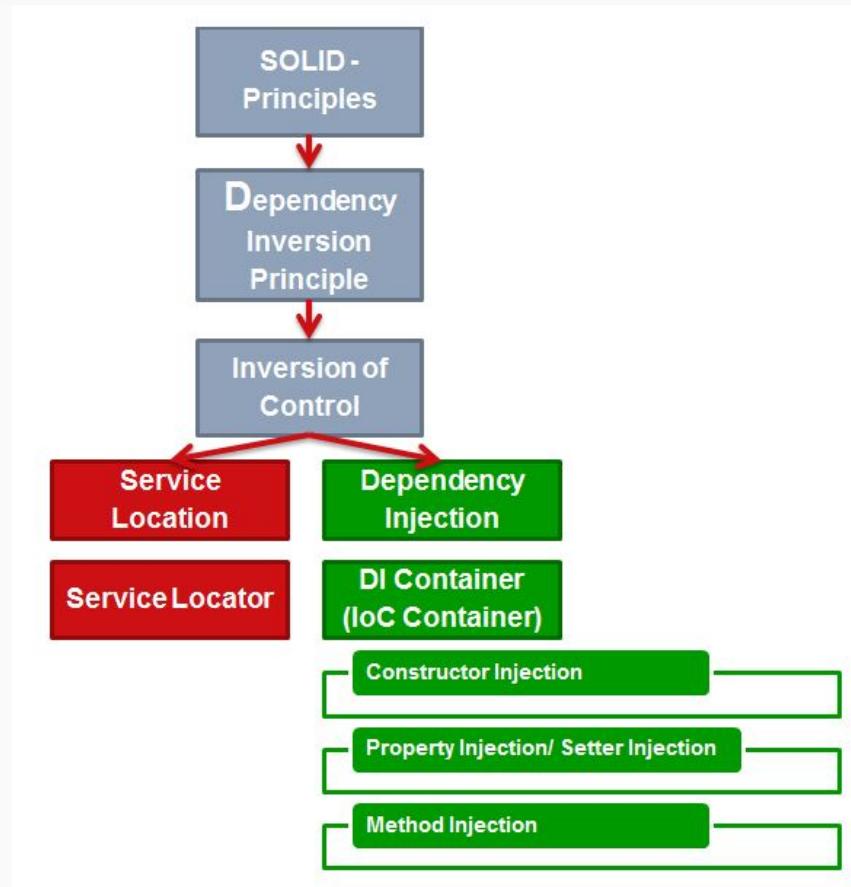
**Listing 3: The OO Copy Program**

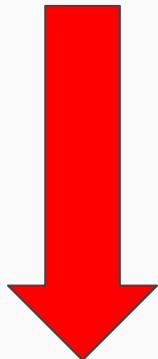
```
class Reader
{
public:
    virtual int Read() = 0;
};

class Writer
{
public:
    virtual void Write(char) = 0;
};

void Copy(Reader& r, Writer& w)
{
    int c;
    while((c=r.Read()) != EOF)
        w.Write(c);
}
```

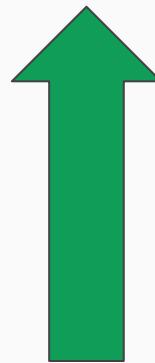
# SOLID > Dependency Inversion





Gắn “chết”

(Coupling)

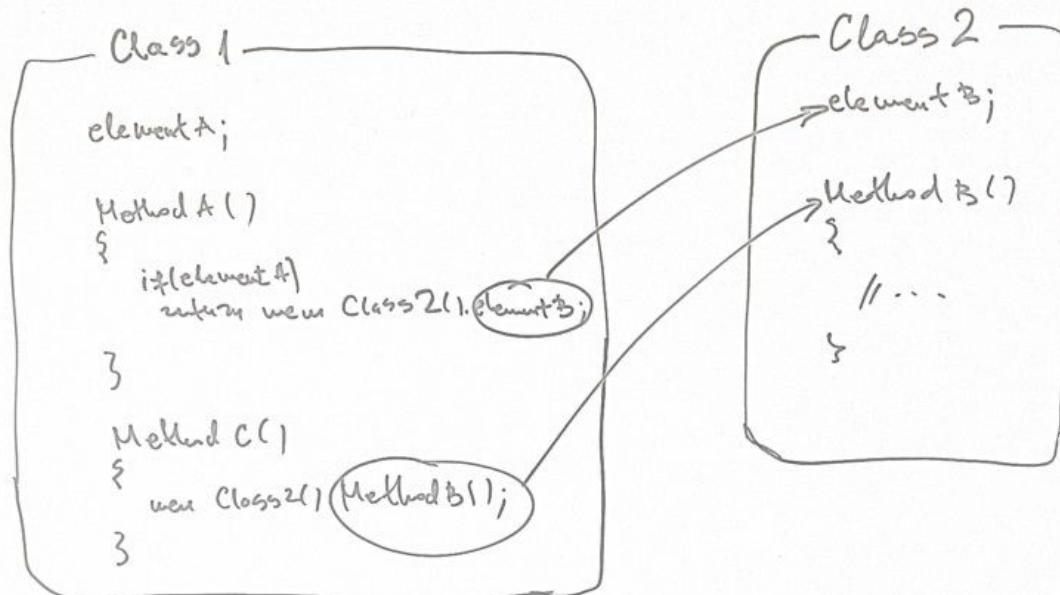


Gắn kết

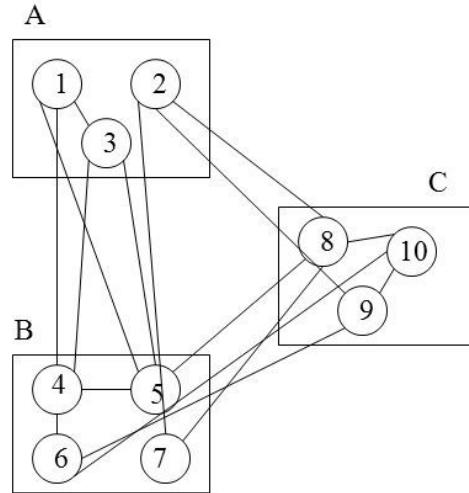
(Cohesion)

# Coupling | Cohesion

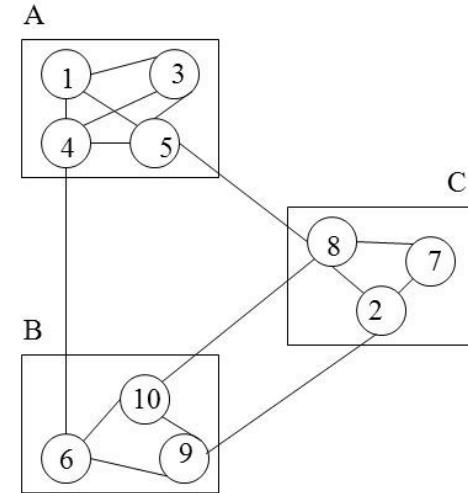
## TIGHT COUPLING



## Cohesion and Coupling



Bad modularization:  
low cohesion, high coupling



Good modularization:  
high cohesion, low coupling

## Advices

- A class should have only one reason to change.
- Favor Composition over inheritance
- Program to interfaces, not implementations.
- DRY Principle: Don't Repeat Yourself

