ĐẠI HỌC BÁCH KHOA HÀ NỘI  
 TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

BÁO CÁO THỰC HÀNH  
 **IT3103-744527-2024.1**

BÀI THỰC HÀNH 04

Họ và tên sv: **Trịnh Hữu An**

MSSV: **20225593**  
 Lớp: **Việt Nhật 03 – K67**

GVHD: **Lê Thị Hoa**

TA: **Đặng Mạnh Cường**

Hà Nội 12/2024

Table of Contents

[1. Create the Book class 5](#_Toc184081167)

[2. Creating the abstract Media class 6](#_Toc184081168)

[3. Creating the CompactDisc class 8](#_Toc184081169)

[3.1. Create the Disc class extending the Media clas 8](#_Toc184081170)

[3.2. Create the Track class which models a track on a compact disc and will store information incuding the title and length of the track. 11](#_Toc184081171)

[3.3. Open the CompactDisc class 12](#_Toc184081172)

[4. Create the Playable interface 13](#_Toc184081173)

[5. Update the Cart class to work with Media 14](#_Toc184081174)

[6. Update the Store class to work with Media 16](#_Toc184081175)

[7. Constructors of whole classes and parent classes 17](#_Toc184081176)

[8. Unique item in a list 18](#_Toc184081177)

[9. Polymorphism with toString() method 19](#_Toc184081178)

[10. Sort media in the car 20](#_Toc184081179)

[11. Create a complete console application in the Aims class 22](#_Toc184081180)

[11.1. Người dùng chọn 1: View store 22](#_Toc184081181)

[11.1.1. Người dùng tiếp tục chọn 1. See a media’s details 23](#_Toc184081182)

[11.1.2. Người dùng chọn 2: Add a media to the cart 23](#_Toc184081183)

[11.1.3. Người dùng chọn 3: Play a media 24](#_Toc184081184)

[11.1.4. Người dùng chọn 4: See current cart 24](#_Toc184081185)

[11.2. Người dùng chọn 2: Update store 24](#_Toc184081186)

[11.2.1. Người dùng chọn 1: Add a media to the store 25](#_Toc184081187)

[11.2.2. Người dùng chọn 2: Remove a media from the store 26](#_Toc184081188)

[11.3. Người dùng chọn 3: See current cart 27](#_Toc184081189)

[11.3.1. Người dùng chọn 1: Filter medias in cart 28](#_Toc184081190)

[11.3.2. Người dùng chọn 2: Sort medias in cart 29](#_Toc184081191)

[11.3.3. Người dùng chọn 3: Remove media from cart 30](#_Toc184081192)

[11.3.4. Người dùng chọn 4: Play a media 30](#_Toc184081193)

[11.3.5. Người dùng chọn 5: Place order 31](#_Toc184081194)

[12. Class Diagram 31](#_Toc184081195)

[13. Class Diagram 32](#_Toc184081196)

[14. Answer Questions 34](#_Toc184081197)

Table of Figures

[Figure 1: Book Class 6](#_Toc184082437)

[Figure 2: Media Class 8](#_Toc184082438)

[Figure 3: Disc Class 9](#_Toc184082439)

[Figure 4: DigitalVideoDisc Class 11](#_Toc184082440)

[Figure 5: CompactDisc Class 11](#_Toc184082441)

[Figure 6: Track Class 13](#_Toc184082442)

[Figure 7: CompactDisc Class 14](#_Toc184082443)

[Figure 8: Playable interface 14](#_Toc184082444)

[Figure 9: Method play() của DigitalVideoDisc 14](#_Toc184082445)

[Figure 10: Method play() của Track 15](#_Toc184082446)

[Figure 11: Method play() của CompactDisc 15](#_Toc184082447)

[Figure 12: Cart Class 17](#_Toc184082448)

[Figure 13: Store Class 18](#_Toc184082449)

[Figure 14: Constructor Track Class 18](#_Toc184082450)

[Figure 15: Constructor CompactDisc Class 19](#_Toc184082451)

[Figure 16: Constructor Media Class 19](#_Toc184082452)

[Figure 17: Constructor Disc Class 19](#_Toc184082453)

[Figure 18: Override equals() in Media Class 19](#_Toc184082454)

[Figure 19: Override Equals in Track Class 20](#_Toc184082455)

[Figure 20: Code mô phỏng Polymorphism 20](#_Toc184082456)

[Figure 21: Override toString() in Media Class 20](#_Toc184082457)

[Figure 22: Result Demo 21](#_Toc184082458)

[Figure 23: Add the comparators as attributes of the Media class 21](#_Toc184082459)

[Figure 24: MediaComparatorByCostTitle Class 22](#_Toc184082460)

[Figure 25: MediaComparatorByTitleCost Class 22](#_Toc184082461)

[Figure 26: Màn hình chính 23](#_Toc184082462)

[Figure 27: vào trang View Store 23](#_Toc184082463)

[Figure 28: See a media's details 24](#_Toc184082464)

[Figure 29: Add to cart 24](#_Toc184082465)

