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

BÁO CÁO THỰC HÀNH
IT3103-744528-2024.1

BÀI THỰC HÀNH SỐ 4

Họ và tên sv: Đỗ Đình Trung

MSSV: 20226128
Lớp: **CNTT Việt Pháp 01-K67**

GVHD: Lê Thị Hoa

TA: **Đặng Mạnh Cường**

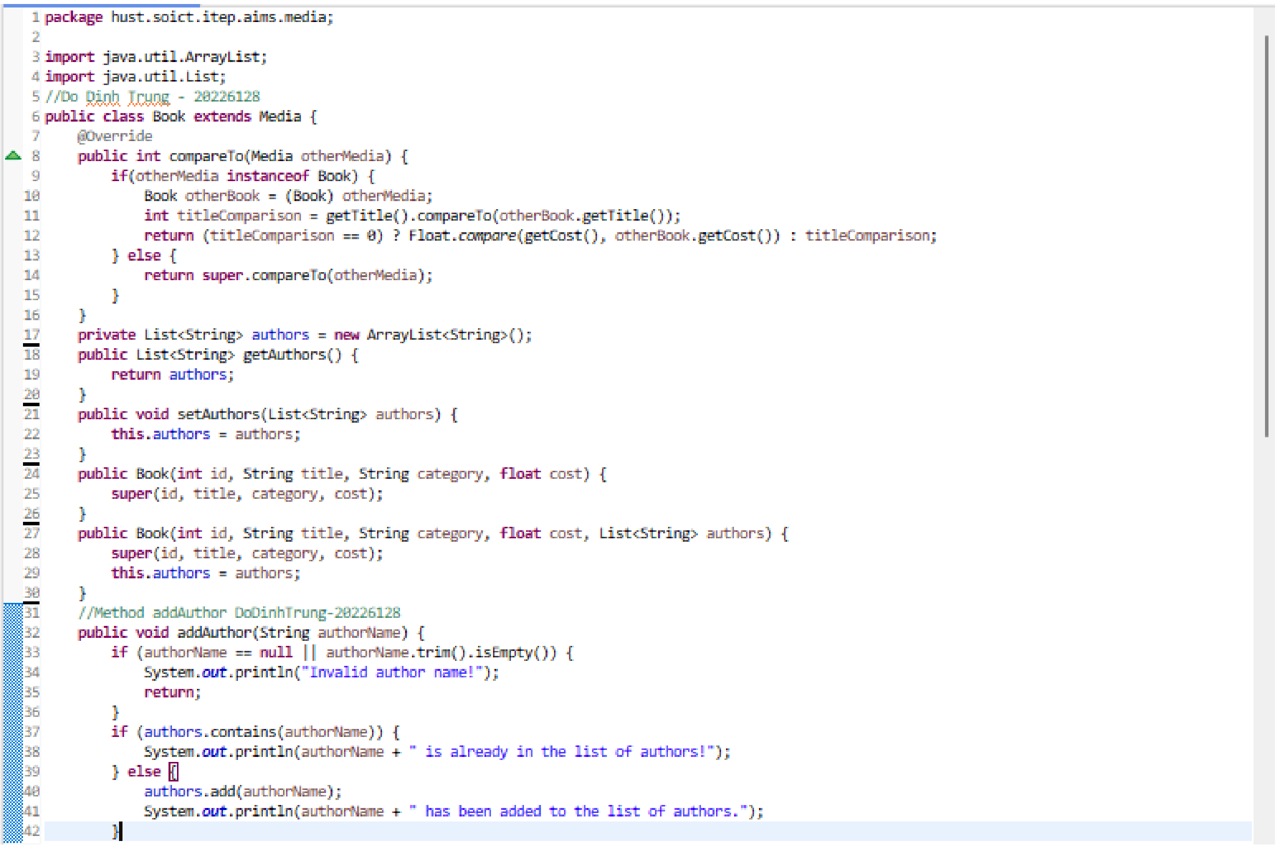
Hà Nội 12/2024

BÁO CÁO THỰC HÀNH LAB 4 LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Contents

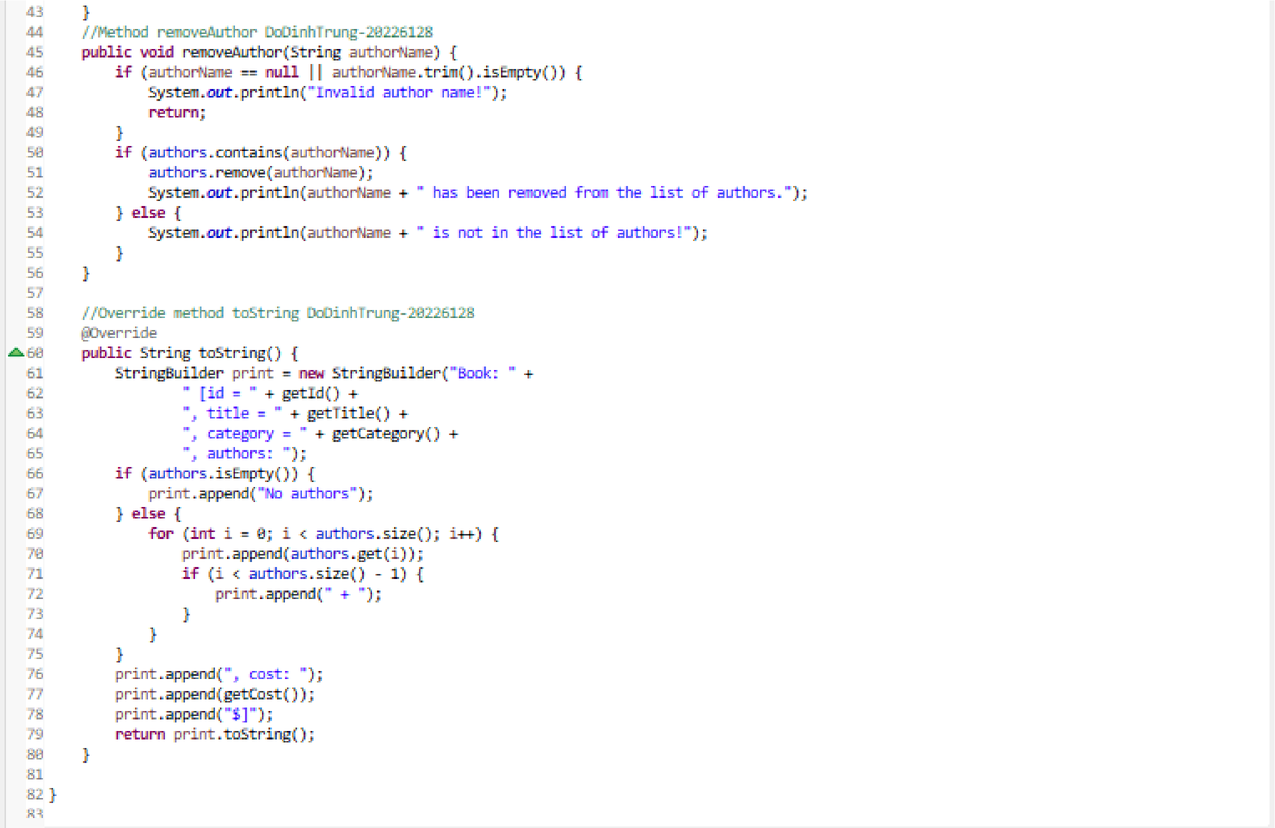
Table of Figures

1 Create the Book class



```java
package hust.soict.itep.aims.media;

import java.util.ArrayList;
import java.util.List;
//Do Dinh Trung - 20226128
public class Book extends Media {
    @Override
    public int compareTo(Media otherMedia) {
        if(otherMedia instanceof Book) {
            Book otherBook = (Book) otherMedia;
            int titleComparison = getTitle().compareTo(otherBook.getTitle());
            return (titleComparison == 0) ? Float.compare(getCost(), otherBook.getCost()) : titleComparison;
        } else {
            return super.compareTo(otherMedia);
        }
    }
    private List<String> authors = new ArrayList<String>();
    public List<String> getAuthors() {
        return authors;
    }
    public void setAuthors(List<String> authors) {
        this.authors = authors;
    }
    public Book(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }
    public Book(int id, String title, String category, float cost, List<String> authors) {
        super(id, title, category, cost);
        this.authors = authors;
    }
    //Method addAuthor DoDinhTrung-20226128
    public void addAuthor(String authorName) {
        if (authorName == null || authorName.trim().isEmpty()) {
            System.out.println("Invalid author name!");
            return;
        }
        if (authors.contains(authorName)) {
            System.out.println(authorName + " is already in the list of authors!");
        } else {
            authors.add(authorName);
            System.out.println(authorName + " has been added to the list of authors.");
        }
```