# Design Pattern

**Design patterns** are typical solutions to common problems in software **design**

## THE 23 GANG OF FOUR DESIGN PATTERNS

C Abstract Factory

S Adapter

S Bridge

C Builder

B Chain of Responsibility

B Command

S Composite

S Decorator

S Facade

C Factory Method

S Flyweight

B Interpreter

B Iterator

B Mediator

B Memento

C Prototype

S Proxy

B Observer

C Singleton

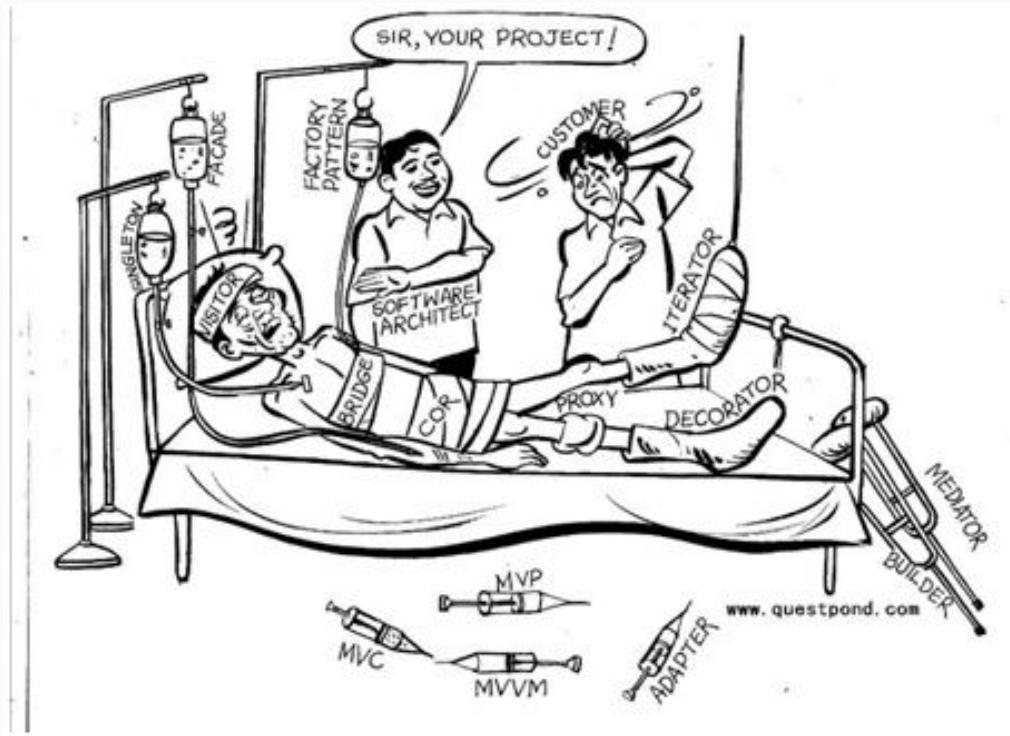
B State

B Strategy

B Template Method

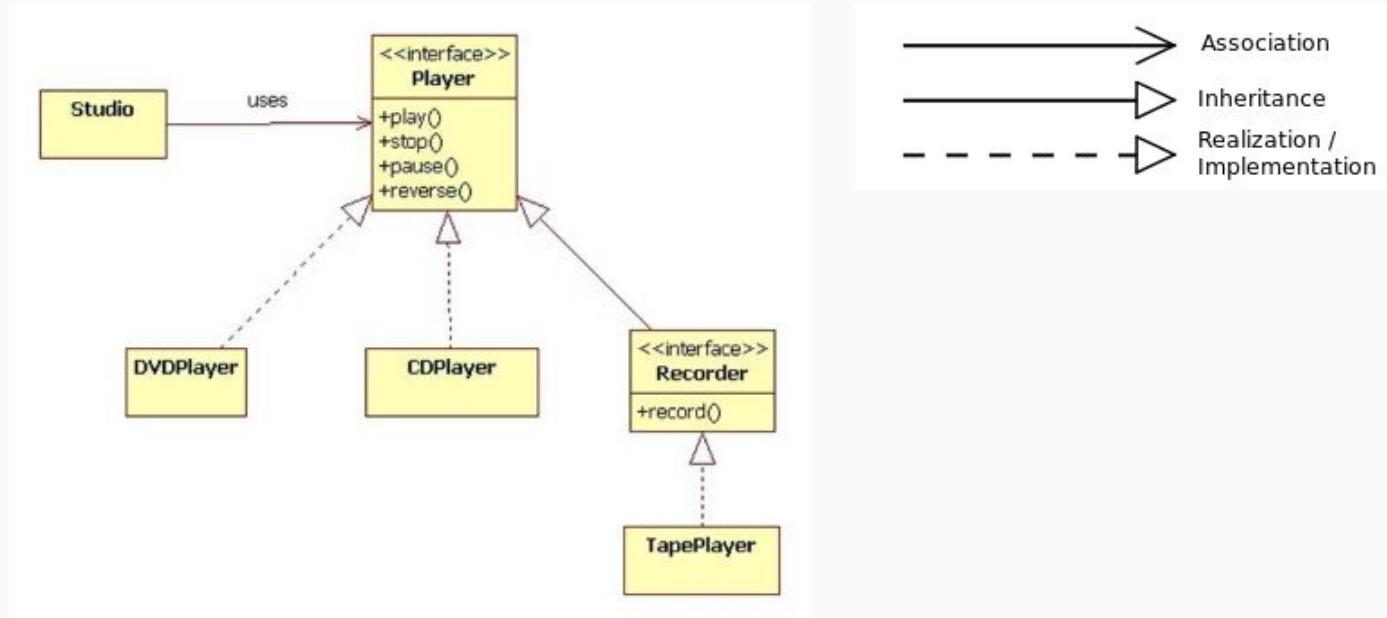
B Visitor

# Design Pattern



<https://gist.github.com/lolzballs/2152bc0f31ee0286b722>

## Design Pattern > Diagram



## Decorator Pattern



WHY?

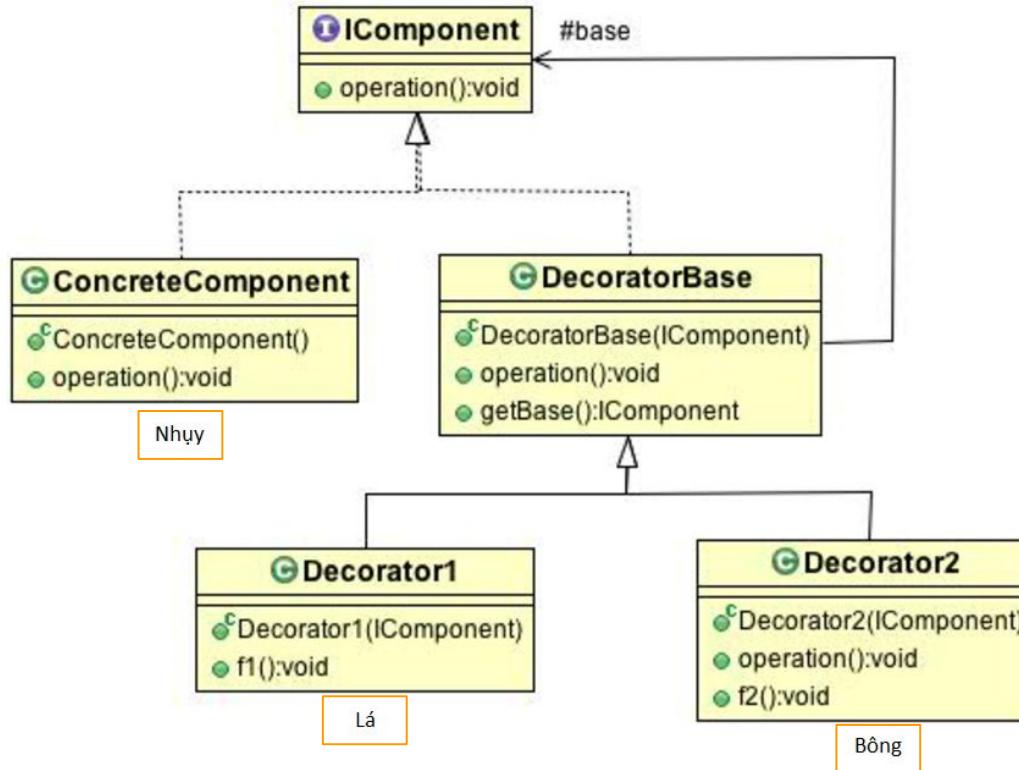
## Decorator Pattern

Attach additional responsibility dynamically during runtime.



