

BÁO CÁO THỰC HÀNH LAB 2

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Table of Contents

Table of Figures	1
1. Bài toán đặt ra.....	2
2. Yêu cầu hệ thống.....	2
2.1. Đối với Customer:	2
2.2. Đối với Store Manager:	2
3. Use Case Diagram	3
4. Class Diagram	4
5. Source Code	5
5.1. Aims Class.....	5
5.2. DigitalVideoDisc Class	6
5.3. Cart Class	8
6. Kết quả demo.....	10
7. Reading Assignment.....	12
8. Answer the question	12

Table of Figures

Figure 1: Use Case Diagram.....	3
Figure 2: Class Diagram	4
Figure 3: Aims Class.....	5
Figure 4: DigitalVideoDisc Class	6
Figure 5: Cart Class	8
Figure 6: Result	10
Figure 7: Getter and Setter Methods.....	12

1. Bài toán đặt ra

Có thể trong tương lai, Tiki và Sendo sẽ đàm phán về việc sáp nhập để cạnh tranh với các nền tảng thương mại điện tử khác, đặc biệt là những nền tảng có nhà đầu tư nước ngoài. Việc sáp nhập này sẽ tạo ra một công ty mới có tên là *Ti-do*, với "Ti" từ Tiki và "do" từ Sendo, mang ý nghĩa một công ty tỷ đô trong tiếng Việt. Công ty Ti-do muốn bạn giúp họ tạo ra một hệ thống hoàn toàn mới cho dự án AIMS (*An Internet Media Store*). Hiện tại, chỉ có một loại phương tiện truyền thông: đĩa video kỹ thuật số (DVD).

2. Yêu cầu hệ thống

2.1. Đối với Customer:

Khách hàng có thể duyệt qua danh sách các DVD có sẵn trong cửa hàng, thứ tự hiển thị dựa trên ngày thêm vào, từ mới nhất đến cũ nhất. Khi khách hàng muốn tìm kiếm các DVD để thêm vào giỏ hàng, họ có thể chọn một trong ba tùy chọn tìm kiếm. Phần mềm sẽ hiển thị danh sách các kết quả phù hợp (DVD mới nhất sẽ được ưu tiên hiển thị) với đầy đủ thông tin của chúng. Khách hàng cũng có thể chọn phát một DVD cụ thể. Phần mềm sẽ phát một đoạn DVD (phần demo). Nếu một DVD có độ dài 0 hoặc nhỏ hơn, hệ thống phải thông báo cho khách hàng rằng không thể phát DVD đó.

- **Khi tìm kiếm theo tiêu đề DVD**, khách hàng nhập vào một chuỗi từ khóa. Nếu tiêu đề của DVD có chứa bất kỳ từ nào trong chuỗi từ khóa, nó sẽ được tính là một kết quả phù hợp. Lưu ý rằng việc so sánh từ khóa này không phân biệt chữ hoa, chữ thường.
- **Khi tìm kiếm theo thể loại**, khách hàng nhập vào tên thể loại. Nếu bất kỳ DVD nào có thể loại trùng khớp (không phân biệt chữ hoa, chữ thường), nó sẽ được tính là một kết quả phù hợp.
- **Khi tìm kiếm theo giá**, khách hàng có thể cung cấp cả giá trị tối thiểu và tối đa hoặc chỉ cung cấp giá trị tối đa.

Khách hàng có thể xem thông tin chi tiết của DVD từ danh sách DVD. Họ có thể thêm DVD vào giỏ hàng từ danh sách DVD hoặc từ màn hình chi tiết.

Khi khách hàng muốn xem giỏ hàng hiện tại, hệ thống sẽ hiển thị tất cả thông tin của các DVD, cùng với tổng chi phí. Khách hàng có thể nghe một đoạn DVD (phần demo) trong giỏ hàng trước khi xác nhận đặt hàng. Khách hàng có thể sắp xếp tất cả DVD trong giỏ hàng theo tiêu đề hoặc theo giá:

- **Sắp xếp theo tiêu đề**: hệ thống hiển thị tất cả các DVD theo thứ tự chữ cái của tiêu đề. Nếu có cùng tiêu đề, DVD có giá cao hơn sẽ được hiển thị trước.
- **Sắp xếp theo giá**: hệ thống hiển thị tất cả các DVD theo thứ tự giảm dần của giá. Nếu có cùng giá, DVD sẽ được sắp xếp theo thứ tự tăng dần của tiêu đề.