[Figure 30: Thêm media vào cart 24](#_Toc184082466)

[Figure 31: Play a media 25](#_Toc184082467)

[Figure 32: See current cart after sort 25](#_Toc184082468)

[Figure 33: Update Store 25](#_Toc184082469)

[Figure 34: Add a media to store 26](#_Toc184082470)

[Figure 35: Resul 27](#_Toc184082471)

[Figure 36: Remove a media 27](#_Toc184082472)

[Figure 37: Result 28](#_Toc184082473)

[Figure 38: See current cart 28](#_Toc184082474)

[Figure 39: Media in cart 28](#_Toc184082475)

[Figure 40: Filter Cart by Id 29](#_Toc184082476)

[Figure 41: Filter Cart by Title 30](#_Toc184082477)

[Figure 42: Sort Cart by Title 30](#_Toc184082478)

[Figure 43: Sort Cart by Cost 30](#_Toc184082479)

[Figure 44: Remove media from Cart 31](#_Toc184082480)

[Figure 45: Play a media 31](#_Toc184082481)

[Figure 46: Result 31](#_Toc184082482)

[Figure 47: Place Order 32](#_Toc184082483)

[Figure 48: Result 32](#_Toc184082484)

[Figure 49: Class Diagram 33](#_Toc184082485)

[Figure 50: Use Case Diagram 34](#_Toc184082486)

# Create the Book class

A screen shot of a computer

Description automatically generated

A screen shot of a computer code

Description automatically generated

A computer screen with colorful text

Description automatically generated with medium confidence

Figure 1: Book Class

# Creating the abstract Media class

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A screen shot of a computer code

Description automatically generated

Figure 2: Media Class

# Creating the CompactDisc class

## Create the Disc class extending the Media clas

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Figure 3: Disc Class

A screenshot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

A computer code on a black background

Description automatically generated

Figure 4: DigitalVideoDisc Class

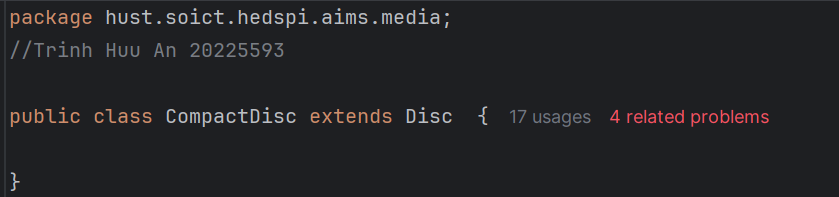


Figure 5: CompactDisc Class

## Create the Track class which models a track on a compact disc and will store information incuding the title and length of the track.

A screen shot of a computer

Description automatically generated

A computer screen shot of a program code

Description automatically generated

A black rectangle with white text

Description automatically generated

Figure 6: Track Class

## Open the CompactDisc class

A screenshot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A computer screen shot of a black screen

Description automatically generated

A computer screen with text on it

Description automatically generated

Figure 7: CompactDisc Class

# Create the Playable interface

A black screen with white text

Description automatically generatedFigure 8: Playable interface

Implement play() cho các class DigitalVideoDisc, Track, CompactDisc

A computer code on a black background

Description automatically generated

Figure 9: Method play() của DigitalVideoDisc

A screen shot of a computer code

Description automatically generated

Figure 10: Method play() của Track

A computer code on a black background

Description automatically generatedFigure 11: Method play() của CompactDisc

# Update the Cart class to work with Media

Lớp Cart bây giờ cần có khả năng tương tác với các đối tượng DVD, CD và Book. Vì các lớp DVD, CD và Book đều kế thừa từ lớp Media, nên thay vì làm việc trực tiếp với từng lớp con, lớp cart chỉ cần giao tiếp với lớp Media là có thể hoạt động được với tất cả.



A computer screen shot of a program code

Description automatically generated

A computer screen shot of a black screen

Description automatically generated

A screen shot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

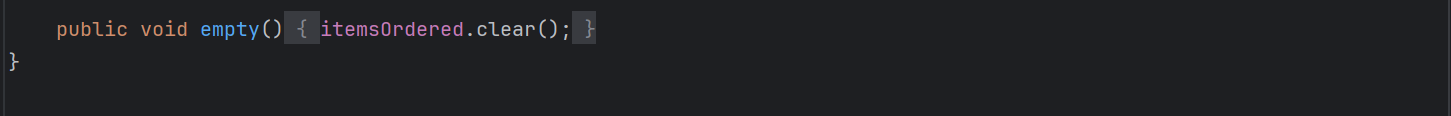


Figure 12: Cart Class

# Update the Store class to work with Media



A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

Figure 13: Store Class

# Constructors of whole classes and parent classes

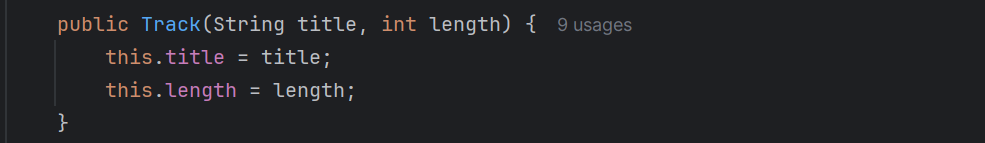


Figure 14: Constructor Track Class



Figure 15: Constructor CompactDisc Class

Lớp Disc kế thừa lớp Media, khi đó lớp Media là lớp cha, lớp Disc là lớp con.



