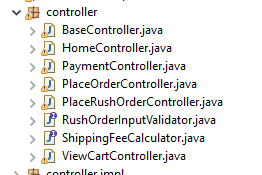
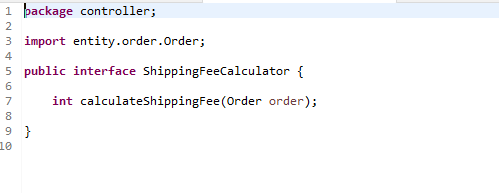
# Các yêu cầu mở rộng

## Thay đổi phí ship

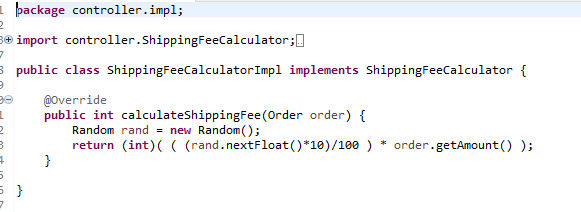
Giải pháp: ta sẽ tạo ra một interface ShippingFeeCalculator với phương thức trừu tượng là calculateShippingFee. Khi chúng ta muốn thêm một hay nhiều cách tính phí ship thì chúng ta chỉ cần viết thêm một lớp mới implement interface trên. Và tại PlaceOrderController chúng ta khởi tạo một đối tượng là interface ShippingFeeCalculator với instance là loại chiến lược tính phí ship mà chúng ta muốn.



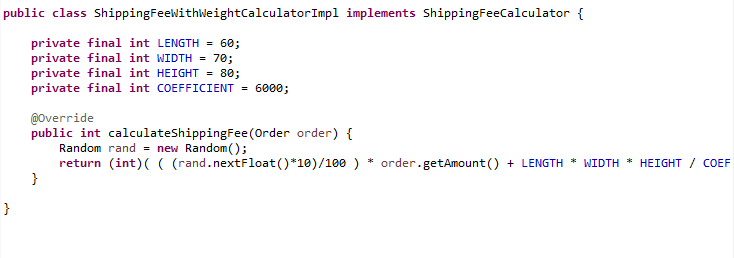
Interface ShippingFeeCalculator



Class ShippingFeeCalculatorImpl implement ShippingFeesInterface:

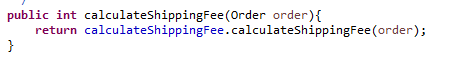


Class Class ShippingFeeWithWeightCalculatorImpl implement ShippingFeesInterface:



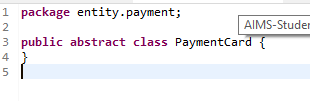
Gọi đến phương thức tính ship



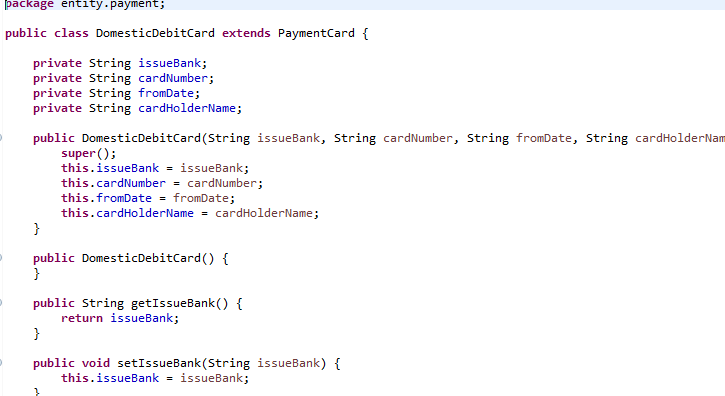


## Thay đổi cách tính tiền mới

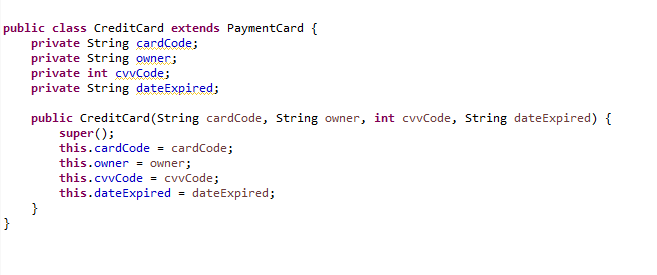
Tạo abstract class PaymentCard



Entity class DomesticDebitCard extends PaymentCard



Thay đổi tương tự với CreditCard



# Bài tập

## Coupling và Cohesion

### Coupling

#### Content coupling

*Không có*

#### Control coupling

*Không có*

#### Common coupling

*Không có*

#### Stamp coupling

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | RushInfoScreenHandler | Hàm calculateAdditionalFees nhận tham số là Order. Điều này là k cần thiết vì nó chỉ dùng phương thức setAdditionalFees và setRushInfo | Có thể cài đặt Interface cho những phương thức cần dùng và implement ở một lớp control khác |