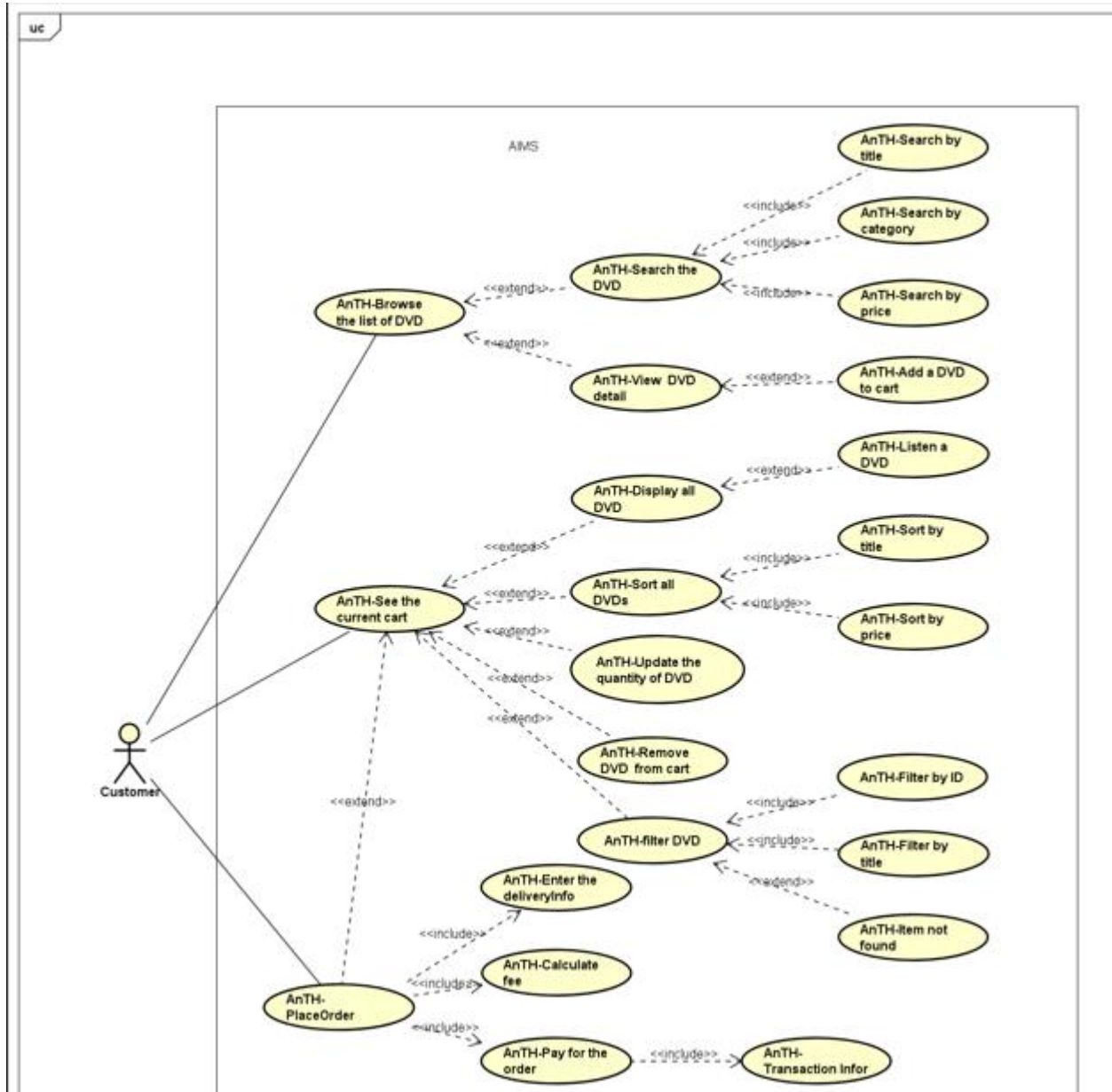
Khách hàng có thể cập nhật số lượng DVD trong giỏ hàng hoặc xóa DVD khỏi giỏ. Để tăng nhu cầu tiêu dùng và thúc đẩy doanh số, khách hàng được phép nhận một mặt hàng miễn phí, được chọn ngẫu nhiên từ giỏ hàng bởi hệ thống. Khách hàng có thể lọc DVD trong giỏ hàng bằng cách cung cấp ID hoặc tiêu đề của nó. Nếu tìm thấy mặt hàng, hệ thống sẽ hiển thị thông tin của mặt hàng trong giỏ hàng. Nếu không, thông báo rằng mặt hàng không được tìm thấy trong giỏ hàng hiện tại.