Figure 16: Constructor Media Class

A black background with white text

Description automatically generatedFigure 17: Constructor Disc Class

# Unique item in a list

A black screen with text

Description automatically generated

Figure 18: Override equals() in Media Class

A screen shot of a computer code

Description automatically generatedFigure 19: Override Equals in Track Class

# Polymorphism with toString() method

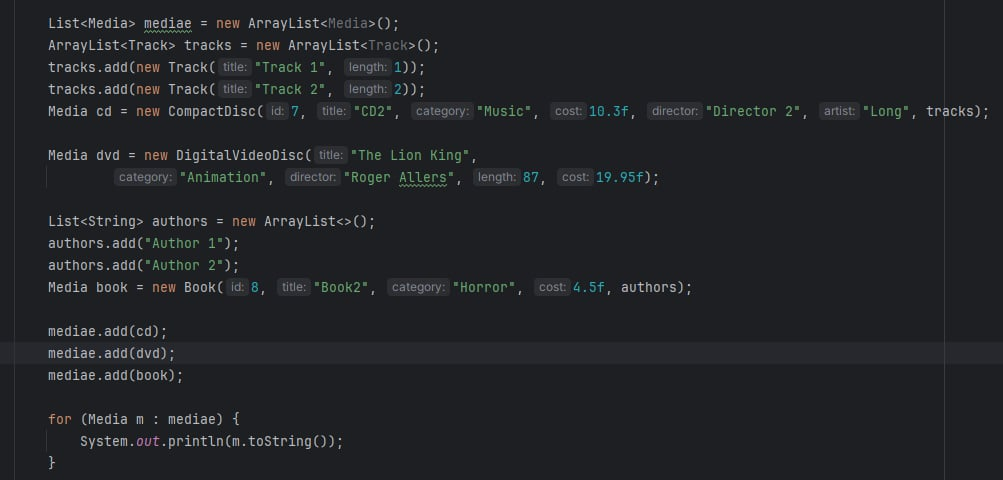


Figure 20: Code mô phỏng Polymorphism

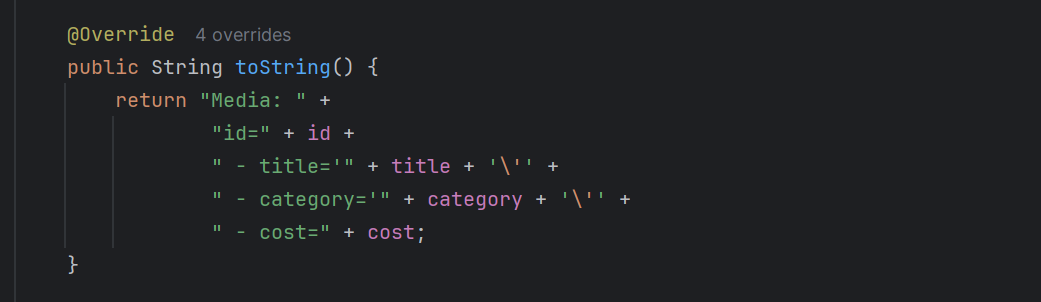


Figure 21: Override toString() in Media Class

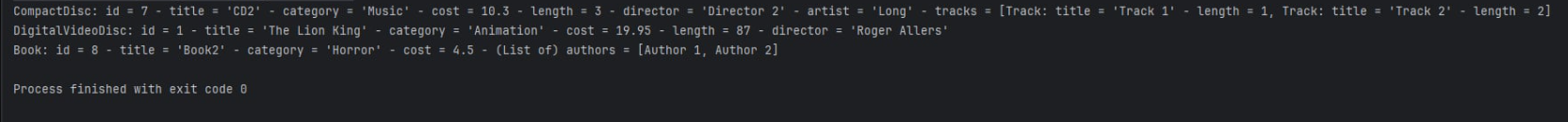


Figure 22: Result Demo

* Lớp Media là lớp cha cơ bản, được các lớp con như CompactDisc, DigitalVideoDisc, và Book kế thừa để đại diện cho các loại phương tiện cụ thể hơn. Khi chúng ta tạo các đối tượng từ các lớp con như cd, dvd, hoặc book và gán chúng vào biến kiểu Media, điều này được gọi là **upcasting**. Upcasting cho phép một đối tượng lớp con được xử lý như thể nó thuộc về lớp cha.
* Một ví dụ tiêu biểu về tính **đa hình động** là khi chúng ta thêm các đối tượng này vào một danh sách List<Media> và duyệt qua danh sách để in thông tin của từng phần tử. Phương thức toString() được gọi trên mỗi đối tượng trong danh sách, nhưng thay vì luôn sử dụng phiên bản của lớp cha, Java tự động sử dụng phiên bản được ghi đè trong lớp con tương ứng.
* Điều này nghĩa là, dù các phần tử đều được xử lý như kiểu Media, kết quả thực tế của phương thức toString() sẽ khác nhau tùy thuộc vào đối tượng thuộc lớp con nào. Do đó, ta không cần kiểm tra kiểu cụ thể của từng đối tượng trong danh sách nhưng vẫn nhận được thông tin phù hợp với loại phương tiện thực tế.
* Dù cả ba phần tử trong danh sách đều thuộc kiểu Media, phương thức toString() được gọi sẽ là phiên bản tương ứng với lớp con thực sự của từng đối tượng (CompactDisc, DigitalVideoDisc, hoặc Book). Điều này minh họa tính đa hình, nơi hành vi của một phương thức thay đổi dựa trên đối tượng thực sự đang được sử dụng.