Figure 1.1: Book Class 1

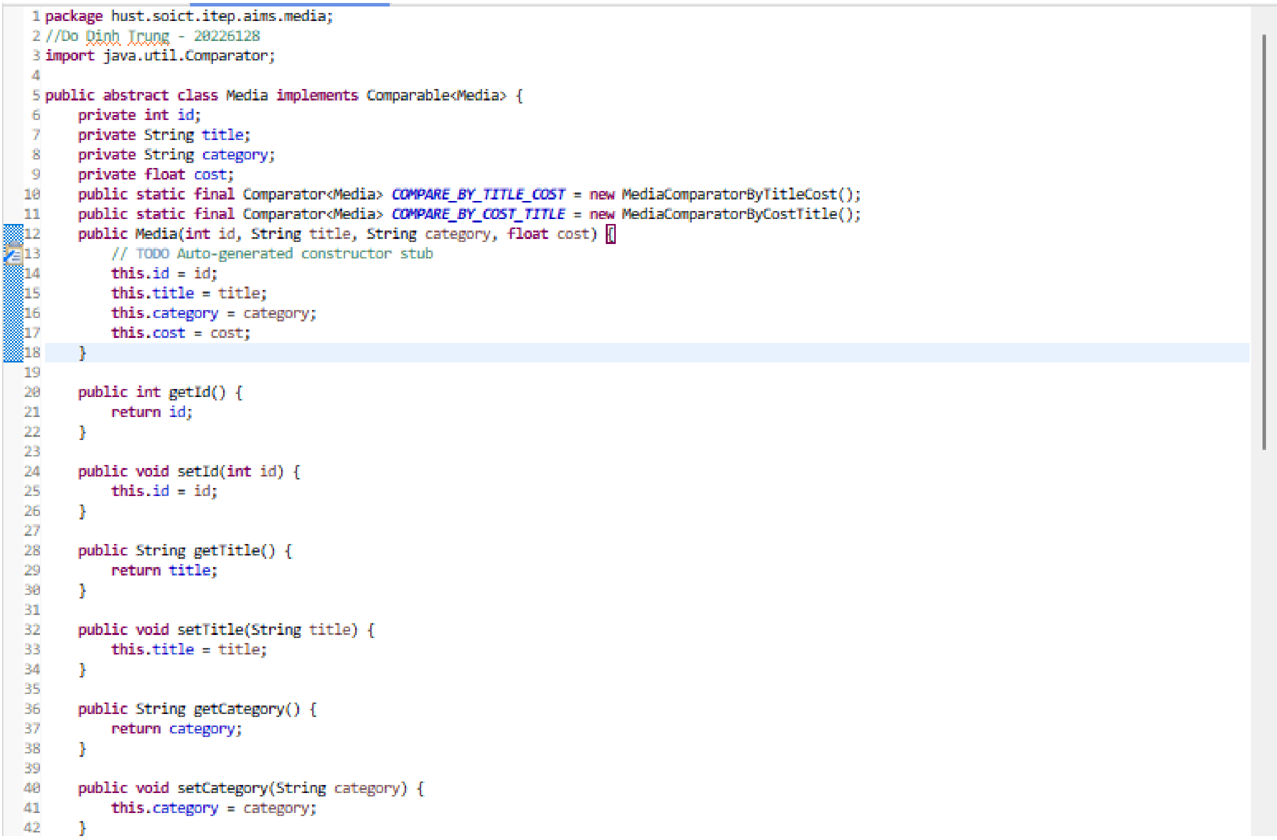


```
    }
    //Method removeAuthor DoDinhTrung-20226128
    public void removeAuthor(String authorName) {
        if (authorName == null || authorName.trim().isEmpty()) {
            System.out.println("Invalid author name!");
            return;
        }
        if (authors.contains(authorName)) {
            authors.remove(authorName);
            System.out.println(authorName + " has been removed from the list of authors.");
        } else {
            System.out.println(authorName + " is not in the list of authors!");
        }
    }

    //Override method toString DoDinhTrung-20226128
    @Override
    public String toString() {
        StringBuilder print = new StringBuilder("Book: " +
                " [id = " + getId() +
                ", title = " + getTitle() +
                ", category = " + getCategory() +
                ", authors: ");
        if (authors.isEmpty()) {
            print.append("No authors");
        } else {
            for (int i = 0; i < authors.size(); i++) {
                print.append(authors.get(i));
                if (i < authors.size() - 1) {
                    print.append(" + ");
                }
            }
        }
        print.append(", cost: ");
        print.append(getCost());
        print.append("$]");
        return print.toString();
    }

}
```

Figure 1.2: Book Class 2

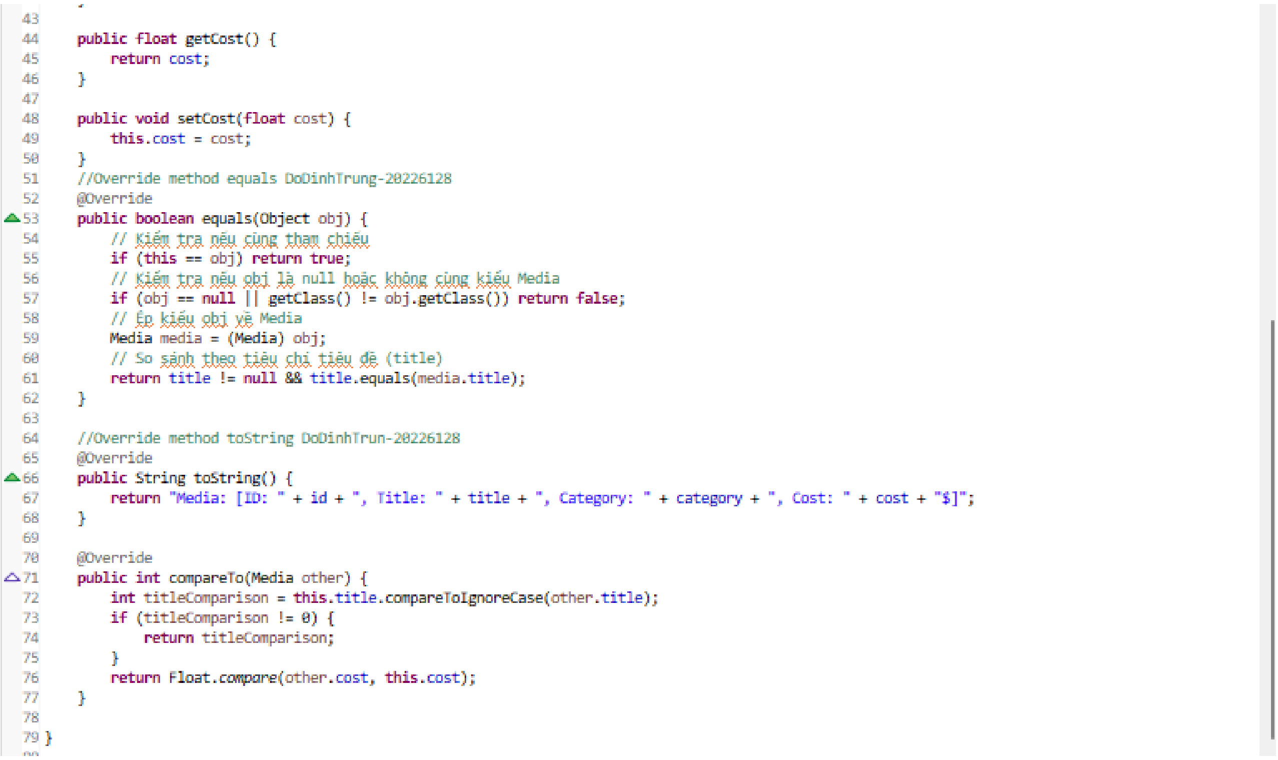
2 Creating the abstract Media class

Đây sẽ là lớp cha để các lớp DigitalVideoDisc, Book kế thừa.



```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
import java.util.Comparator;

public abstract class Media implements Comparable<Media> {
    private int id;
    private String title;
    private String category;
    private float cost;
    public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();
    public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();
    public Media(int id, String title, String category, float cost) {
        // TODO Auto-generated constructor stub
        this.id = id;
        this.title = title;
        this.category = category;
        this.cost = cost;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getCategory() {
        return category;
    }

    public void setCategory(String category) {
        this.category = category;
    }
```

Figure 2.1: Media Class 1

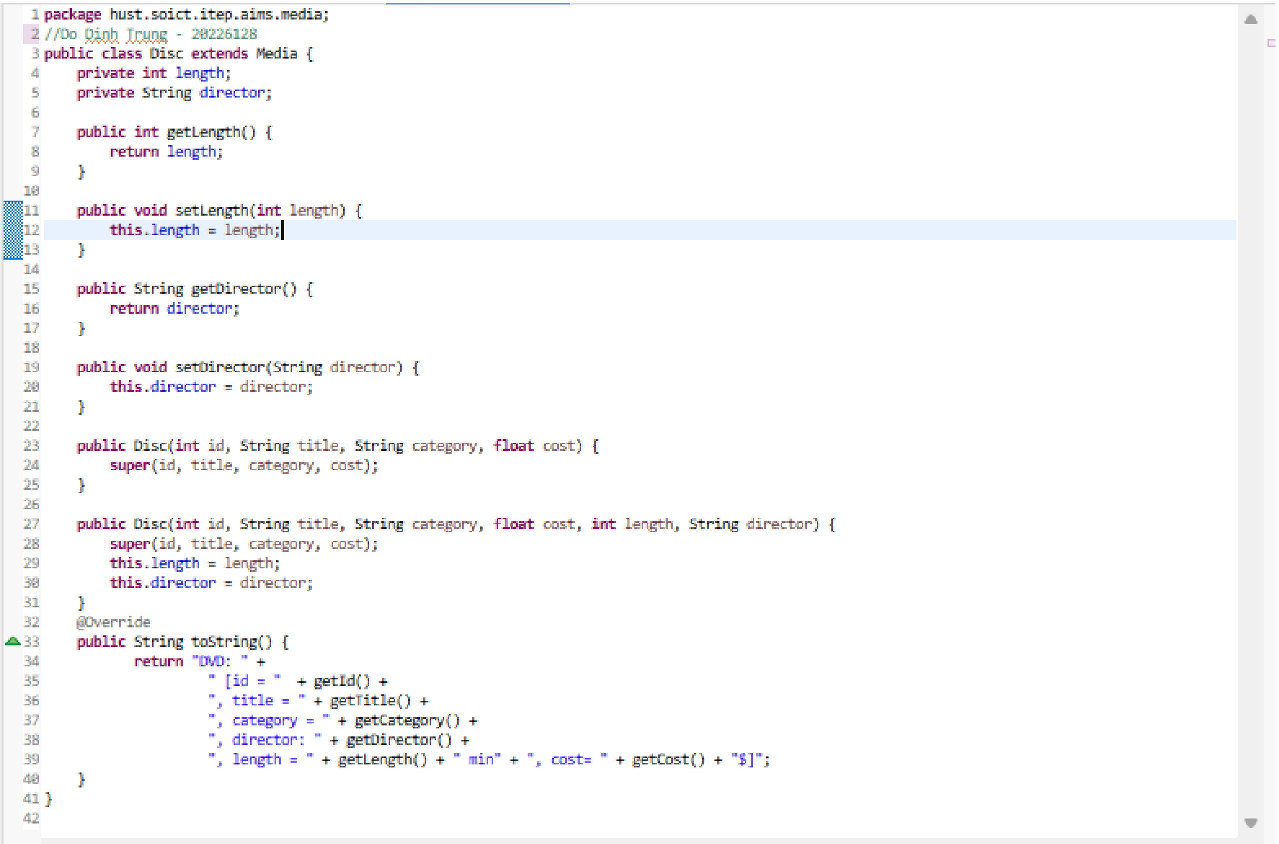


```
    public float getCost() {
        return cost;
    }

    public void setCost(float cost) {
        this.cost = cost;
    }
    //Override method equals DoDinhTrung-20226128
    @Override
    public boolean equals(Object obj) {
        // Kiểm tra nếu cùng tham chiếu
        if (this == obj) return true;
        // Kiểm tra nếu obj là null hoặc không cùng kiểu Media
        if (obj == null || getClass() != obj.getClass()) return false;
        // Ép kiểu obj về Media
        Media media = (Media) obj;
        // So sánh theo tiêu chí tiêu đề (title)
        return title != null && title.equals(media.title);
    }

    //Override method toString DoDinhTrun-20226128
    @Override
    public String toString() {
        return "Media: [ID: " + id + ", Title: " + title + ", Category: " + category + ", Cost: " + cost + "$]";
    }

    @Override
    public int compareTo(Media other) {
        int titleComparison = this.title.compareToIgnoreCase(other.title);
        if (titleComparison != 0) {
            return titleComparison;
        }
        return Float.compare(other.cost, this.cost);
    }

}
```