Khách hàng có thể yêu cầu đặt hàng khi đang xem giỏ hàng hiện tại. Để đơn giản hóa, khách hàng không cần đăng nhập để đặt hàng. Ứng dụng sẽ yêu cầu khách hàng nhập thông tin giao hàng và chỉ dẫn giao hàng. Phần mềm sẽ tính phí giao hàng dựa trên tổng khối lượng của đơn hàng và vị trí giao hàng. Sau đó, hệ thống sẽ hiển thị hóa đơn bao gồm danh sách DVD, tổng chi phí trước VAT, tổng chi phí sau VAT, và phí giao hàng. Khách hàng sau đó có thể tiến hành thanh toán cho đơn hàng. Hiện tại, chỉ có một phương thức thanh toán – thẻ tín dụng – được hỗ trợ bằng cách kết nối với hệ thống thẻ để kiểm tra tính hợp lệ của thẻ hoặc thực hiện giao dịch thanh toán. Sau giao dịch, phần mềm AIMS sẽ hiển thị tất cả các thông tin chi tiết như ID giao dịch, chủ thẻ, số tiền giao dịch, thông điệp giao dịch, số dư, ngày giao dịch cho khách hàng. Đơn hàng sẽ ở trạng thái chờ xử lý và thông tin của đơn hàng và giao dịch sẽ được gửi đến email của khách hàng.

2.2. Đối với Store Manager:

Quản lý cửa hàng cần đăng nhập vào hệ thống để điều hướng đến chế độ quản lý. Họ có thể xem danh sách các đơn hàng đang chờ xử lý, sau đó chọn bất kỳ đơn hàng nào để xem chi tiết và phê duyệt hoặc từ chối đơn hàng đó. Quản lý cửa hàng có thể thêm DVD mới vào cửa hàng. Họ phải cung cấp đầy đủ thông tin của DVD mới, bao gồm ID, tiêu đề, thể loại, đạo diễn, độ dài, và giá. Ngoài ra, quản lý cũng có thể xóa DVD khỏi cửa hàng.