*Trong đầm gì đẹp bằng sen  
Lá xanh bông trắng lại chen nhụy vàng*

# Decorator Pattern



## Decorator Pattern > Example



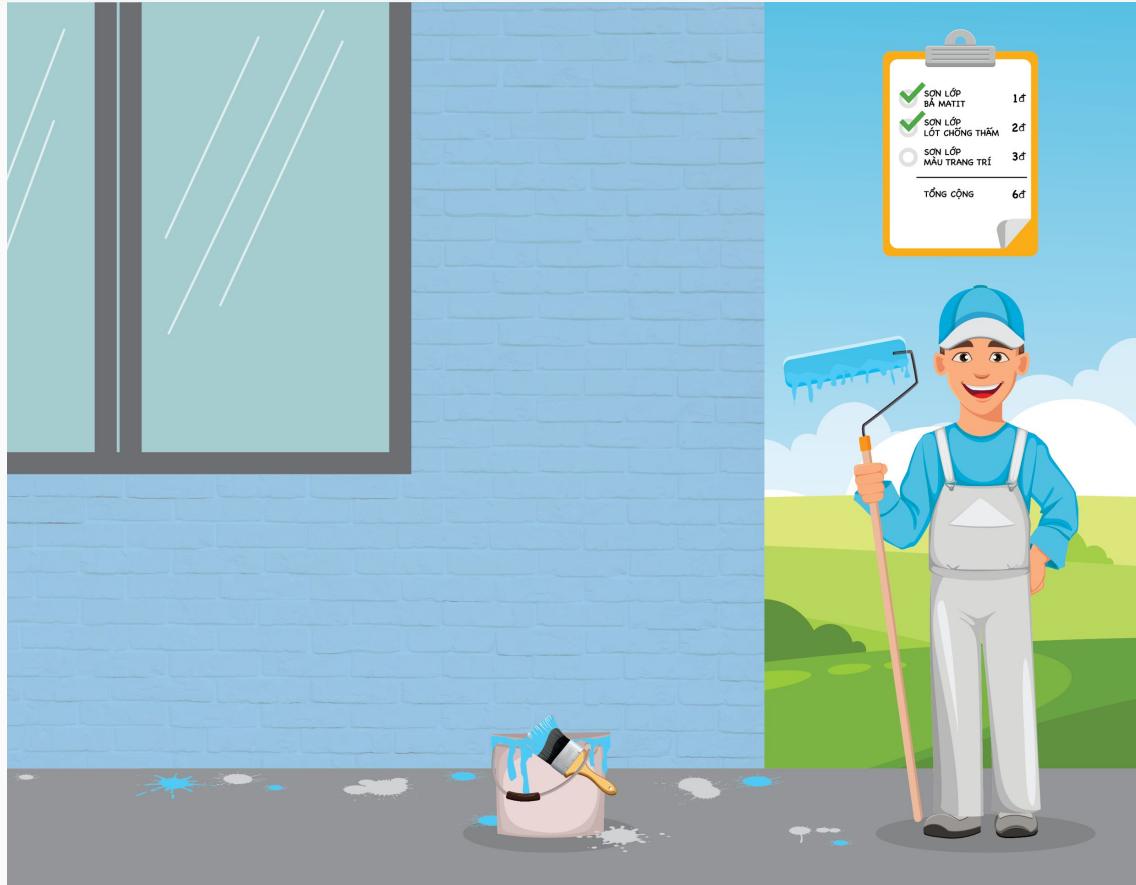
## Decorator Pattern > Example



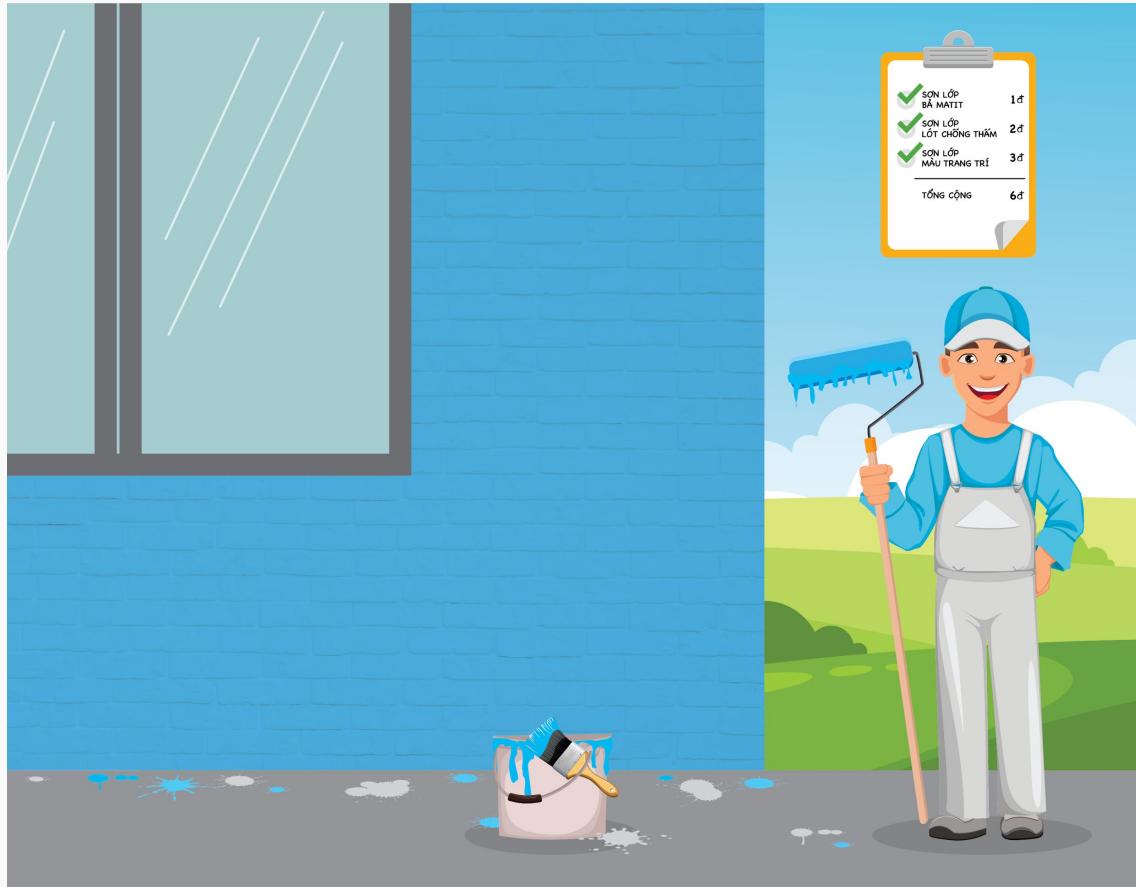
## Decorator Pattern > Example



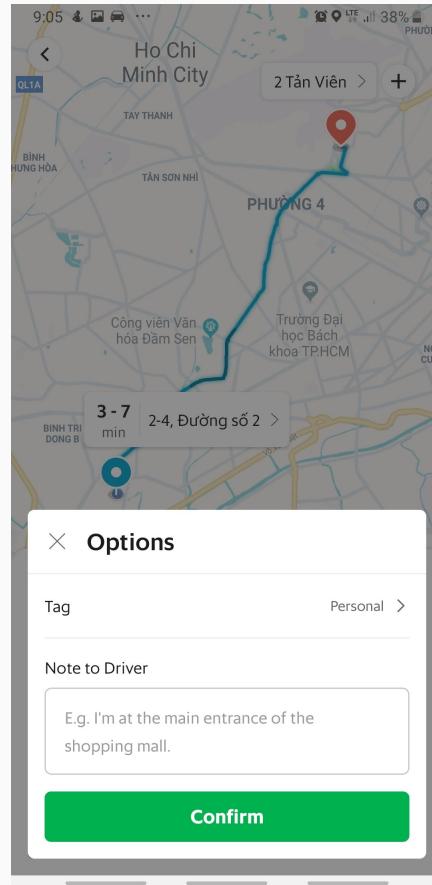
## Decorator Pattern > Example



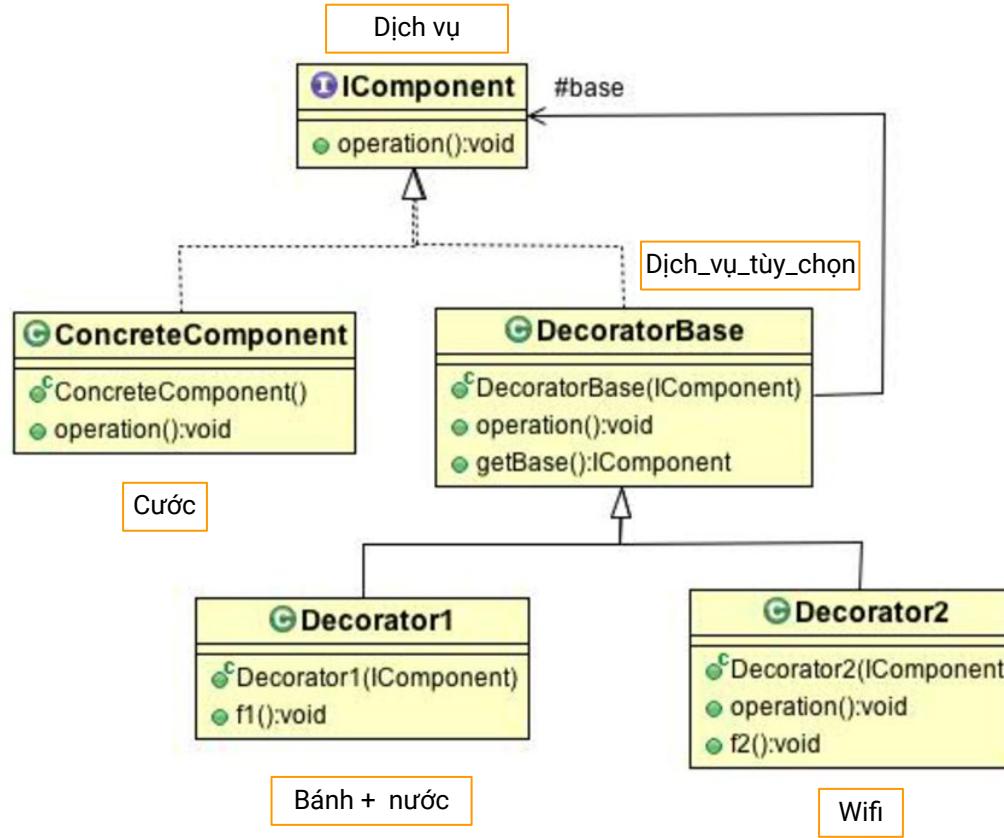
## Decorator Pattern > Example



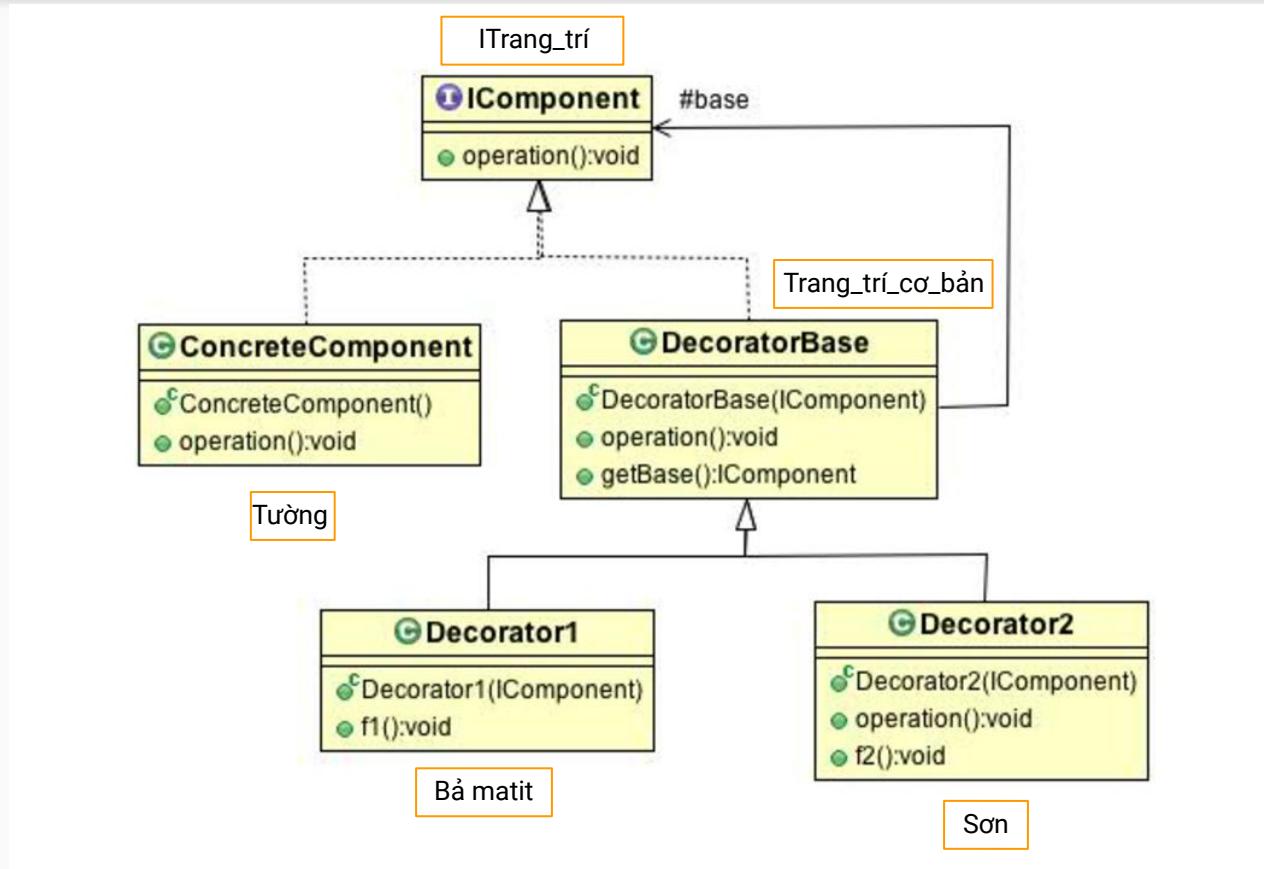
## Decorator Pattern > Real Example



## Decorator Pattern > Real Example



# Decorator Pattern

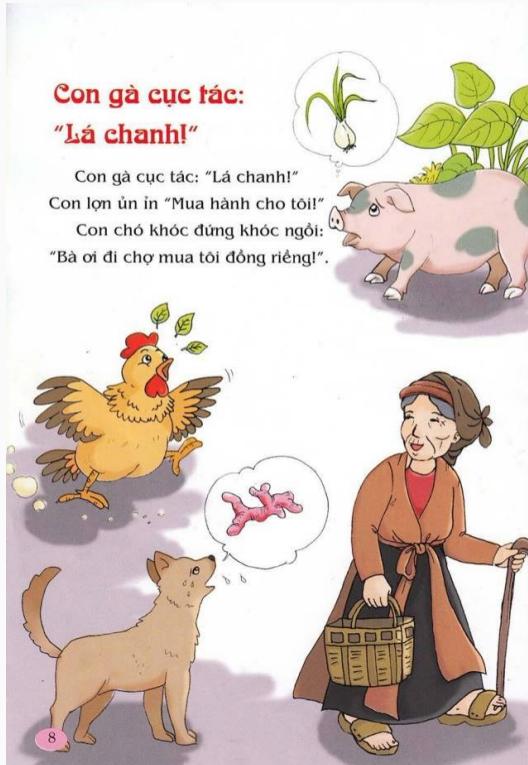


## Strategy Pattern

A graphic featuring four yellow sticky notes arranged in a staggered, overlapping fashion. Each note has a large, bold, black question mark or exclamation mark on it. The notes are tilted at different angles, creating a sense of depth. The background is white.

# Strategy Pattern

Define a family of algorithms, encapsulate each one, and make them interchangeable.



**Con gà cục tác:**

**"Lá chanh!"**

Con gà cục tác: "Lá chanh!"

Con lợn ủn ỉn "Mua hành cho tôi!"

Con chó khóc đứng khóc ngồi:

"Bà ơi đi chợ mua tôi đồng riêng!".