Figure 2.2: Media Class 2

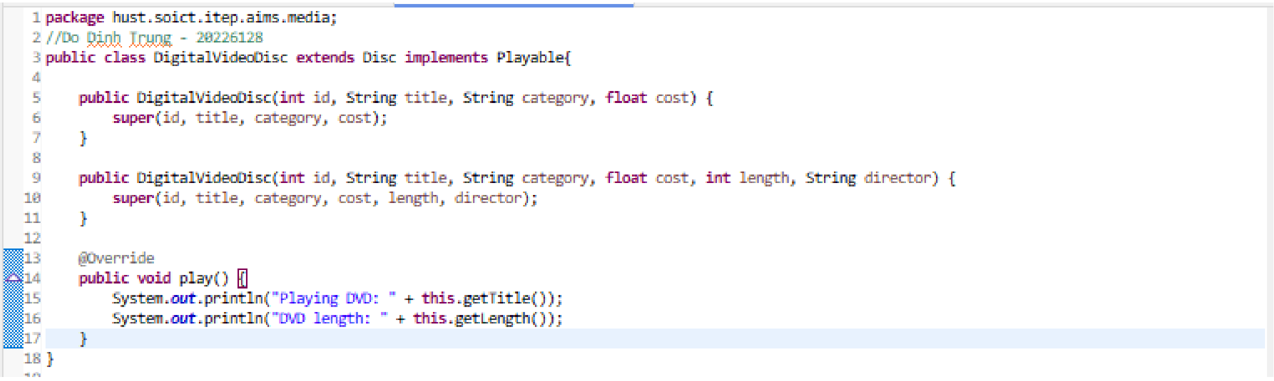
3 Creating the CompactDisc class

3.1 Create the Disc class extending the Media class



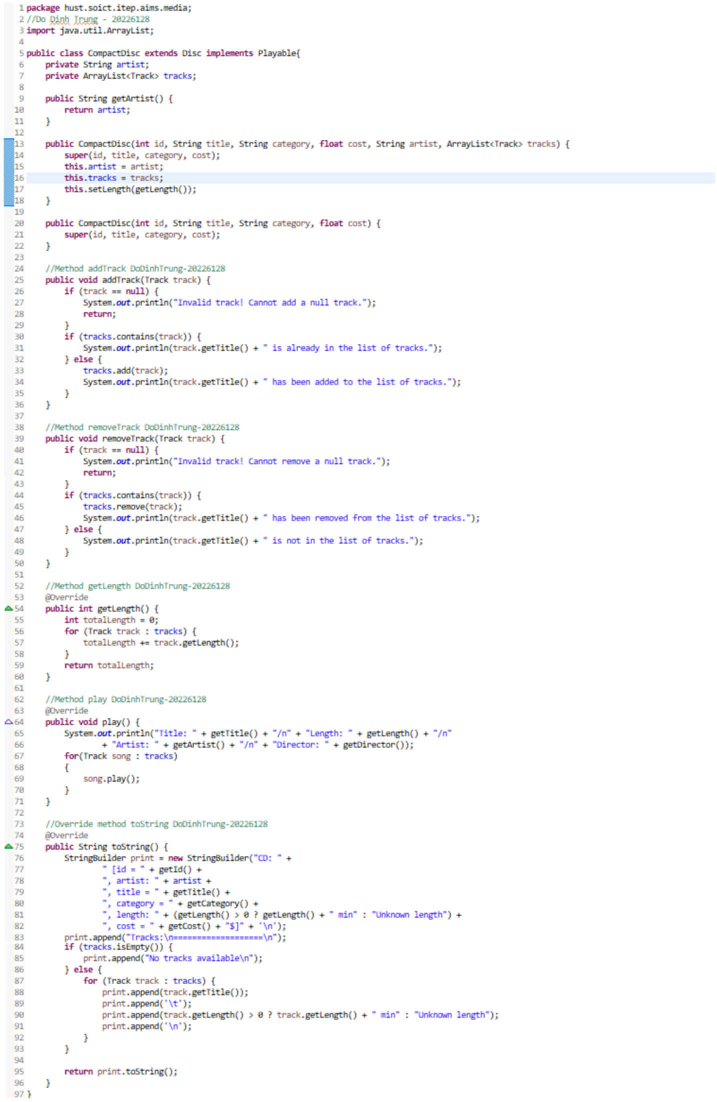
```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
public class Disc extends Media {
    private int length;
    private String director;

    public int getLength() {
        return length;
    }

    public void setLength(int length) {
        this.length = length;
    }

    public String getDirector() {
        return director;
    }

    public void setDirector(String director) {
        this.director = director;
    }

    public Disc(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }

    public Disc(int id, String title, String category, float cost, int length, String director) {
        super(id, title, category, cost);
        this.length = length;
        this.director = director;
    }
    @Override
    public String toString() {
            return "DVD: " +
                        " [id = "  + getId() +
                        ", title = " + getTitle() +
                        ", category = " + getCategory() +
                        ", director: " + getDirector() +
                        ", length = " + getLength() + " min" + ", cost= " + getCost() + "$]";
    }
}

```

Figure 3.1: Disc Class



```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
public class DigitalVideoDisc extends Disc implements Playable{

    public DigitalVideoDisc(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }

    public DigitalVideoDisc(int id, String title, String category, float cost, int length, String director) {
        super(id, title, category, cost, length, director);
    }

    @Override
    public void play() {
        System.out.println("Playing DVD: " + this.getTitle());
        System.out.println("DVD length: " + this.getLength());
    }
}
```