3. Use Case Diagram



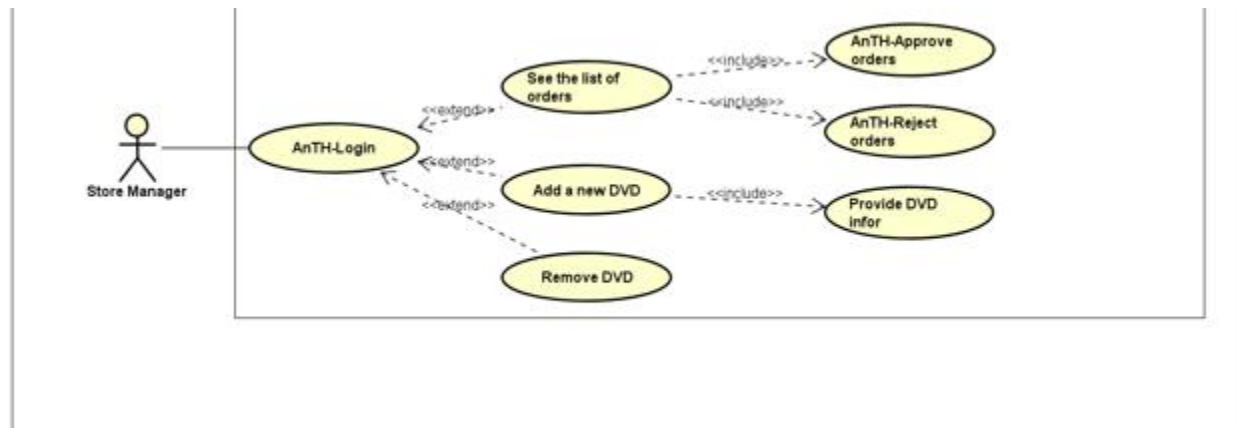


Figure 1: Use Case Diagram

4. Class Diagram

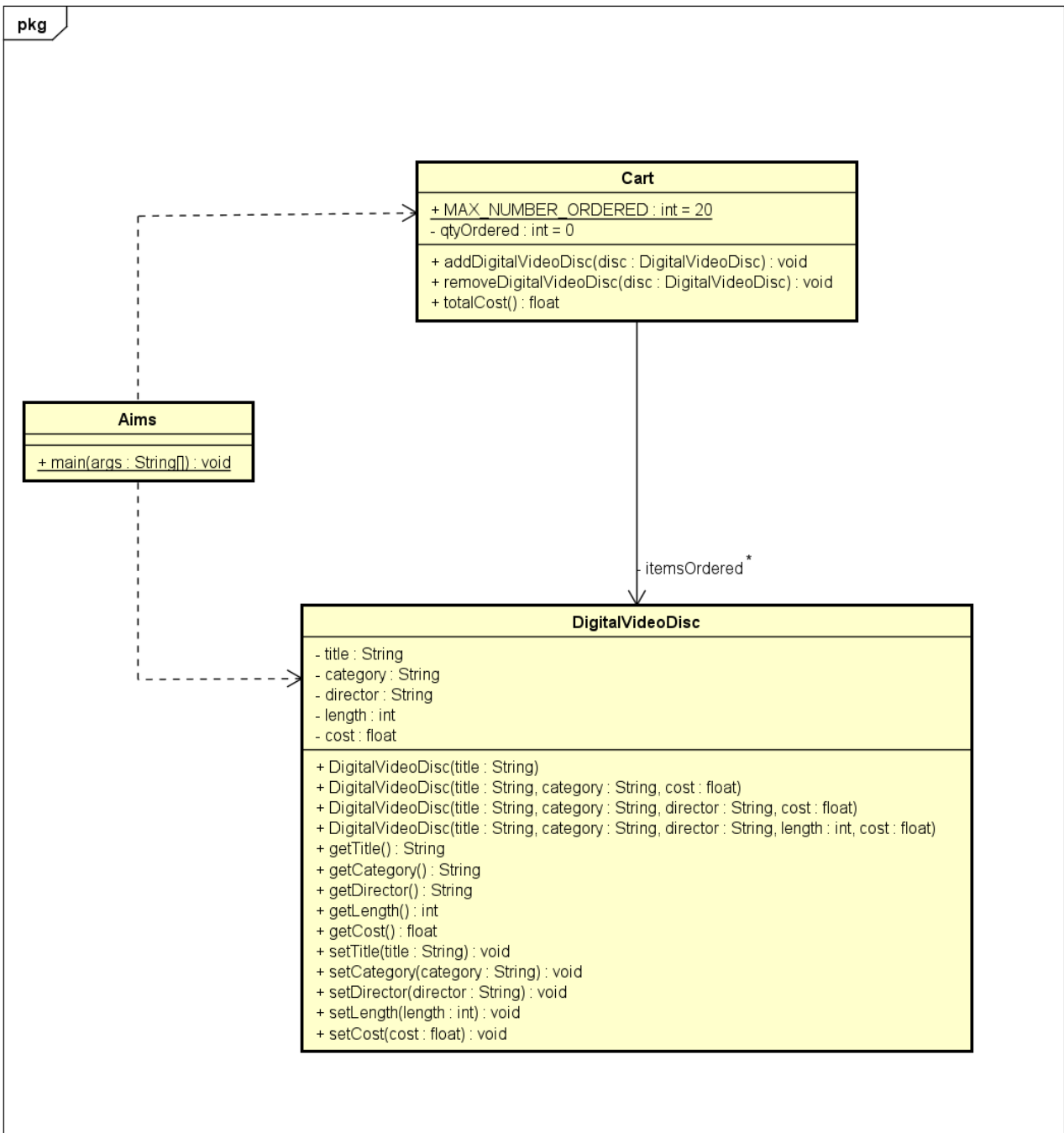


Figure 2: Class Diagram

5. Source Code

5.1. Aims Class

```
1 public class Aims {  
2     public static void main(String[] args) {  
3         Cart anOrder = new Cart();  
4         // Trinh Huu An 20225593  
5         DigitalVideoDisc dvd1 = new DigitalVideoDisc( title: "The Lion King",  
6             category: "Animation", director: "Roger Allers", length: 87, cost: 19.95f);  
7         anOrder.addDigitalVideoDisc(dvd1);  
8  
9         DigitalVideoDisc dvd2 = new DigitalVideoDisc( title: "Star Wars",  
10            category: "Science Fiction", director: "George Lucas", length: 87, cost: 24.95f);  
11        anOrder.addDigitalVideoDisc(dvd2);  
12  
13        DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladdin",  
14            category: "Animation", cost: 18.99f);  
15        anOrder.addDigitalVideoDisc(dvd3);  
16  
17        anOrder.removeDigitalVideoDisc(dvd2);  
18  
19        System.out.println("Total cost is: ");  
20        System.out.print(anOrder.totalCost());  
21    }  
22 }
```