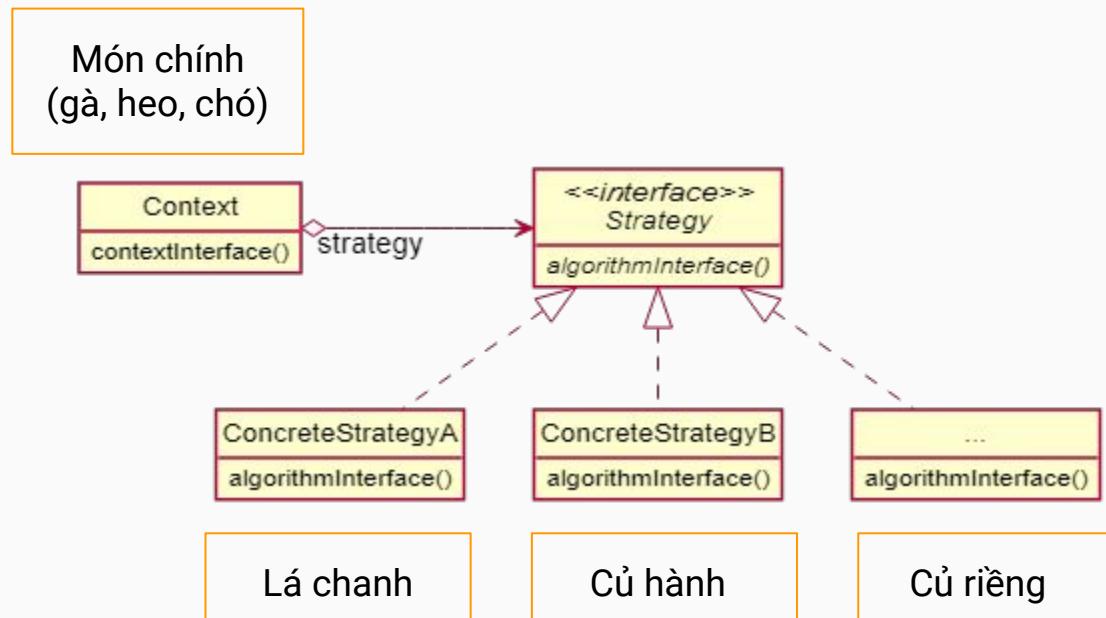
*Con gà cục tác: "lá chanh"*

*Con lợn ủn ỉn "Mua hành cho tôi"*

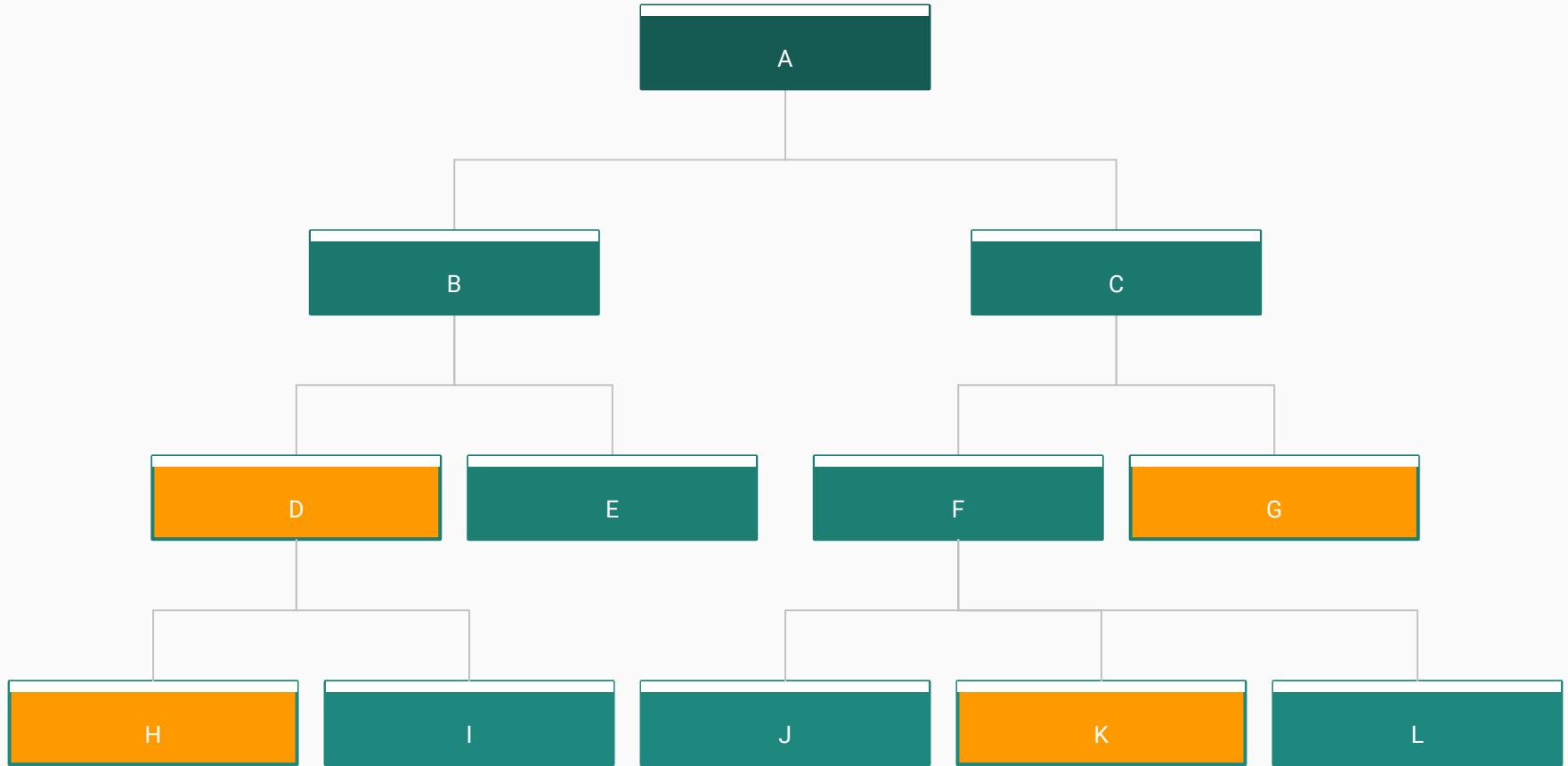
*Con chó khóc đứng khóc ngồi:*

*"Bà ơi đi chợ mua tôi đồng riêng".*

## Strategy Pattern > Diagram



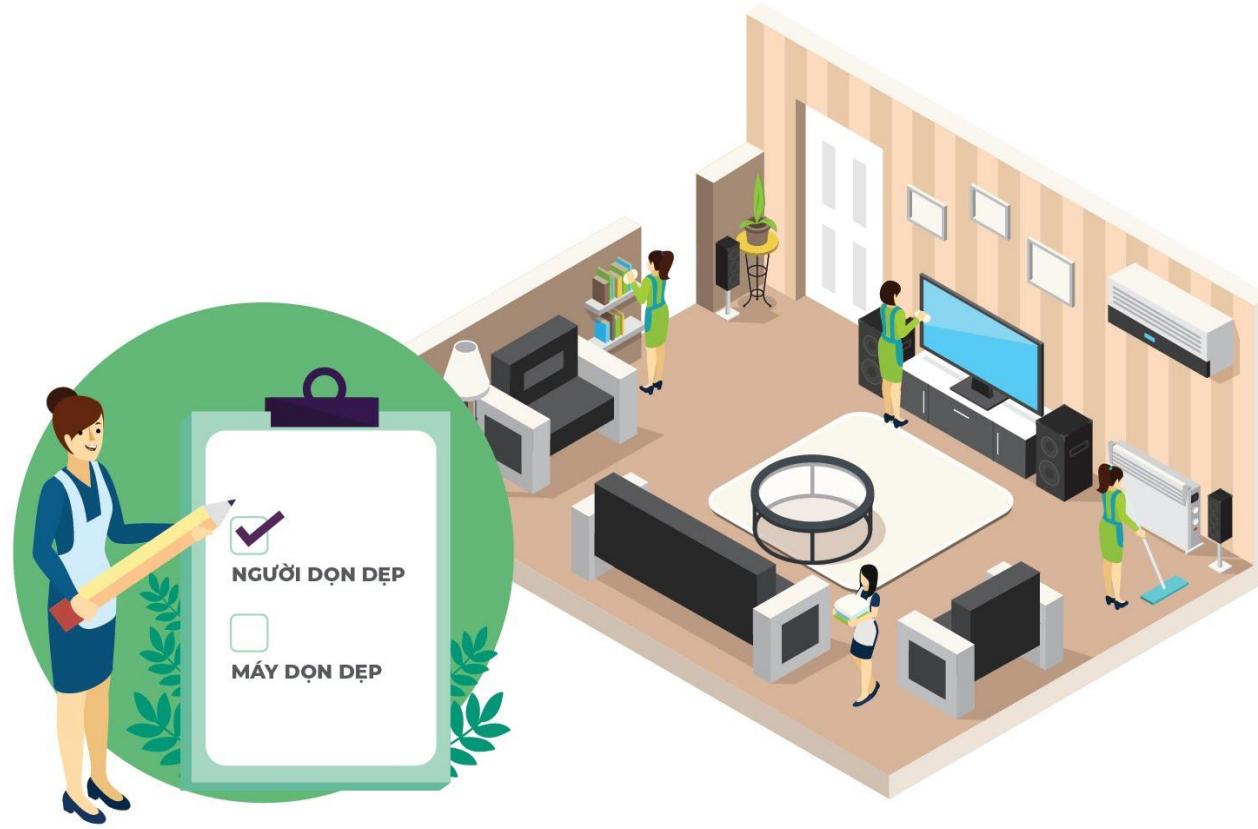
# Strategy Pattern



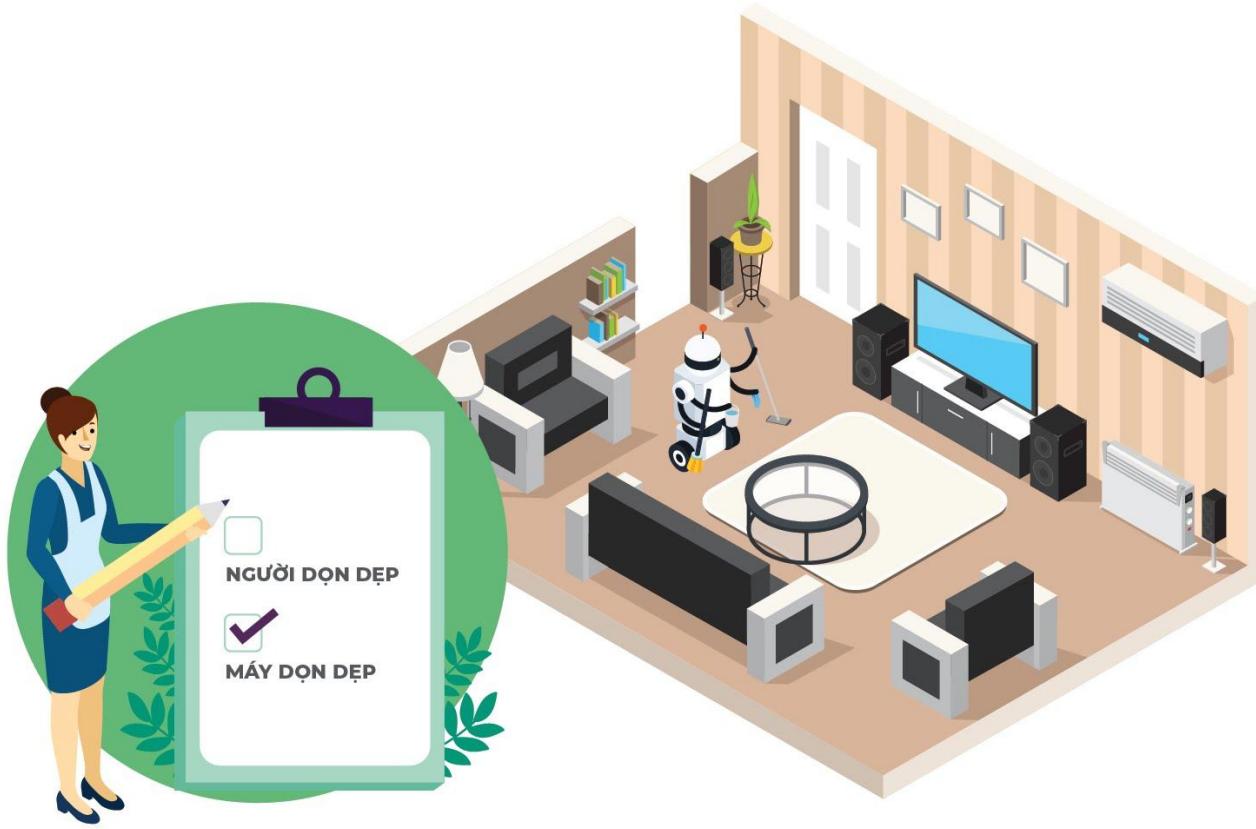
## Strategy Pattern > Example



## Strategy Pattern > Example



## Strategy Pattern > Example



# Strategy Pattern > Real Example

## 2. Thanh toán bằng Tiki Xu



Sử dụng 5.912 Xu (ứng với 5.912 ₫) để thanh toán.

Bạn có Phiếu quà tặng Tiki/Got It/UrBox muốn đổi thành Tiki Xu? [Nhập tại đây](#)

## 3. Chọn hình thức thanh toán



Thanh toán tiền mặt khi nhận hàng

Rất tiếc, trong giỏ hàng của bạn có sản phẩm nhập khẩu không hỗ trợ thanh toán bằng tiền mặt.



Thanh toán bằng thẻ quốc tế Visa, Master, JCB