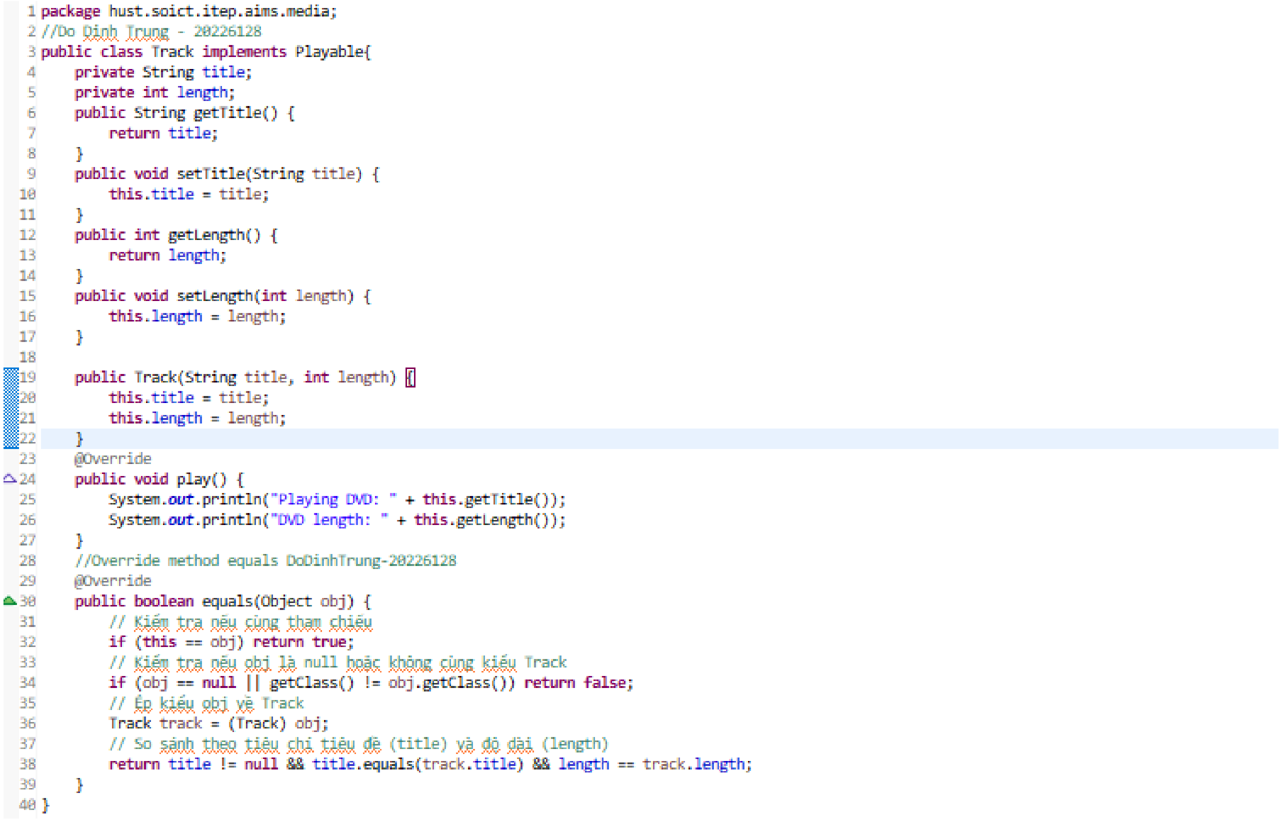
Figure 3.2: DigitalVideoDisc Class



```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
import java.util.ArrayList;

public class CompactDisc extends Disc implements Playable{
    private String artist;
    private ArrayList<Track> tracks;

    public String getArtist() {
        return artist;
    }

    public CompactDisc(int id, String title, String category, float cost, String artist, ArrayList<Track> tracks) {
        super(id, title, category, cost);
        this.artist = artist;
        this.tracks = tracks;
        this.setLength(getLength());
    }

    public CompactDisc(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }

    //Method addTrack DoDinhTrung-20226128
    public void addTrack(Track track) {
        if (track == null) {
            System.out.println("Invalid track! Cannot add a null track.");
            return;
        }
        if (tracks.contains(track)) {
            System.out.println(track.getTitle() + " is already in the list of tracks.");
        } else {
            tracks.add(track);
            System.out.println(track.getTitle() + " has been added to the list of tracks.");
        }
    }

    //Method removeTrack DoDinhTrung-20226128
    public void removeTrack(Track track) {
        if (track == null) {
            System.out.println("Invalid track! Cannot remove a null track.");
            return;
        }
        if (tracks.contains(track)) {
            tracks.remove(track);
            System.out.println(track.getTitle() + " has been removed from the list of tracks.");
        } else {
            System.out.println(track.getTitle() + " is not in the list of tracks.");
        }
    }

    //Method getLength DoDinhTrung-20226128
    @Override
    public int getLength() {
        int totalLength = 0;
        for (Track track : tracks) {
            totalLength += track.getLength();
        }
        return totalLength;
    }

    //Method play DoDinhTrung-20226128
    @Override
    public void play() {
        System.out.println("Title: " + getTitle() + "/n" + "Length: " + getLength() + "/n"
                + "Artist: " + getArtist() + "/n" + "Director: " + getDirector());
        for(Track song : tracks)
        {
            song.play();
        }
    }

    //Override method toString DoDinhTrung-20226128
    @Override
    public String toString() {
        StringBuilder print = new StringBuilder("CD: " +
                " [id = " + getId() +
                ", artist: " + artist +
                ", title = " + getTitle() +
                ", category = " + getCategory() +
                ", length: " + (getLength() > 0 ? getLength() + " min" : "Unknown length") +
                ", cost = " + getCost() + "$]" + '\n');
        print.append("Tracks:\n===================\n");
        if (tracks.isEmpty()) {
            print.append("No tracks available\n");
        } else {
            for (Track track : tracks) {
                print.append(track.getTitle());
                print.append('\t');
                print.append(track.getLength() > 0 ? track.getLength() + " min" : "Unknown length");
                print.append('\n');
            }
        }

        return print.toString();
    }
}
```

Figure 3.3: CompactDisc Class

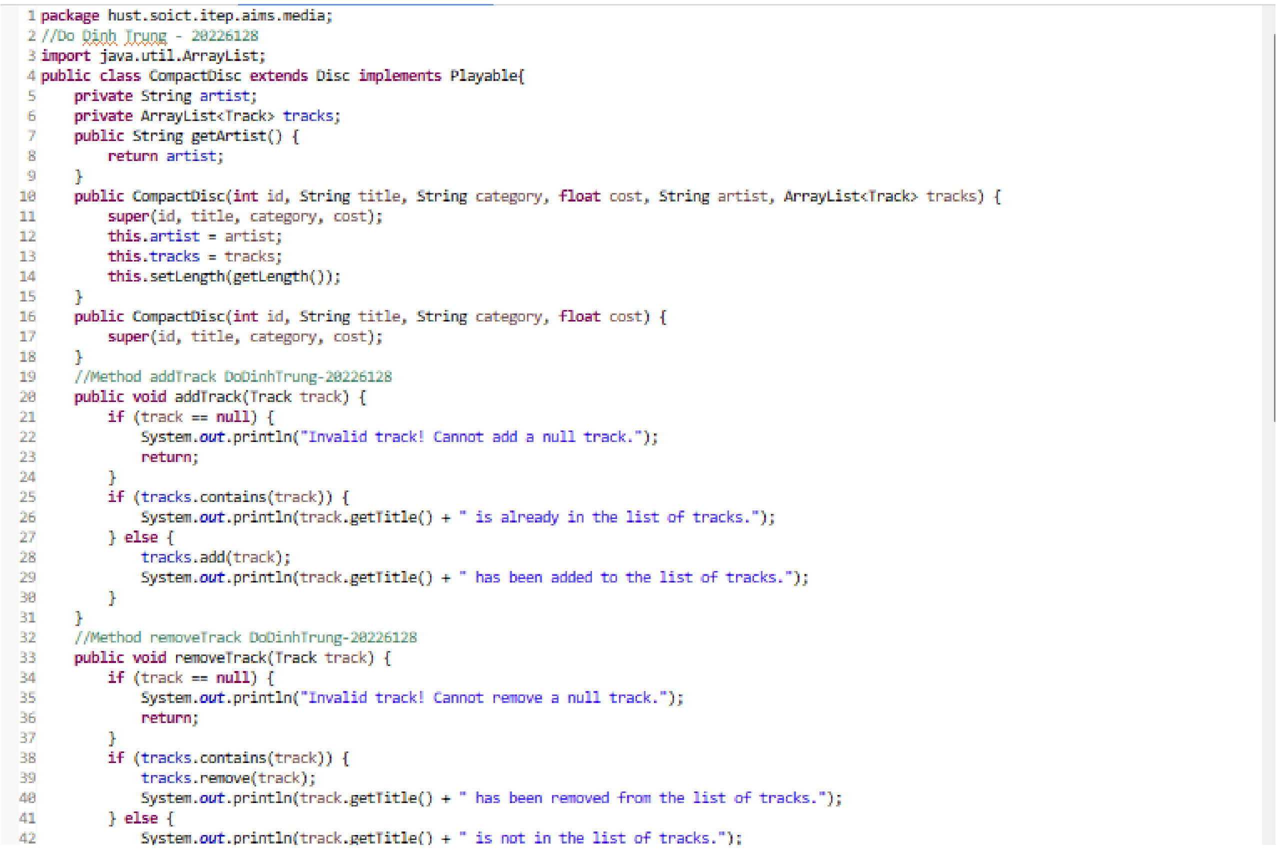
3.2 Create the Track class which models a track on a compact disc and will store information incuding the title and length of the track.



```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
public class Track implements Playable{
    private String title;
    private int length;
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
    public int getLength() {
        return length;
    }
    public void setLength(int length) {
        this.length = length;
    }

    public Track(String title, int length) {
        this.title = title;
        this.length = length;
    }
    @Override
    public void play() {
        System.out.println("Playing DVD: " + this.getTitle());
        System.out.println("DVD length: " + this.getLength());
    }
    //Override method equals DoDinhTrung-20226128
    @Override
    public boolean equals(Object obj) {
        // Kiểm tra nếu cùng tham chiếu
        if (this == obj) return true;
        // Kiểm tra nếu obj là null hoặc không cùng kiểu Track
        if (obj == null || getClass() != obj.getClass()) return false;
        // Ép kiểu obj về Track
        Track track = (Track) obj;
        // So sánh theo tiêu chí tiêu đề (title) và độ dài (length)
        return title != null && title.equals(track.title) && length == track.length;
    }
}
```

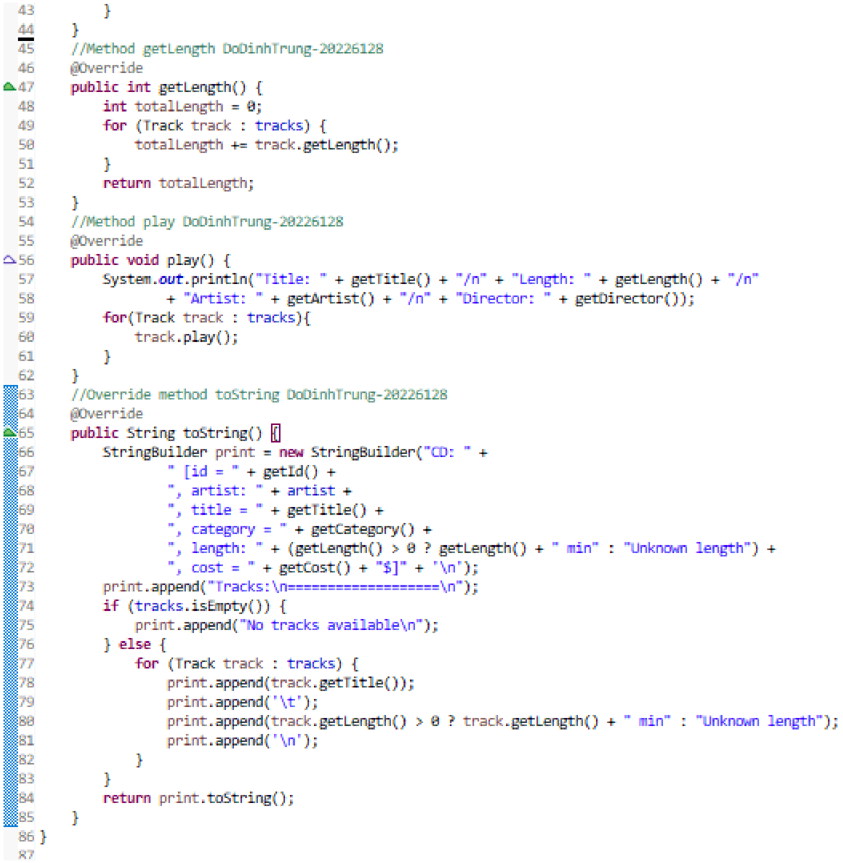
Figure 3.4: Track Class

3.3 Open the CompactDisc class



```
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
import java.util.ArrayList;
public class CompactDisc extends Disc implements Playable{
    private String artist;
    private ArrayList<Track> tracks;
    public String getArtist() {
        return artist;
    }
    public CompactDisc(int id, String title, String category, float cost, String artist, ArrayList<Track> tracks) {
        super(id, title, category, cost);
        this.artist = artist;
        this.tracks = tracks;
        this.setLength(getLength());
    }
    public CompactDisc(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }
    //Method addTrack DoDinhTrung-20226128
    public void addTrack(Track track) {
        if (track == null) {
            System.out.println("Invalid track! Cannot add a null track.");
            return;
        }
        if (tracks.contains(track)) {
            System.out.println(track.getTitle() + " is already in the list of tracks.");
        } else {
            tracks.add(track);
            System.out.println(track.getTitle() + " has been added to the list of tracks.");
        }
    }
    //Method removeTrack DoDinhTrung-20226128
    public void removeTrack(Track track) {
        if (track == null) {
            System.out.println("Invalid track! Cannot remove a null track.");
            return;
        }
        if (tracks.contains(track)) {
            tracks.remove(track);
            System.out.println(track.getTitle() + " has been removed from the list of tracks.");
        } else {
            System.out.println(track.getTitle() + " is not in the list of tracks.");
```