Figure 3: Aims Class

5.2. DigitalVideoDisc Class

```
1 public class DigitalVideoDisc { 10 usages new *
2     private String title; 5 usages
3     private String category; 4 usages
4     private String director; 3 usages
5     private int length; 2 usages
6     private float cost; 4 usages
7     // Trinh Huu An 20225593
8     public DigitalVideoDisc(String title) { no usages new *
9         this.title = title;
10    }
11
12    public DigitalVideoDisc(String title, String category, float cost) { 1 usage new *
13        this.title = title;
14        this.category = category;
15        this.cost = cost;
16    }
17
18    public DigitalVideoDisc(String title, String category, String director, float cost) { no usages new *
19        this.title = title;
20        this.category = category;
21        this.director = director;
22        this.cost = cost;
23    }
24
25    public DigitalVideoDisc(String title, String category, String director, int length, float cost) {
26        this.title = title;
27        this.category = category;
28        this.director = director;
29        this.length = length;
30        this.cost = cost;
31    }
32
33    public String getTitle() { 3 usages new *
34        return title;
35    }
36
37    public String getCategory() { no usages new *
38        return category;
39    }
40
41    public String getDirector() { no usages new *
42        return director;
43    }
44 }
```



```
44
45     public int getLength() { no usages new *
46         return length;
47     }
48
49     public float getCost() { 1 usage new *
50         return cost;
51     }
52
53     public void setTitle(String title) { no usages new *
54         this.title = title;
55     }
56
57     public void setCategory(String category) { no usages new *
58         this.category = category;
59     }
60
61     public void setDirector(String director) { no usages new *
62         this.director = director;
63     }
64
65     public void setLength(int length) { no usages new *
66         this.length = length;
67     }
68
69     public void setCost(float cost) { no usages new *
70         this.cost = cost;
71     }
72 }
```