Thẻ ATM nội địa/Internet Banking (Miễn phí thanh toán)



Thanh toán bằng ví MoMo - [Chi tiết](#)

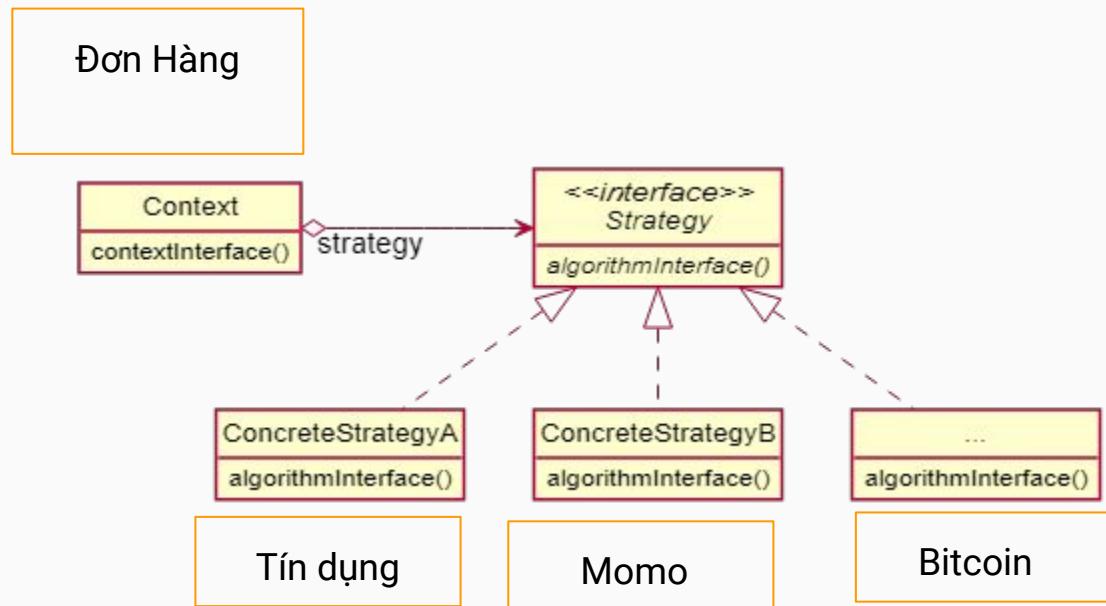
Nhập mã MOMOBH giảm 50K cho ĐH Làm đẹp-Sức khỏe từ 300K. Nhập mã MOMOFA giảm 10% tối đa 100K cho ĐH Thời trang



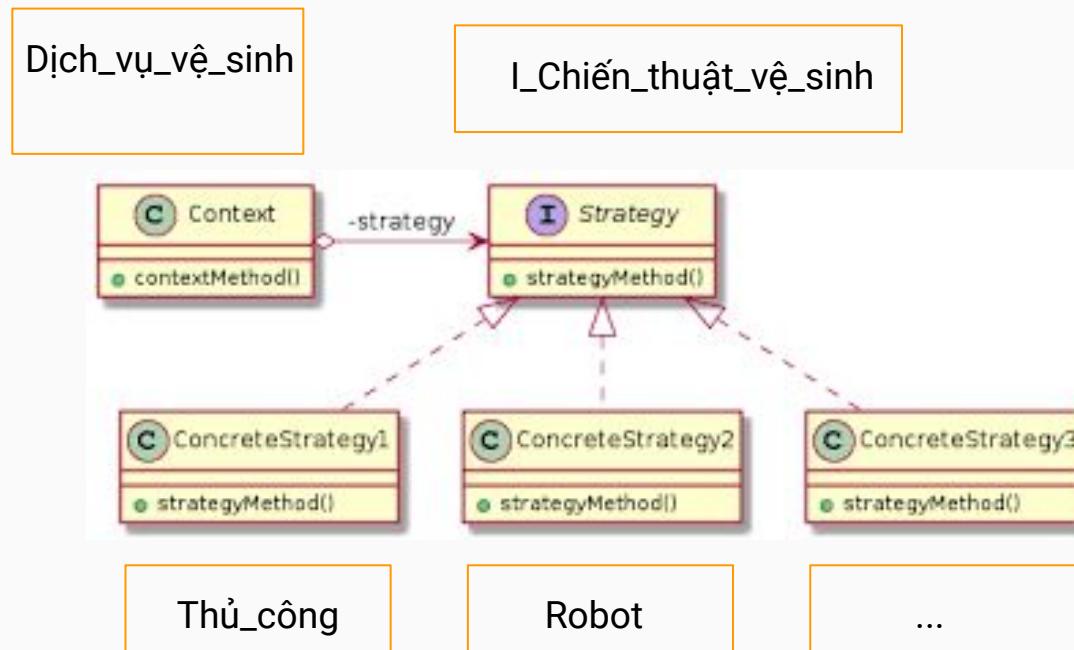
ưu đãi Thẻ quốc tế SCB giảm 100K cho đơn hàng từ 800K - [Chú ý](#)

Chưa đến thời gian tham gia chương trình hoặc số lượng đơn hàng ưu đãi đã hết!

## Strategy Pattern > Real Example



## Strategy Pattern > Diagram



## Visitor Pattern



WHY?

The image shows four yellow sticky notes arranged in a staggered, overlapping fashion. Each note contains one letter of the word "WHY?". The 'W' is on the leftmost note, 'H' is on the second note from the left, 'Y' is on the third note from the left, and a question mark is on the rightmost note. The letters are written in a bold, black, sans-serif font.