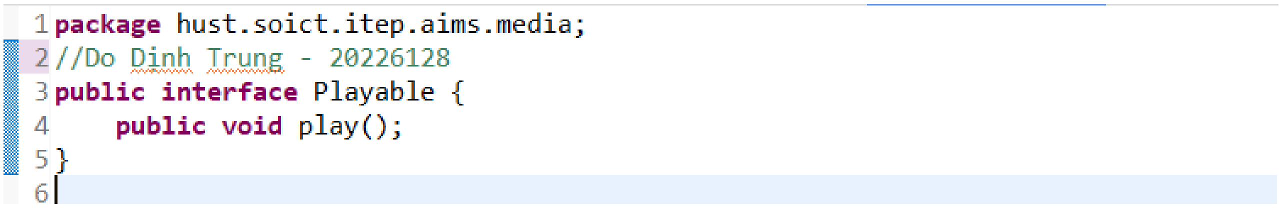
Figure 3.5: CompactDisc Class 1



```java
        }
    }
    //Method getLength DoDinhTrung-20226128
    @Override
    public int getLength() {
        int totalLength = 0;
        for (Track track : tracks) {
            totalLength += track.getLength();
        }
        return totalLength;
    }
    //Method play DoDinhTrung-20226128
    @Override
    public void play() {
        System.out.println("Title: " + getTitle() + "/n" + "Length: " + getLength() + "/n"
                + "Artist: " + getArtist() + "/n" + "Director: " + getDirector());
        for(Track track : tracks){
            track.play();
        }
    }
    //Override method toString DoDinhTrung-20226128
    @Override
    public String toString() {
        StringBuilder print = new StringBuilder("CD: " +
                " [id = " + getId() +
                ", artist: " + artist +
                ", title = " + getTitle() +
                ", category = " + getCategory() +
                ", length: " + (getLength() > 0 ? getLength() + " min" : "Unknown length") +
                ", cost = " + getCost() + "$]" + '\n');
        print.append("Tracks:\n===================\n");
        if (tracks.isEmpty()) {
            print.append("No tracks available\n");
        } else {
            for (Track track : tracks) {
                print.append(track.getTitle());
                print.append('\t');
                print.append(track.getLength() > 0 ? track.getLength() + " min" : "Unknown length");
                print.append('\n');
            }
        }
        return print.toString();
    }
}
```

Figure 3.6: CompactDisc Class 2

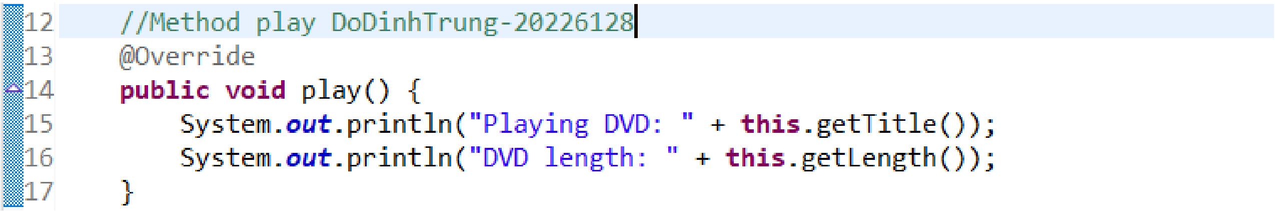
4 Create the Playable interface



```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
public interface Playable {
    public void play();
}

```

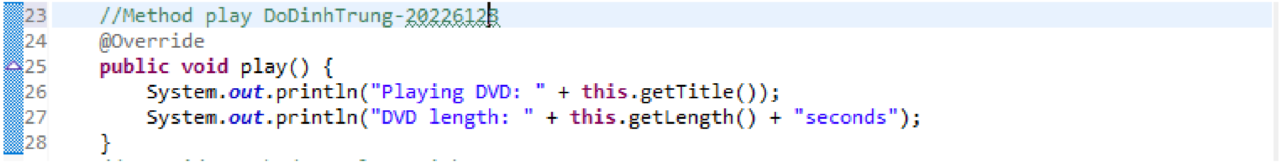
Figure 4.1: Playable interface

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



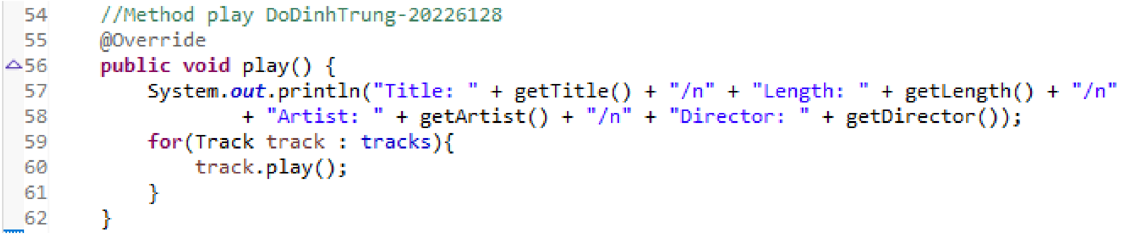
```java
    //Method play DoDinhTrung-20226128
    @Override
    public void play() {
        System.out.println("Playing DVD: " + this.getTitle());
        System.out.println("DVD length: " + this.getLength());
    }
```

Figure 4.2: Method play() của DigitalVideoDisc



```java
    //Method play DoDinhTrung-20226128
    @Override
    public void play() {
        System.out.println("Playing DVD: " + this.getTitle());
        System.out.println("DVD length: " + this.getLength() + "seconds");
    }
```

Figure 4.3: Method play() của Track

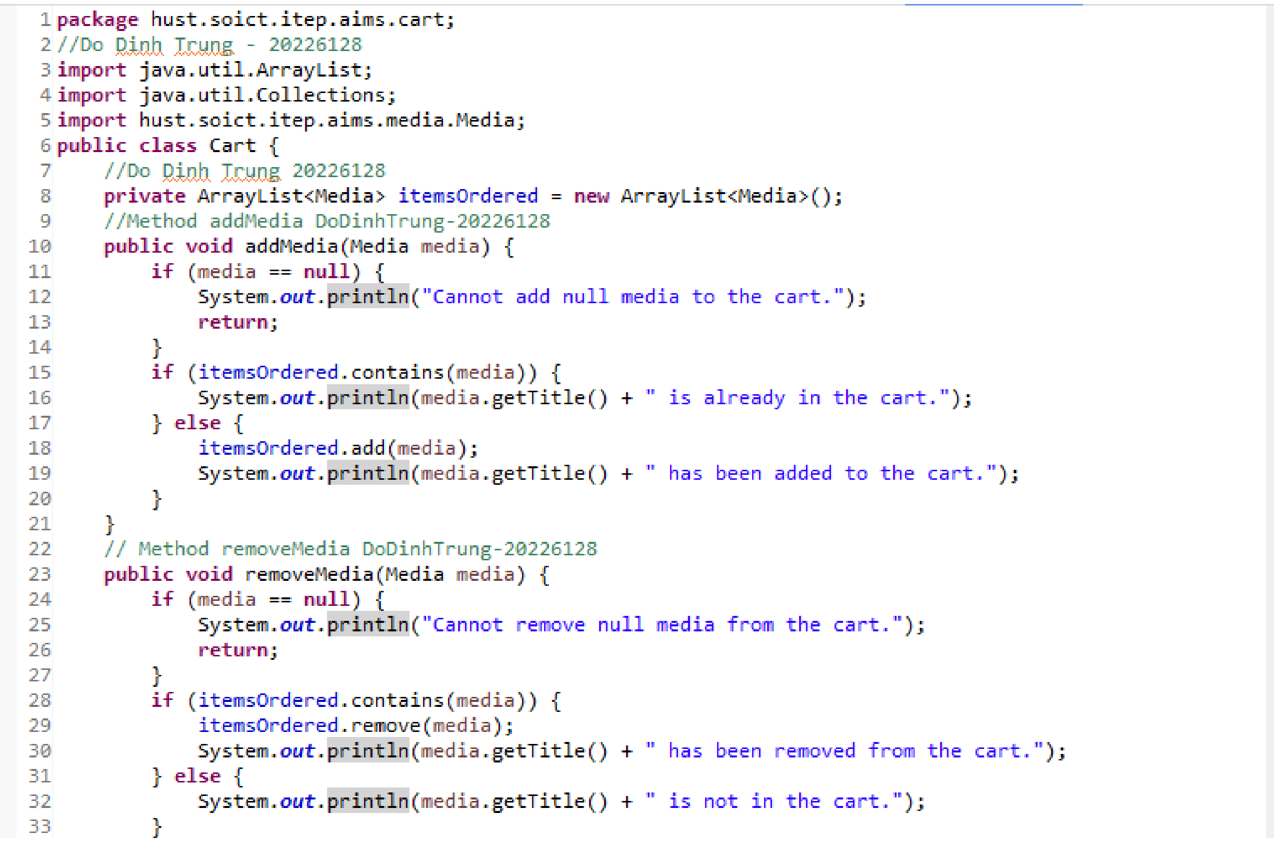


```java
    //Method play DoDinhTrung-20226128
    @Override
    public void play() {
        System.out.println("Title: " + getTitle() + "/n" + "Length: " + getLength() + "/n"
                + "Artist: " + getArtist() + "/n" + "Director: " + getDirector());
        for(Track track : tracks){
            track.play();
        }
    }
```