# Sort media in the car

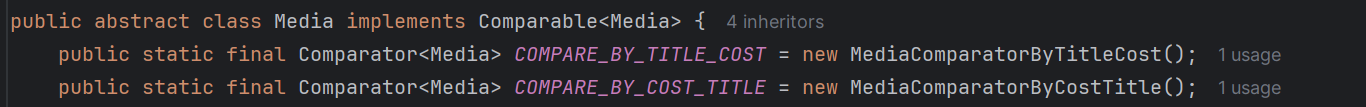


Figure 23: Add the comparators as attributes of the Media class

A computer screen shot of a program code

Description automatically generated

Figure 24: MediaComparatorByCostTitle Class

A computer screen shot of a black screen

Description automatically generated

Figure 25: MediaComparatorByTitleCost Class

# Create a complete console application in the Aims class

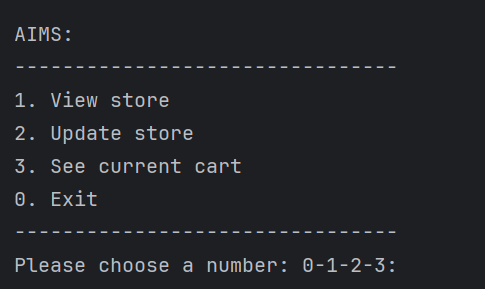


Figure 26: Màn hình chính

## Người dùng chọn 1: View store

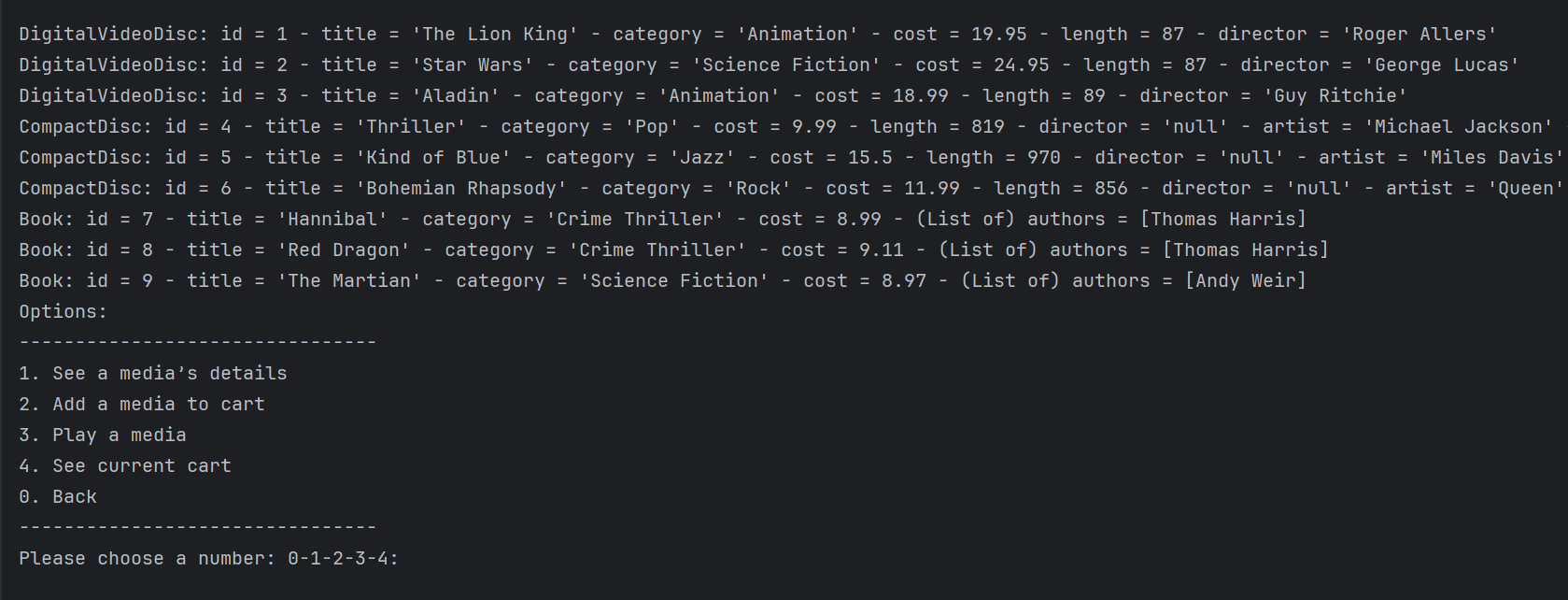


Figure 27: vào trang View Store

### Người dùng tiếp tục chọn 1. See a media’s details

A computer screen shot of text

Description automatically generated

Figure 28: See a media's details

A screen shot of a computer

Description automatically generatedFigure 29: Add to cart

### Người dùng chọn 2: Add a media to the cart

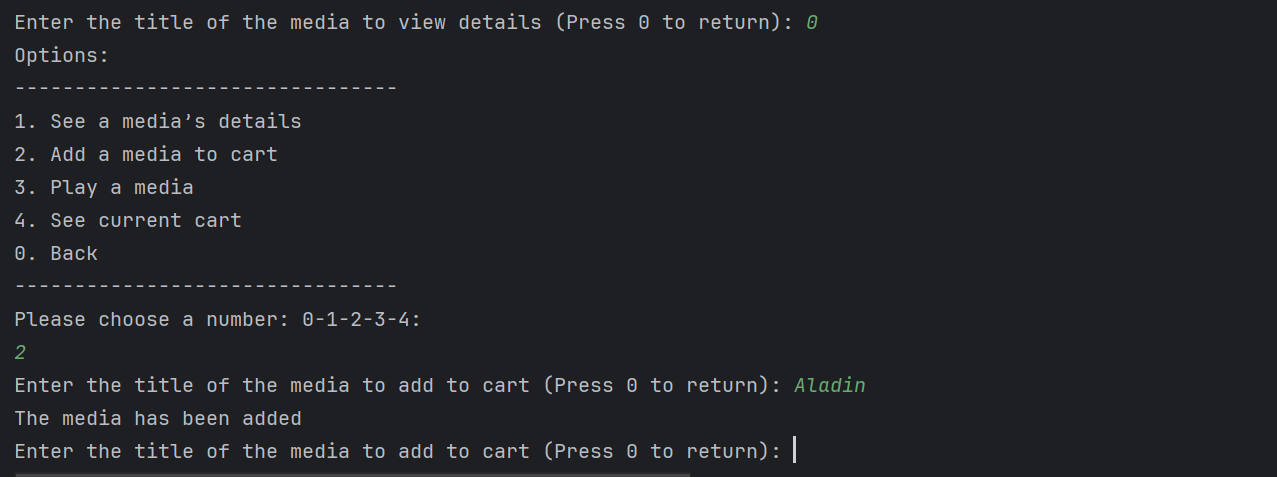


Figure 30: Thêm media vào cart

### Người dùng chọn 3: Play a media

A screen shot of a computer

Description automatically generated

A black background with white text