Figure 4: DigitalVideoDisc Class

5.3. Cart Class

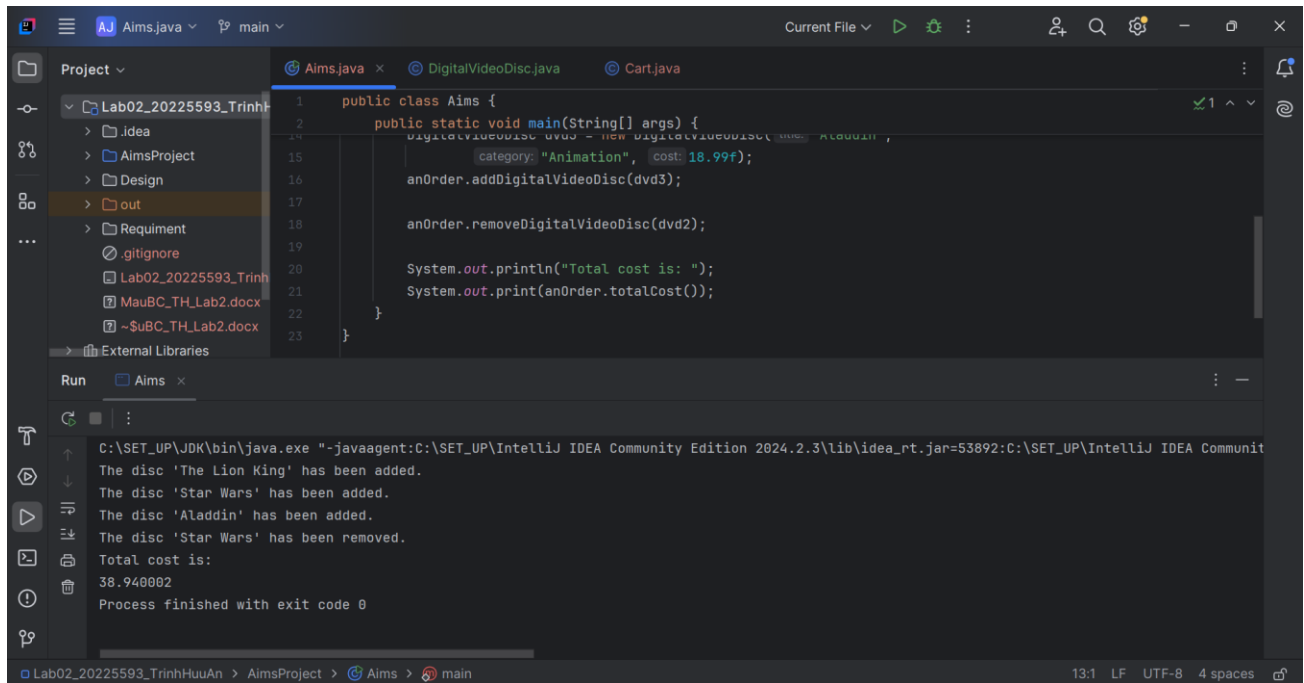
```

1 public class Cart { 2 usages
2     public static final int MAX_NUMBERS_ORDERED = 20; 4 usages
3     private DigitalVideoDisc itemsOrdered[] = new DigitalVideoDisc[MAX_NUMBERS_ORDERED]; 5 usages
4     private int qtyOrdered = 0; 7 usages
5     // Trinh Huu An 20225593
6     public void addDigitalVideoDisc(DigitalVideoDisc disc) { 3 usages
7         if (qtyOrdered < MAX_NUMBERS_ORDERED) {
8             itemsOrdered[qtyOrdered] = disc;
9             qtyOrdered++;
10            System.out.println("The disc '" + disc.getTitle() + "' has been added.");
11        } else {
12            System.out.println("The cart is almost full.");
13        }
14    }
15
16    public void removeDigitalVideoDisc(DigitalVideoDisc disc) { 1 usage
17        int check = MAX_NUMBERS_ORDERED + 1;
18        for (int i = 0; i < qtyOrdered; i++) {
19            if (itemsOrdered[i].getTitle().equals(disc.getTitle()))
20                check = i;
21        }
22
23        if (check != MAX_NUMBERS_ORDERED + 1) {
24            for (int i = check; i < qtyOrdered; i++)
25                itemsOrdered[i] = itemsOrdered[i + 1];
26
27            System.out.println("The disc '" + disc.getTitle() + "' has been removed.");
28            qtyOrdered--;
29        }
30    }
31
32    public float totalCost() { 1 usage
33        float totalCost = 0;
34        for(int i = 0; i < qtyOrdered; i++) {
35            totalCost += itemsOrdered[i].getCost();
36        }
37
38        return totalCost;
39    }
40 }

```

Figure 5: Cart Class

6. Kết quả demo



The screenshot displays the IntelliJ IDEA IDE interface. The top toolbar shows the 'Run' button (a green play icon). The main editor window shows the `Aims.java` file with the following code:

```
1 public class Aims {
2     public static void main(String[] args) {
3         DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King",
4             "Animation", 18.99f);
5         DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars",
6             "Animation", 14.99f);
7         DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladdin",
8             "Animation", 13.99f);
9         anOrder.addDigitalVideoDisc(dvd1);
10        anOrder.addDigitalVideoDisc(dvd2);
11        anOrder.addDigitalVideoDisc(dvd3);
12        anOrder.removeDigitalVideoDisc(dvd2);
13        System.out.println("Total cost is: ");
14        System.out.print(anOrder.totalCost());
15    }
16 }
```

The bottom panel shows the 'Run' output, which contains the following text:

```
C:\SET_UP\JDK\bin\java.exe "-javaagent:C:\SET_UP\IntelliJ IDEA Community Edition 2024.2.3\lib\idea_rt.jar=53892:C:\SET_UP\IntelliJ IDEA Communit
The disc 'The Lion King' has been added.
The disc 'Star Wars' has been added.
The disc 'Aladdin' has been added.
The disc 'Star Wars' has been removed.
Total cost is:
38.940002
Process finished with exit code 0
```

Figure 6: Result

7. Reading Assignment

- When should accessor methods be used?