Figure 4.4: Method play() của CompactDisc

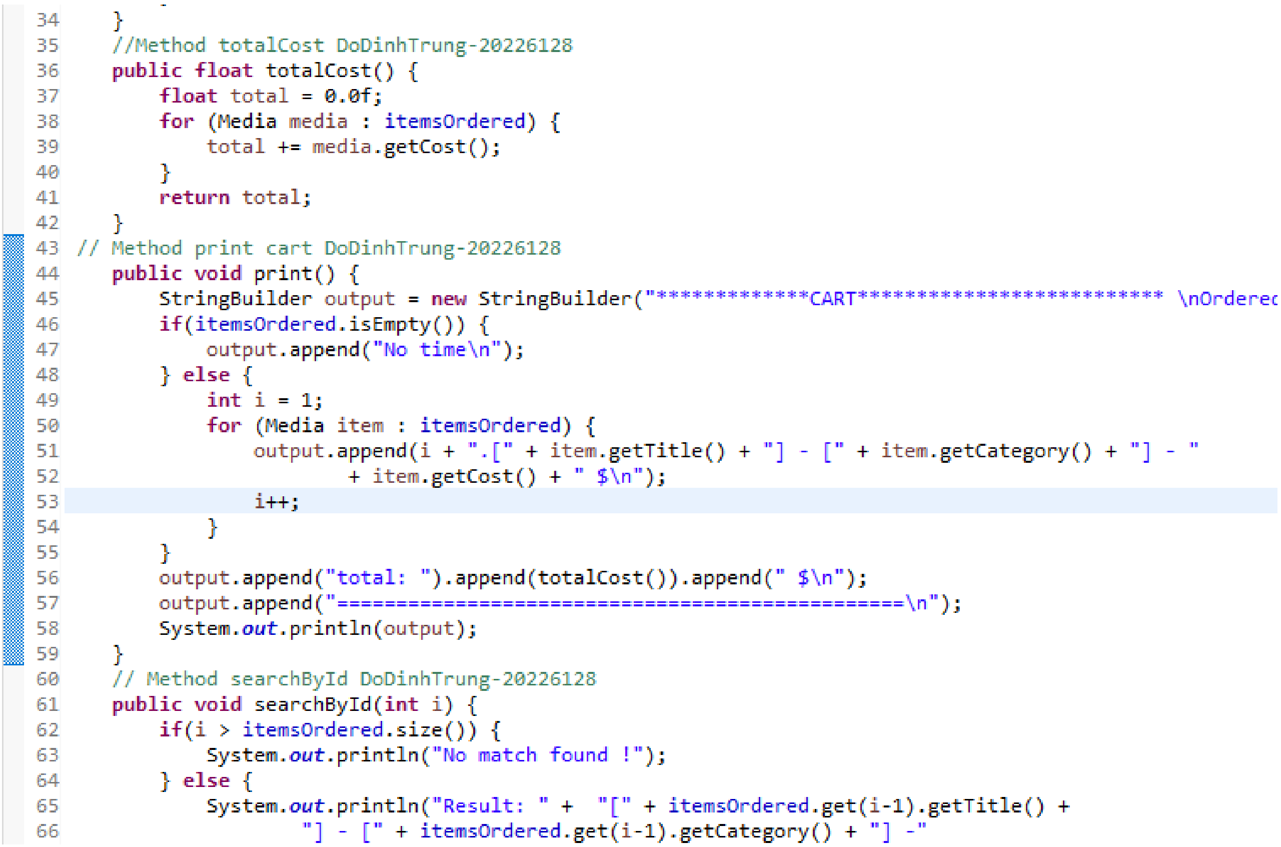
5 Update the Cart class to work with Media

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.



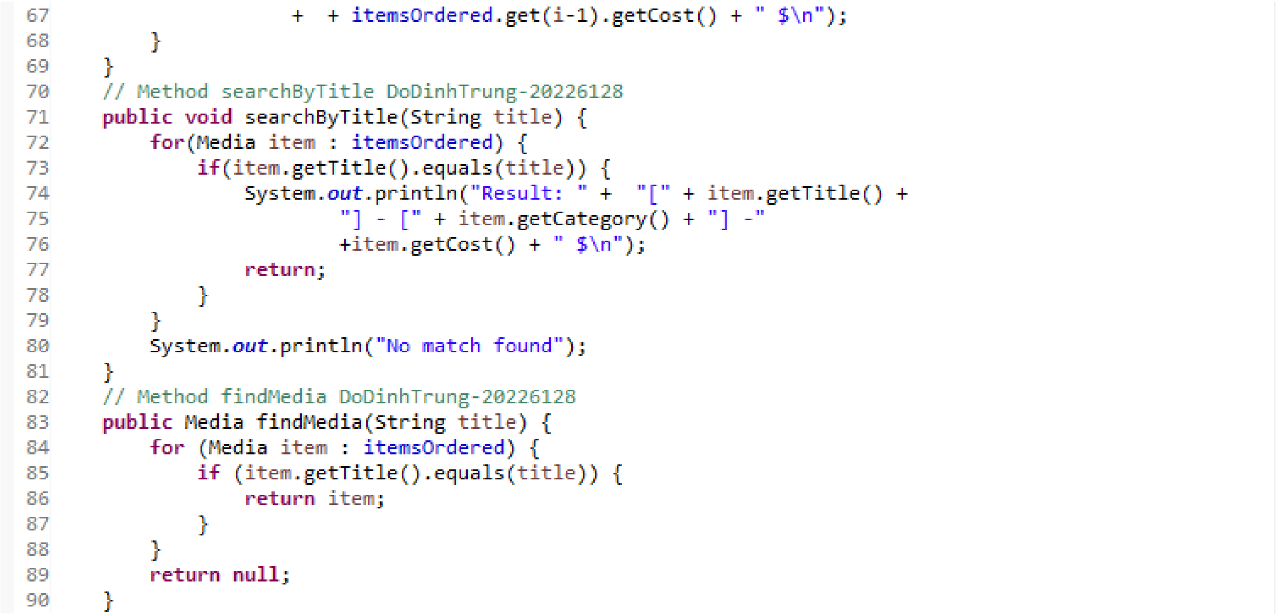
```java
package hust.soict.itep.aims.cart;
//Do Dinh Trung - 20226128
import java.util.ArrayList;
import java.util.Collections;
import hust.soict.itep.aims.media.Media;
public class Cart {
    //Do Dinh Trung 20226128
    private ArrayList<Media> itemsOrdered = new ArrayList<Media>();
    //Method addMedia DoDinhTrung-20226128
    public void addMedia(Media media) {
        if (media == null) {
            System.out.println("Cannot add null media to the cart.");
            return;
        }
        if (itemsOrdered.contains(media)) {
            System.out.println(media.getTitle() + " is already in the cart.");
        } else {
            itemsOrdered.add(media);
            System.out.println(media.getTitle() + " has been added to the cart.");
        }
    }
    // Method removeMedia DoDinhTrung-20226128
    public void removeMedia(Media media) {
        if (media == null) {
            System.out.println("Cannot remove null media from the cart.");
            return;
        }
        if (itemsOrdered.contains(media)) {
            itemsOrdered.remove(media);
            System.out.println(media.getTitle() + " has been removed from the cart.");
        } else {
            System.out.println(media.getTitle() + " is not in the cart.");
        }
```

Figure 5.1: Cart Class 1



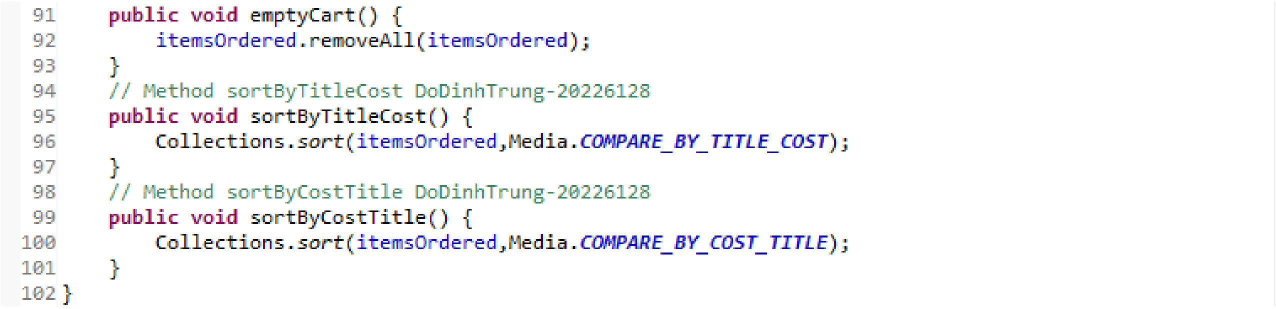
```
    }
    //Method totalCost DoDinhTrung-20226128
    public float totalCost() {
        float total = 0.0f;
        for (Media media : itemsOrdered) {
            total += media.getCost();
        }
        return total;
    }
// Method print cart DoDinhTrung-20226128
    public void print() {
        StringBuilder output = new StringBuilder("*************CART************************** \nOrderec
        if(itemsOrdered.isEmpty()) {
            output.append("No time\n");
        } else {
            int i = 1;
            for (Media item : itemsOrdered) {
                output.append(i + ".[" + item.getTitle() + "] - [" + item.getCategory() + "] - "
                        + item.getCost() + " $\n");
                i++;
            }
        }
        output.append("total: ").append(totalCost()).append(" $\n");
        output.append("================================================\n");
        System.out.println(output);
    }
    // Method searchById DoDinhTrung-20226128
    public void searchById(int i) {
        if(i > itemsOrdered.size()) {
            System.out.println("No match found !");
        } else {
            System.out.println("Result: " +  "[" + itemsOrdered.get(i-1).getTitle() +
                    "] - [" + itemsOrdered.get(i-1).getCategory() + "] -"
```