#### Data coupling

*Các coupling về data là chấp nhận được*

### Cohesion

#### Coincidental Cohesion

*Không có*

#### Logical Cohesion

*Không có*

#### Temporal Cohesion

*Không có*

#### Procedural Cohesion

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | PlaceOrderController | Các phương thức như validateAddress, validateName, validatePhone được gọi lần lượt chứ không liên quan về mặt chứ năng | Đưa các phương thức này vào một module nghiệp vụ chung (Một lớp hoặc một package) có nhiệm vụ validate các thông tin trên |
| 2 | PlaceRushOrderController | Các phương thức như validateDeliveryDate và validateProvince, cũng gặp vấn đề tương tự như trên | Đưa các phương thức này vào một module nghiệp vụ chung (Một lớp hoặc một package) có nhiệm vụ validate các thông tin trên |

#### Communicational Cohesion

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | InterbankSubsystemController | Phương thức payOrder và refund được đặt trong cùng module vì trả về cùng kiểu dữ liệu PaymentTransaction chứ chức năng của chúng tương đối khác nhau | Tách module này thành các module nhỏ hơn để cung cấp phương thức payOrder và refund riêng cho lớp InterbankSubsystem |

#### Coincidental Cohesion

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | Các lớp Entity như Cart, Invoice, Media, CartMedia và Order, cùng các ScreenHandler có liên quan | Các ScreenHandler cùng thao tác trên các attribute của các lớp | Không có |

#### Coincidental Cohesion

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | API | Phương thức setUpConnection chỉ làm nhiệm vụ tạo connection đến server để gọi sau đó trong get và post | Có thể tạo một module riêng, làm nhiệm vụ kết nối tới API server |

## Nguyên lí thiết kế SOLID

### Single Responsibility Principle

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | InterbankSubsystemController | Thực hiện cả công việc mapping thông tin từ thẻ, gọi API giao dịch và trả về kết quả | Tách ra 2 module con, thực hiện việc mapping thông tin thẻ và trả kết quả, còn controller chỉ làm nhiệm vụ gọi API giao dịch |

### Open/Closed Principle

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | DebitCard, NationalBankSubsystem, InterbankInterface | Sau khi cài đặt yêu cầu mở rộng thì đã làm chưa tốt khi thêm cứng Entity DebitCard và thiết kế Interface chưa hợp lí | Tạo Entity thẻ thanh toán chung, để làm tham số trong các phương thức trong InterbankInterface, còn DebitCard, CreditCard thì cho kế thừa loại thẻ chung này |
| 2 | PlaceRushOrderController | Ban đầu cài đặt riêng lẻ, nên khi dùng phải gọi lại nhiều phương thức, và sử dụng chung attribute đã cài đặt trong PlaceOrderController | Kế thừa lại lớp PlaceOrderController để tận dụng các hàm có sẵn trong luồng chạy |

### Liskov Substitution

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | Các lớp Controller | Đã kế thừa BaseController và hoạt động tốt, không gây ra lỗi | Không có |
| 2 | Các lớp CD, DVD, Book | Kế thừa lớp Media, tận dụng được các thuộc tính chung và tính đa hình | Không có |
| 3 | PlaceRushOrderController | Kế thừa PlaceOrderController nhưng không thể hoạt động riêng lẻ được mà vẫn phụ thuộc vào lớp cha | Có thể xem xét PlaceRushOrder là một use case riêng hoàn toàn, để xây dựng module điều khiển riêng, không cho extend từ PlaceOrderController nữa |

### Interface Segregation

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | Các Interface | Đều nhỏ gọn, thực hiện đúng đủ chức năng và không gây dư thừa | Không có |

### Dependency Inversion

|  |  |  |  |
| --- | --- | --- | --- |
| # | Related modules | Description | Improvement |
| 1 | Các Interface | Đều nhỏ gọn, thực hiện đúng đủ chức năng và không gây dư thừa | Không có |
| 2 | Các lớp ScreenHandler liên quan đến Order | Các lớp này đang phụ thuộc chặt chẽ vào Entity Order, nếu có thay đổi về dữ liệu của Order thì các lớp này phải sửa theo | Tạo lớp OrderAbstract hoặc OrderInterface, và các lớp ScreenHandler chỉ tương tác với lớp này để tạo sự phụ thuộc ngược lại của Entity vào lớp giao diện |