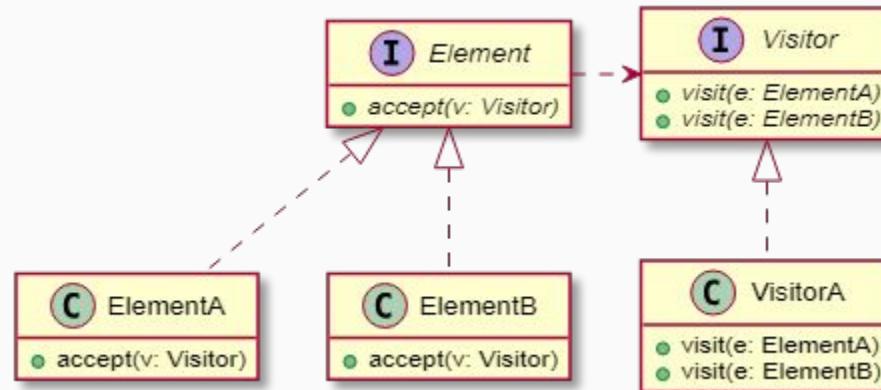
## Visitor Pattern

A way of separating an algorithm from an object structure on which it operates. Visitor lets you add new operations to existent object structures without modifying the structures.



Dᾶn lang nhập tht

# Visitor Pattern

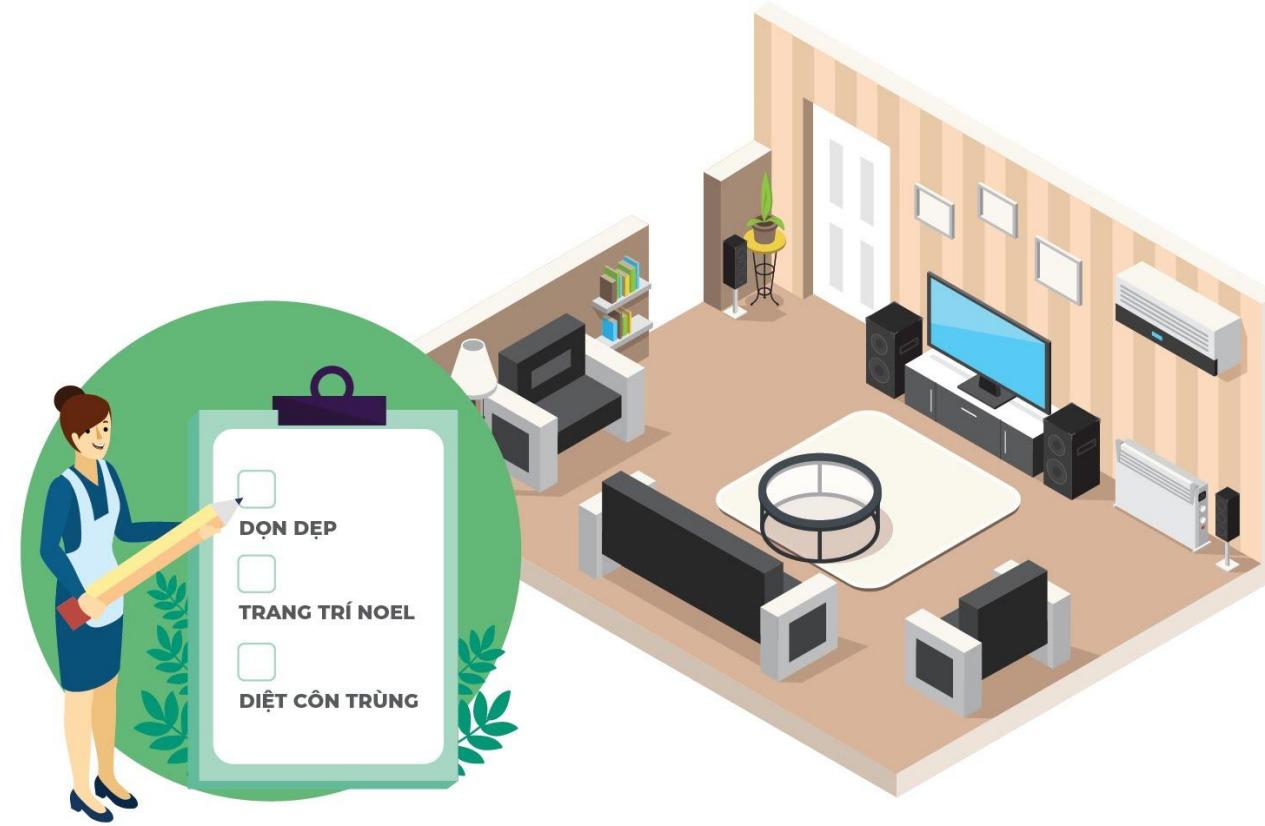


Phòng khách

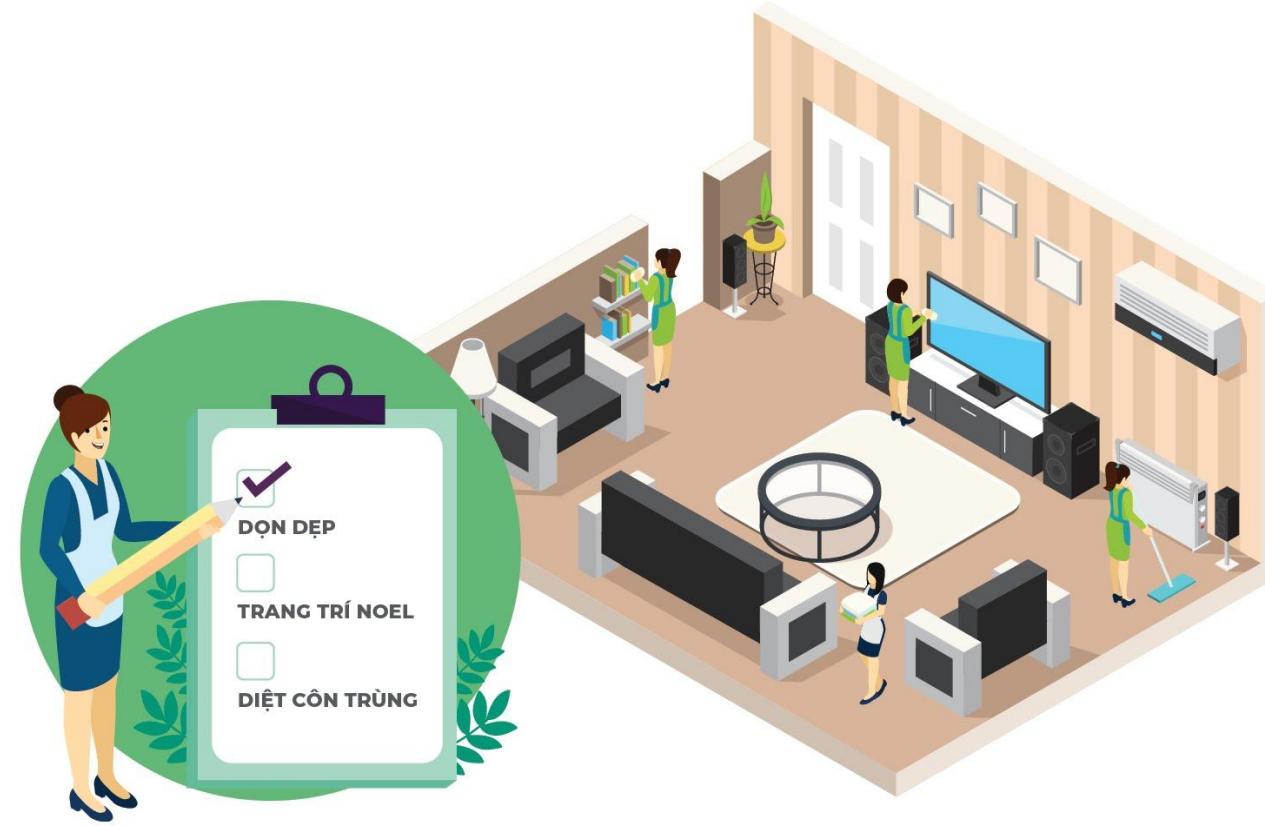
Nhà bếp

Sói

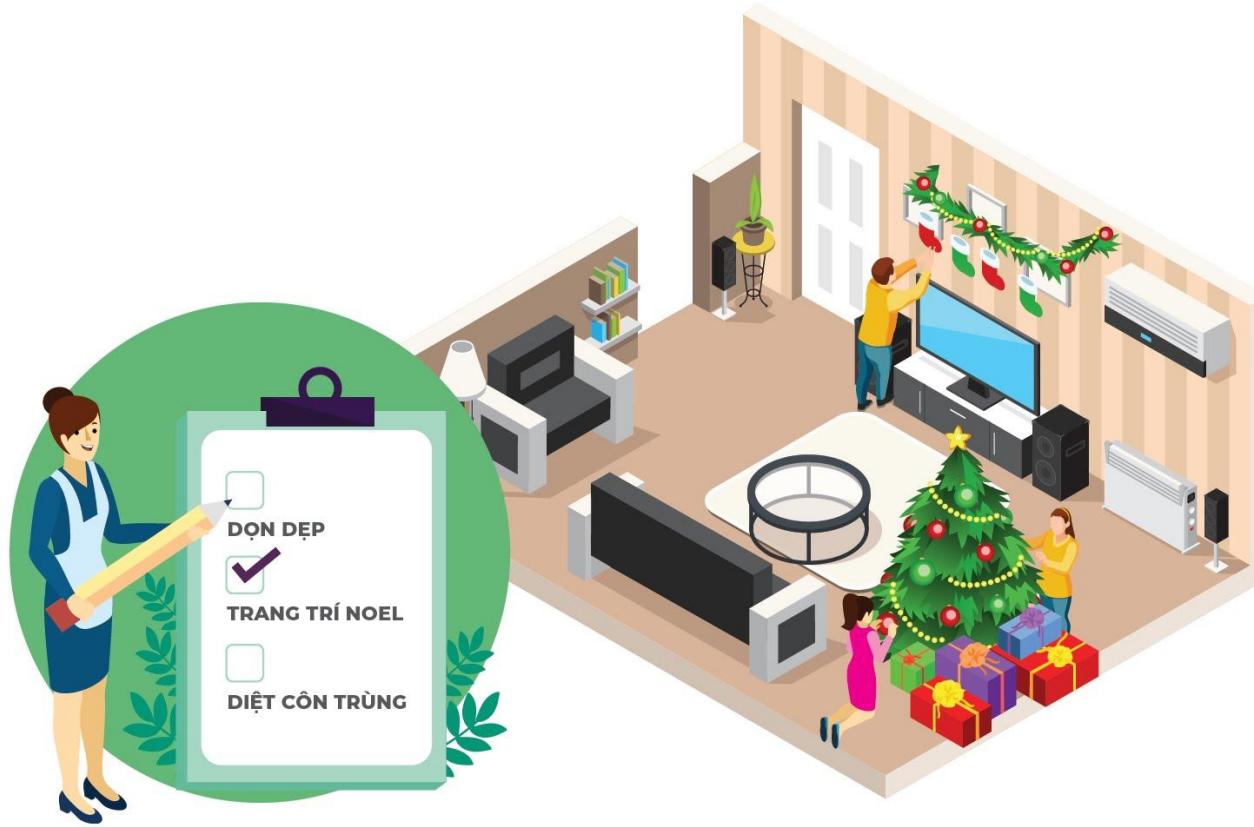
## Visitor Pattern > Example



## Visitor Pattern > Example



## Visitor Pattern > Example



## Visitor Pattern > Example



# Visitor Pattern > Real Example

Trang chủ > Điện Tử - Điện Lạnh > Tivi > Tivi 4K > Smart Tivi LG 43 inch 4K UHD 43UJ750T



Rê chuột lên hình để phóng to

Hình ảnh thực tế từ khách hàng



Xem thêm 38 hình

**TIKINOW | Smart Tivi LG 43 inch 4K UHD 43UJ750T**

★★★★★ (Xem 97 đánh giá)

Đứng thứ 91 trong Top 100 Điện Tử - Điện Lạnh bán chạy tháng này

Thương hiệu: LG SKU: 8901522746778

**8.590.000 đ**

Trả góp 0% chỉ 715.833 đ/tháng trong 12 tháng qua thẻ tín dụng (?)

Tiết kiệm: 51% (8.810.000 đ)

Giá thị trường: 17.400.000 đ

- Loại TV: Smart TV 43 inch
- Độ phân giải: 4K UHD
- Tần số quét: 50Hz
- Hệ điều hành: WebOS 3.5
- Các ứng dụng sẵn có: Trình duyệt web, YouTube, Netflix, Zing TV, VnExpress, Film+