Figure 5.2: Cart Class 2



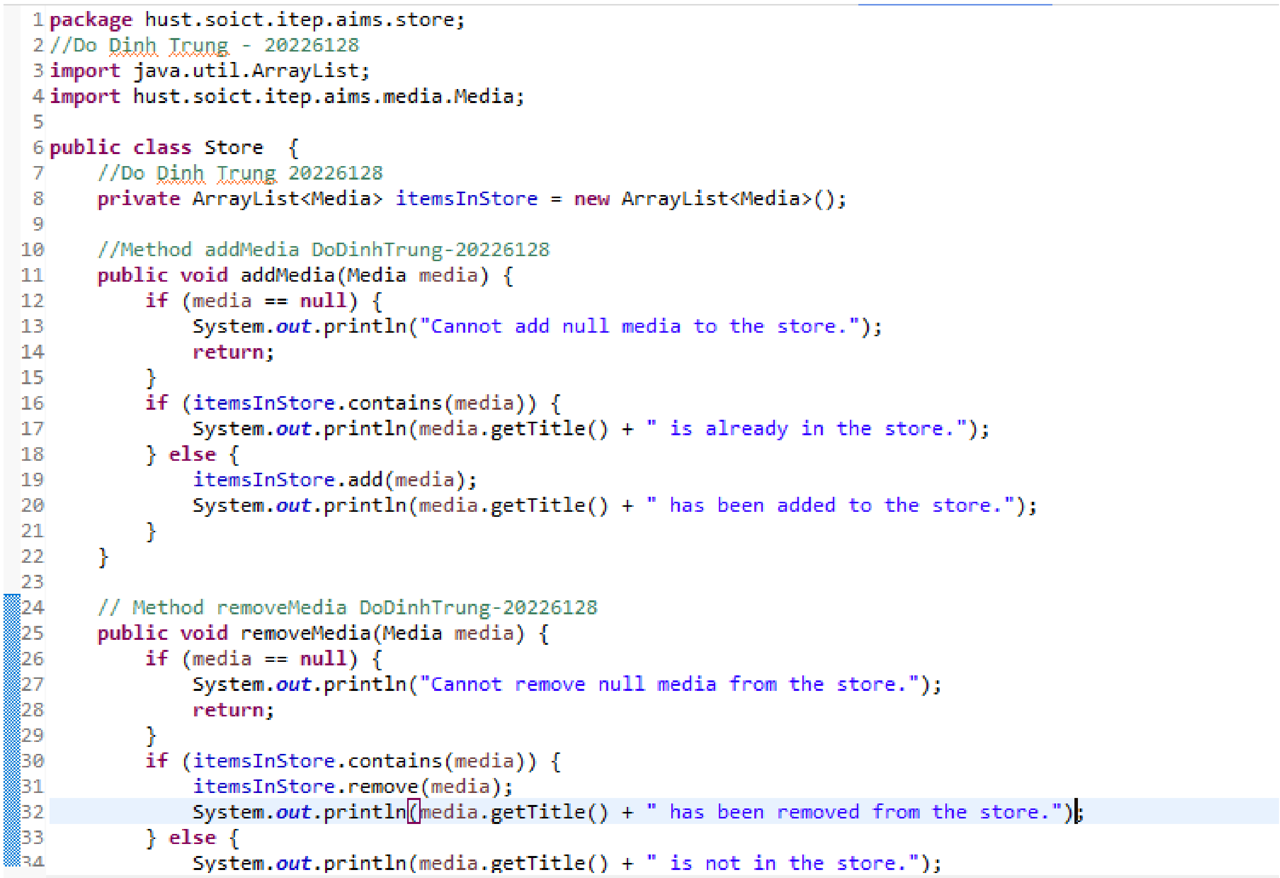
```
                    +  + itemsOrdered.get(i-1).getCost() + " $\n");
        }
    }
    // Method searchByTitle DoDinhTrung-20226128
    public void searchByTitle(String title) {
        for(Media item : itemsOrdered) {
            if(item.getTitle().equals(title)) {
                System.out.println("Result: " +  "[" + item.getTitle() +
                        "] - [" + item.getCategory() + "] -"
                        +item.getCost() + " $\n");
                return;
            }
        }
        System.out.println("No match found");
    }
    // Method findMedia DoDinhTrung-20226128
    public Media findMedia(String title) {
        for (Media item : itemsOrdered) {
            if (item.getTitle().equals(title)) {
                return item;
            }
        }
        return null;
    }
```

Figure 5.3: Cart Class 3



```java
    public void emptyCart() {
        itemsOrdered.removeAll(itemsOrdered);
    }
    // Method sortByTitleCost DoDinhTrung-20226128
    public void sortByTitleCost() {
        Collections.sort(itemsOrdered,Media.COMPARE_BY_TITLE_COST);
    }
    // Method sortByCostTitle DoDinhTrung-20226128
    public void sortByCostTitle() {
        Collections.sort(itemsOrdered,Media.COMPARE_BY_COST_TITLE);
    }
}
```

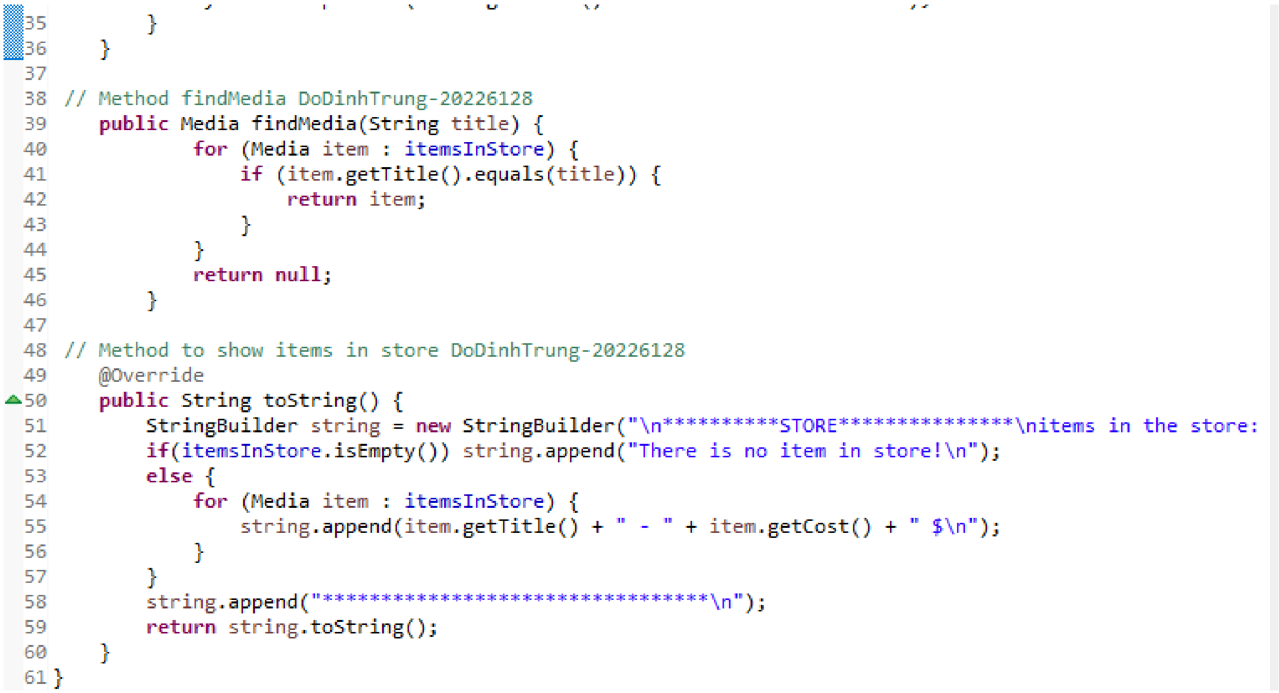
Figure 5.4: Cart Class 4



```java
package hust.soict.itep.aims.store;
//Do Dinh Trung - 20226128
import java.util.ArrayList;
import hust.soict.itep.aims.media.Media;

public class Store  {
    //Do Dinh Trung 20226128
    private ArrayList<Media> itemsInStore = new ArrayList<Media>();

    //Method addMedia DoDinhTrung-20226128
    public void addMedia(Media media) {
        if (media == null) {
            System.out.println("Cannot add null media to the store.");
            return;
        }
        if (itemsInStore.contains(media)) {
            System.out.println(media.getTitle() + " is already in the store.");
        } else {
            itemsInStore.add(media);
            System.out.println(media.getTitle() + " has been added to the store.");
        }
    }

    // Method removeMedia DoDinhTrung-20226128
    public void removeMedia(Media media) {
        if (media == null) {
            System.out.println("Cannot remove null media from the store.");
            return;
        }
        if (itemsInStore.contains(media)) {
            itemsInStore.remove(media);
            System.out.println(media.getTitle() + " has been removed from the store.");
        } else {
            System.out.println(media.getTitle() + " is not in the store.");
```