Description automatically generated

Figure 31: Play a media

### Người dùng chọn 4: See current cart

A computer screen with white text

Description automatically generated

Figure 32: See current cart after sort

## Người dùng chọn 2: Update store

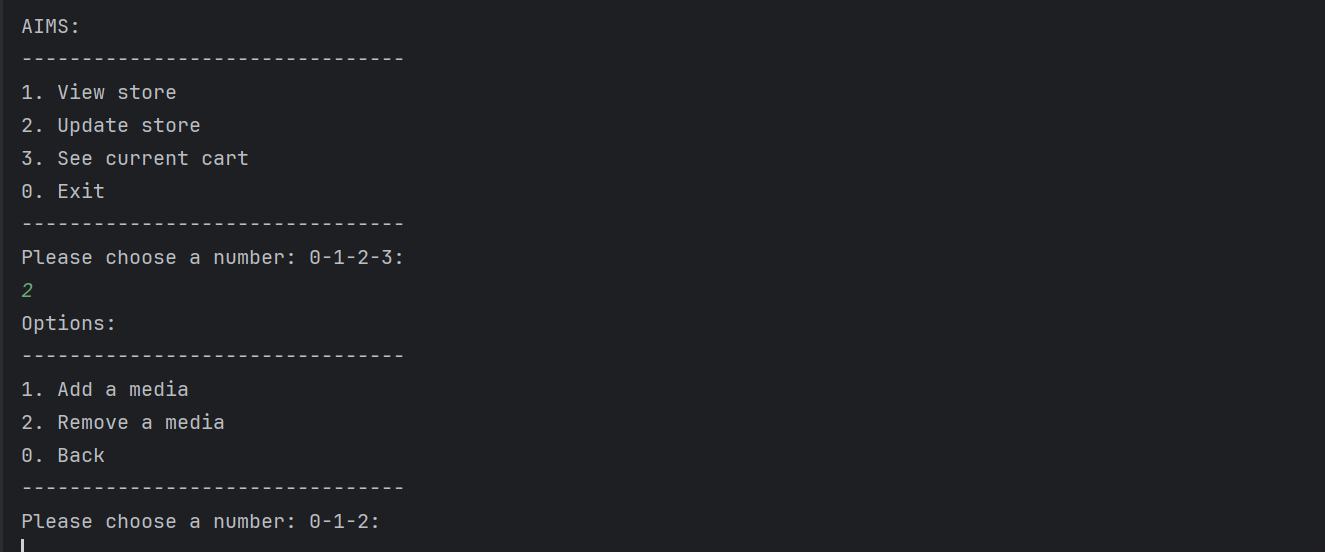


Figure 33: Update Store

### Người dùng chọn 1: Add a media to the store

A screenshot of a computer

Description automatically generated

A black screen with white text

Description automatically generated

Figure 34: Add a media to store

A black background with white text

Description automatically generated

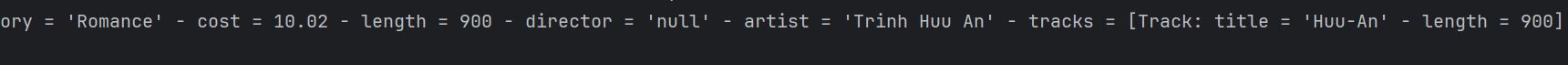


Figure 35: Resul

### Người dùng chọn 2: Remove a media from the store

A screenshot of a computer program

Description automatically generated

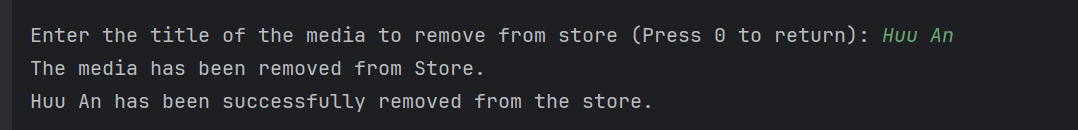


Figure 36: Remove a media

A screen shot of a computer code

Description automatically generated

Figure 37: Result

## Người dùng chọn 3: See current cart

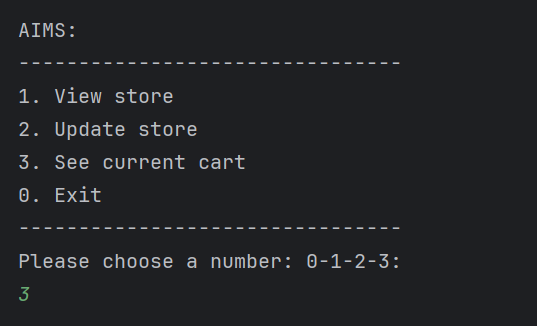


Figure 38: See current cart

Giả sử trong cart lúc này sẽ có các media sau:

A computer screen with text

Description automatically generated

Figure 39: Media in cart

### Người dùng chọn 1: Filter medias in cart

A screenshot of a computer

Description automatically generated

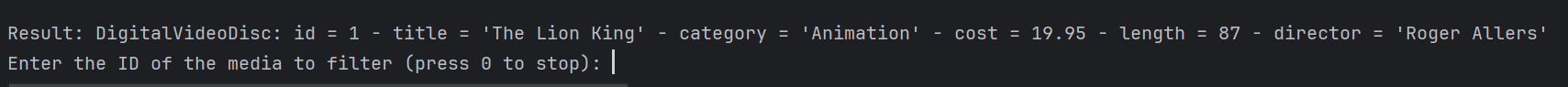


Figure 40: Filter Cart by Id



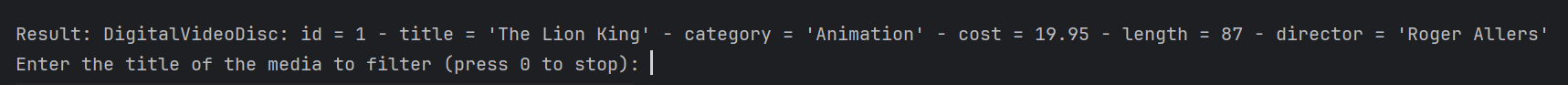


Figure 41: Filter Cart by Title

### Người dùng chọn 2: Sort medias in cart



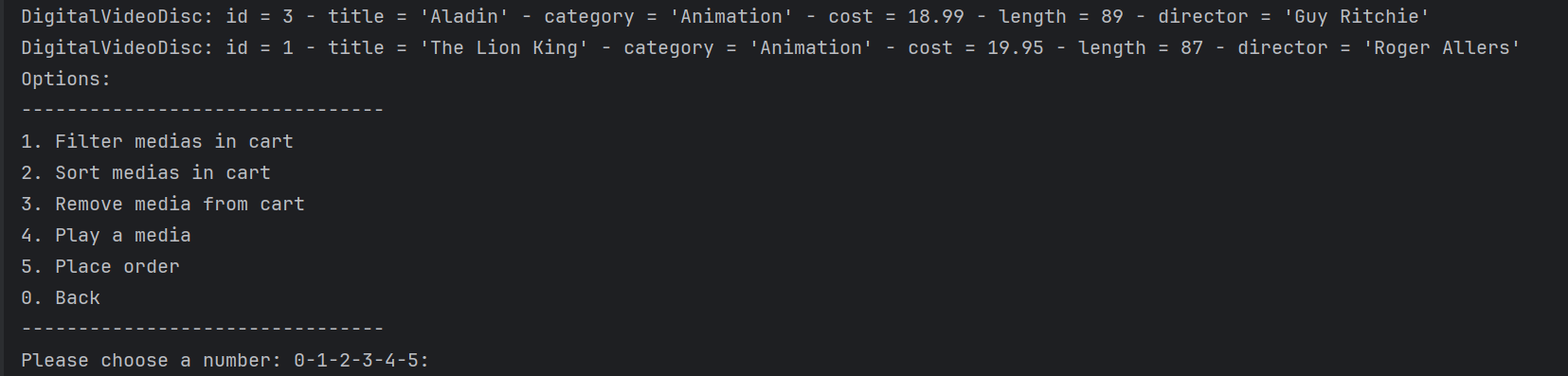


Figure 42: Sort Cart by Title



Figure 43: Sort Cart by Cost

### Người dùng chọn 3: Remove media from cart



Figure 44: Remove media from Cart

### Người dùng chọn 4: Play a media



Figure 45: Play a media

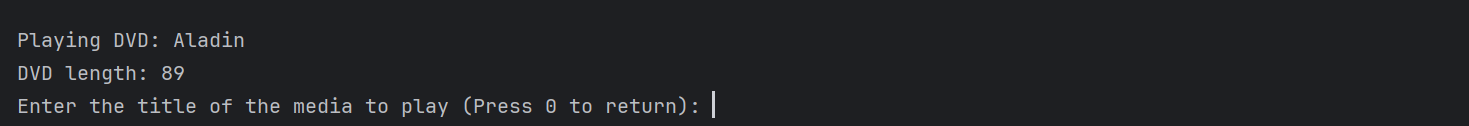


Figure 46: Result

### Người dùng chọn 5: Place order

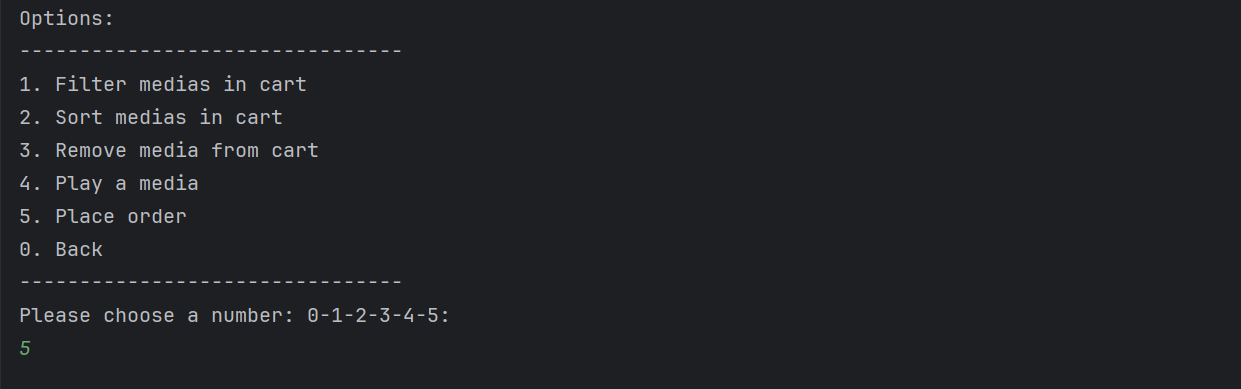


Figure 47: Place Order

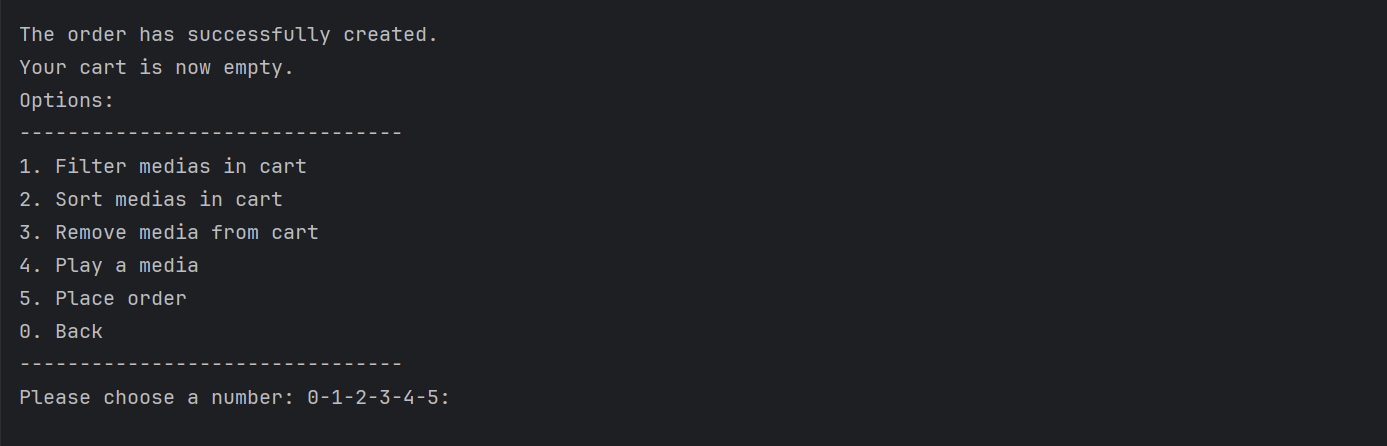


Figure 48: Result

# Class Diagram

# Class Diagram

A screenshot of a computer program

Description automatically generated

Figure 49: Class Diagram

A diagram of a person's work flow

Description automatically generated

Figure 50: Use Case Diagram

# Answer Questions

**10.Unique Item in a List**

Nếu đối tượng được truyền vào không phải là instance của lớp Media, việc so sánh sẽ không thể thực hiện được. Tuy nhiên, nếu đối tượng đó là instance của một lớp con kế thừa từ Media, phép so sánh vẫn hoạt động như bình thường.

Ví dụ minh họa trong lớp Aims:

Media book1 = new Book(5, "Book1", "Horror", 4.5f);

Media book2 = new Book(6, "Book1", "Scientific", 5.5f);