- Các ứng dụng phổ biến có thể tải thêm: FPT Play, Clip.vn, The Karaoke Chanel
- Công nghệ xử lý hình ảnh: ULTRA Luminance, Nano Cell Display, Active HDR with Dolby Vision, HDR Effect, Local Dimming, Nâng cấp hình ảnh gần chuẩn 4K, Tấm nền IPS
- Điều khiển tivi bằng điện thoại: Bằng ứng dụng LG TV Plus

Số lượng:

**CHỌN MUA**



**Tiki Trading**  
Cam kết chính hiệu 100%

**Tiki hoàn tiền 111%**   
Nếu phát hiện hàng giả

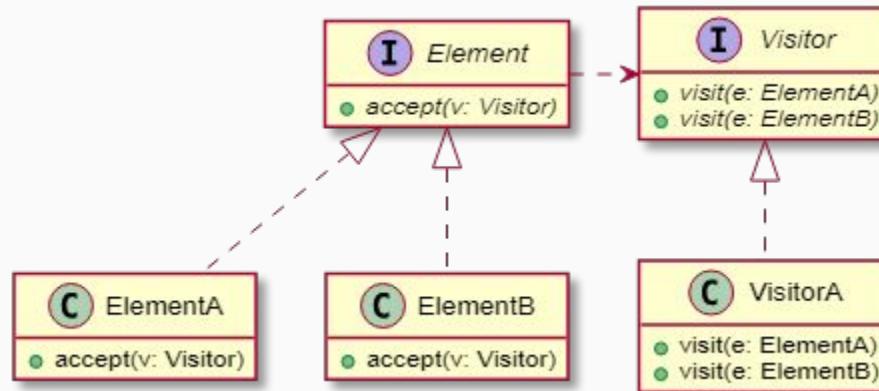
**Thông tin bảo hành**  
Điện tử (Bảo hành chính hãng tận nhà 24 tháng)  
[Chi tiết](#)

**Liên hệ**  
Hotline đặt hàng 1800 6963  
(Miễn phí, 8-21h cả T7, CN)

**Email:** hotro@tiki.vn

Bạn muốn bán hàng cùng Tiki? [Đăng ký](#)

## Visitor Pattern > Real Example

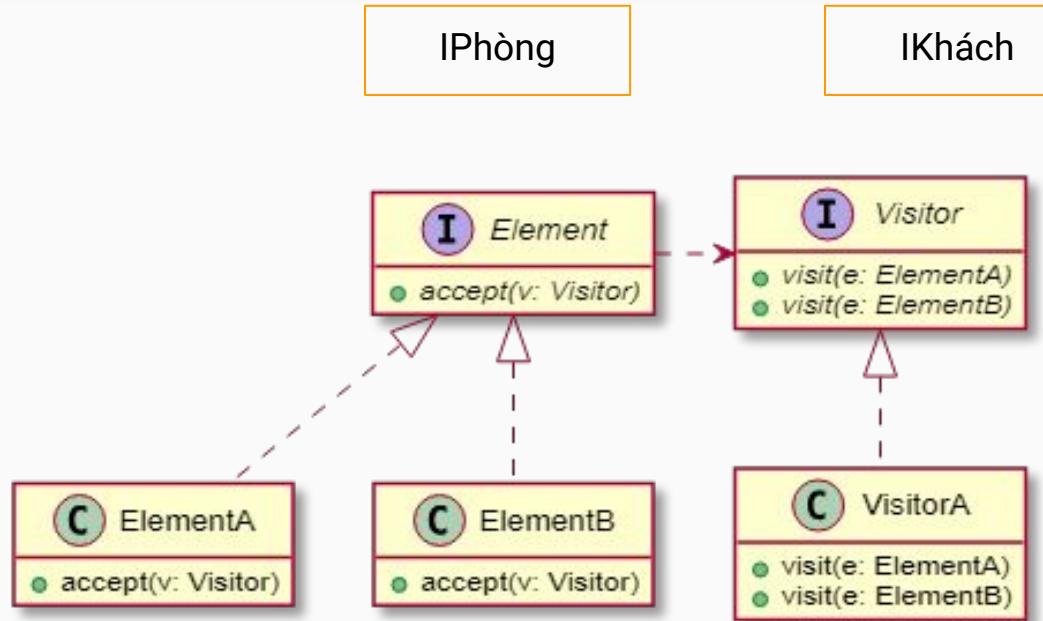


Mô tả SP

Bình luận

Đánh giá SP

# Visitor Pattern



Phòng khách

Nhà bếp

Đội trang trí Noel

# Recap

- SOLID
- Design patterns
  - Decorator
  - Strategy
  - Visitor
- Design patterns & ca dao tục ngữ
- Caution
  - Use the right tool for the job

Example on Github

<https://github.com/khoiquyenvn/designpattern>



# Q & A

# Thanks!

Contact us:

KMS Technology  
2 Tân Viên, P.2, Q.Tân Bình

[khoimnguyen@kms-technology.com](mailto:khoimnguyen@kms-technology.com)  
[www.kms-technology.com](http://www.kms-technology.com)