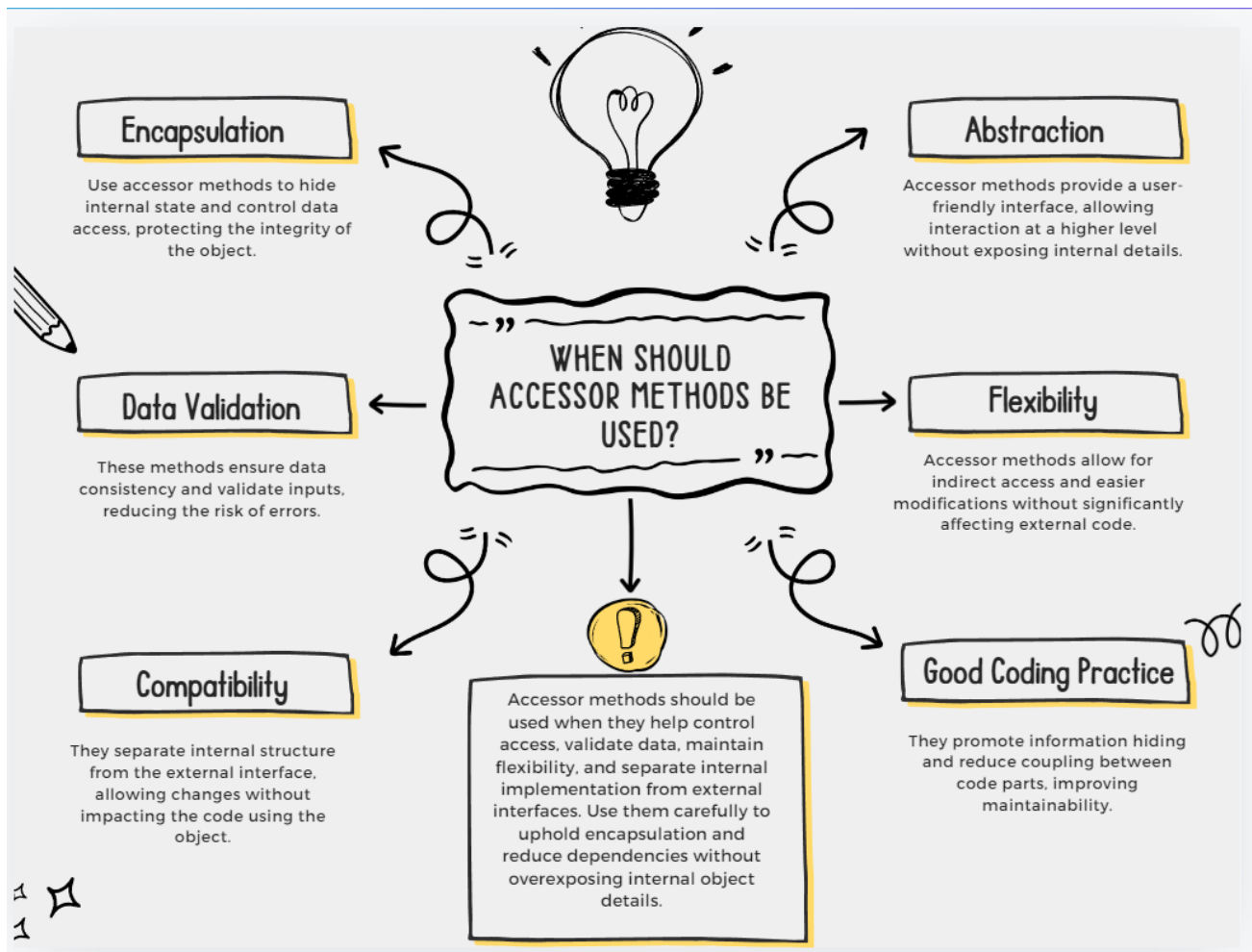


Figure 7: Getter and Setter Methods

8. Answer the question

Question: If you create a constructor method to build a DVD by title then create a constructor method to build a DVD by category. Does JAVA allow you to do this?

Answer: Java **không cho phép** thực hiện điều này bởi vì phương thức khởi tạo với tham số title có kiểu String, do đó nếu ta tạo phương thức khởi tạo với tham số category cũng kiểu String thì khi gọi phương thức, Java sẽ không thể biết được mình đang muốn gọi đến phương thức khởi tạo nào. Nếu muốn sử dụng phương thức khởi tạo có tham số là category, ta phải sử dụng kiểu dữ liệu **khác String**.