if (book1.equals(book2)) {

System.out.println("Two objects are equal");

} else {

System.out.println("Two objects are not equal");

}

**Kết quả:** Two objects are equal, vì cả hai đối tượng đều có tiêu đề (title) giống nhau, và phép so sánh chỉ dựa trên thuộc tính này.

**12.Sort Media in the Cart**

**a. What class should implement the Comparable interface?**

Lớp đại diện cho các đối tượng cần so sánh, chẳng hạn như Media hoặc các lớp con của nó (ví dụ DigitalVideoDisc, Book, CompactDisc), sẽ cần triển khai giao diện Comparable.

**b. In those classes, how should you implement the compareTo() method be to reflect the ordering that we want?**

Phương thức compareTo() cần được triển khai để phản ánh quy tắc sắp xếp mong muốn. Ví dụ, trong lớp Media, có thể sắp xếp theo tiêu đề trước, sau đó đến giá:

@Override

public int compareTo(Media other) {

int titleComparison = this.title.compareTo(other.title);

if (titleComparison != 0) {

return titleComparison;

}

return Float.compare(other.cost, this.cost);

}

**c. Can we have two ordering rules of the item (by title then cost and by cost then title) if we use this Comparable interface approach?**

Không, giao diện Comparable chỉ cho phép định nghĩa một quy tắc sắp xếp duy nhất qua phương thức compareTo(). Nếu cần nhiều quy tắc sắp xếp, cần sử dụng Comparator, chẳng hạn:

* MediaComparatorByTitleCost
* MediaComparatorByCostTitle.

**d. Suppose the DVDs has a different ordering rule from the other media types, that is by title, then decreasing length, then cost. How would you modify your code to allow this?**

Có thể ghi đè phương thức compareTo() trong lớp DigitalVideoDisc:

@Override

public int compareTo(Media other) {

if (!(other instanceof DigitalVideoDisc)) {

return super.compareTo(other);

}

DigitalVideoDisc otherDVD = (DigitalVideoDisc) other;

int titleComparison = this.getTitle().compareTo(other.getTitle());

if (titleComparison != 0) {

return titleComparison;

}

if (this.getLength() == otherDVD.getLength()) {

return Float.compare(otherDVD.getCost(), this.getCost());

}

return Integer.compare(otherDVD.getLength(), this.getLength());

}

Ví dụ kiểm tra:

ArrayList<DigitalVideoDisc> DVDList = new ArrayList<>();

DVDList.add(new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f));

DVDList.add(new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 25f));

DVDList.add(new DigitalVideoDisc("Aladdin", "Animation", "Roger Allers", 90, 12.0f));

System.out.println("Before sorting:");

for (DigitalVideoDisc dvd : DVDList) {

System.out.println(dvd.getTitle() + " " + dvd.getLength() + " " + dvd.getCost());

}

Collections.sort(DVDList);

System.out.println("\nAfter sorting:");

for (DigitalVideoDisc dvd : DVDList) {

System.out.println(dvd.getTitle() + " " + dvd.getLength() + " " + dvd.getCost());

}

Kết quả:

**Before sorting:**

**The Lion King 87 19.95**

**The Lion King 87 25.0**

**Aladdin 90 12.0**

**After sorting:**

**Aladdin 90 12.0**

**The Lion King 87 25.0**

**The Lion King 87 19.95**

Quy tắc sắp xếp được đảm bảo theo thứ tự: **Tiêu đề → Độ dài giảm dần → Giá**.