6 Update the Store class to work with Media

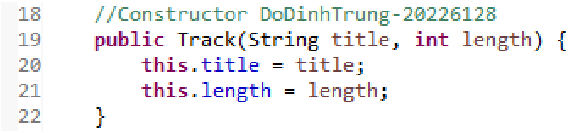
Figure 6.1: Store Class 1



```
        }
    }

// Method findMedia DoDinhTrung-20226128
    public Media findMedia(String title) {
            for (Media item : itemsInStore) {
                if (item.getTitle().equals(title)) {
                    return item;
                }
            }
            return null;
        }

// Method to show items in store DoDinhTrung-20226128
    @Override
    public String toString() {
        StringBuilder string = new StringBuilder("\n**********STORE***************\nitems in the store:
        if(itemsInStore.isEmpty()) string.append("There is no item in store!\n");
        else {
            for (Media item : itemsInStore) {
                string.append(item.getTitle() + " - " + item.getCost() + " $\n");
            }
        }
        string.append("*********************************\n");
        return string.toString();
    }
}
```

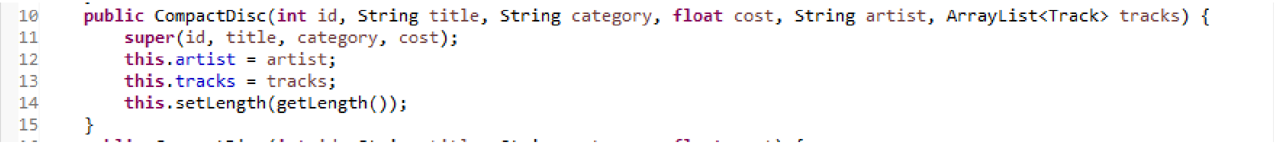
Figure 6.2: Store Class 2

7 Constructors of whole classes and parent classes



```java
    //Constructor DoDinhTrung-20226128
    public Track(String title, int length) {
        this.title = title;
        this.length = length;
    }
```

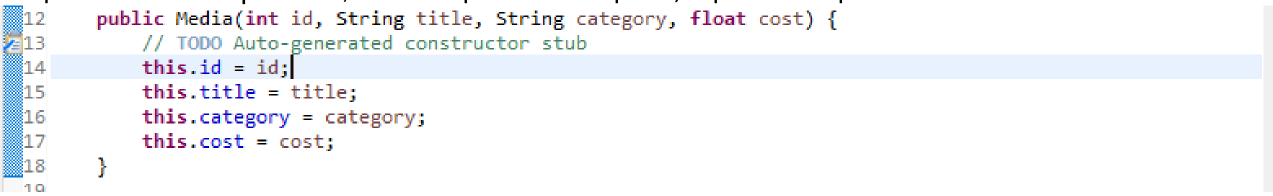
Figure 7.1: Constructor Track Class



```java
    public CompactDisc(int id, String title, String category, float cost, String artist, ArrayList<Track> tracks) {
        super(id, title, category, cost);
        this.artist = artist;
        this.tracks = tracks;
        this.setLength(getLength());
    }
```

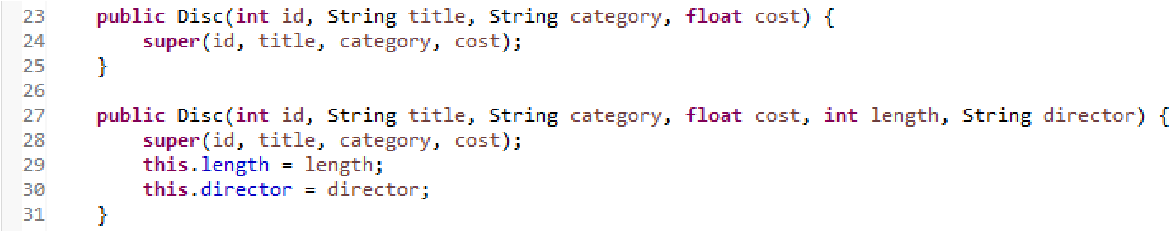
Figure 7.2: Constructor CompactDisc Class

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



```java
    public Media(int id, String title, String category, float cost) {
        // TODO Auto-generated constructor stub
        this.id = id;
        this.title = title;
        this.category = category;
        this.cost = cost;
    }
```

Figure 7.3: Constructor Media Class

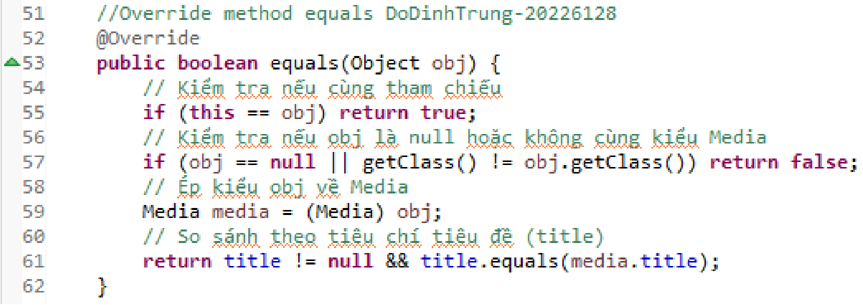


```java
    public Disc(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }

    public Disc(int id, String title, String category, float cost, int length, String director) {
        super(id, title, category, cost);
        this.length = length;
        this.director = director;
    }
```

Figure 7.4: Constructor Disc Class

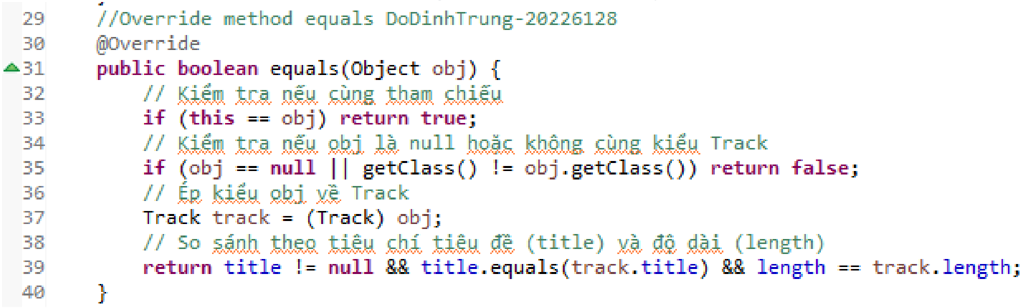
8 Unique item in a list

Để tránh trùng lặp các phần tử media trong giỏ hàng hoặc các track trong một đĩa CD, chúng ta có thể ghi đè lại phương thức equals() mặc định kế thừa từ lớp Object. Việc này cho phép so sánh bản chất thay vì so sánh vị trí ô nhớ của các đối tượng, qua đó ngăn chặn thêm các phần tử bị trùng lắp vào danh sách.



```java
    //Override method equals DoDinhTrung-20226128
    @Override
    public boolean equals(Object obj) {
        // Kiểm tra nếu cùng tham chiếu
        if (this == obj) return true;
        // Kiểm tra nếu obj là null hoặc không cùng kiểu Media
        if (obj == null || getClass() != obj.getClass()) return false;
        // Ép kiểu obj về Media
        Media media = (Media) obj;
        // So sánh theo tiêu chí tiêu đề (title)
        return title != null && title.equals(media.title);
    }
```

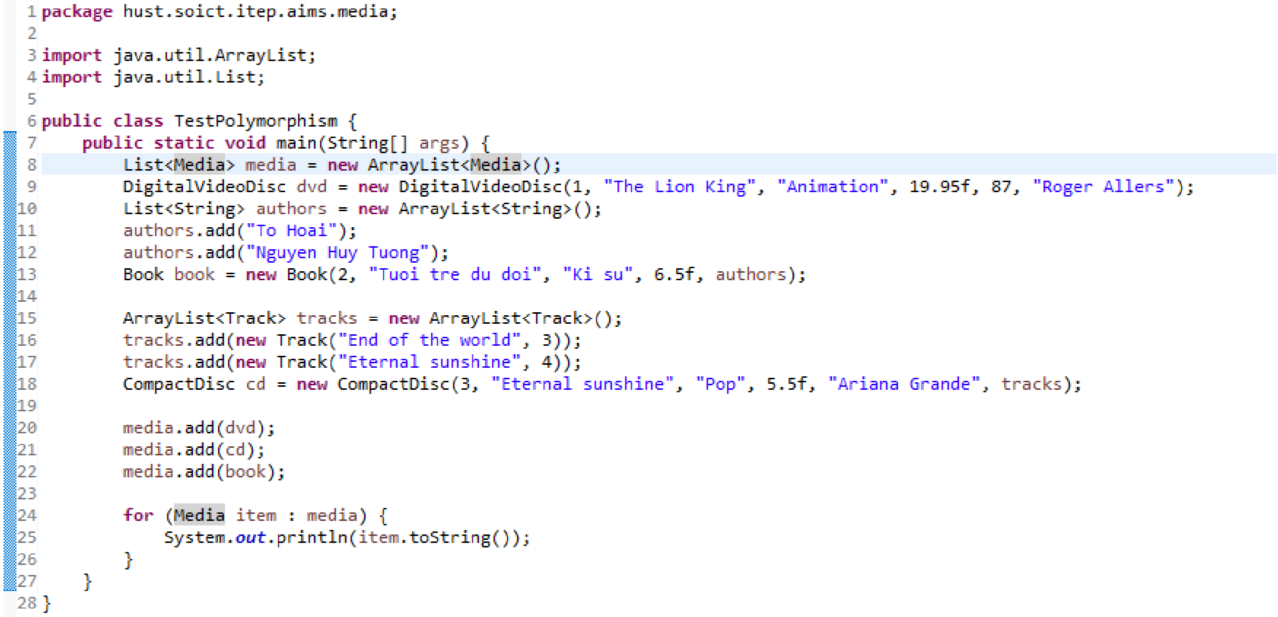
Figure 8.1: Override equals in Media Class



```java
    //Override method equals DoDinhTrung-20226128
    @Override
    public boolean equals(Object obj) {
        // Kiểm tra nếu cùng tham chiếu
        if (this == obj) return true;
        // Kiểm tra nếu obj là null hoặc không cùng kiểu Track
        if (obj == null || getClass() != obj.getClass()) return false;
        // Ép kiểu obj về Track
        Track track = (Track) obj;
        // So sánh theo tiêu chí tiêu đề (title) và độ dài (length)
        return title != null && title.equals(track.title) && length == track.length;
    }
```

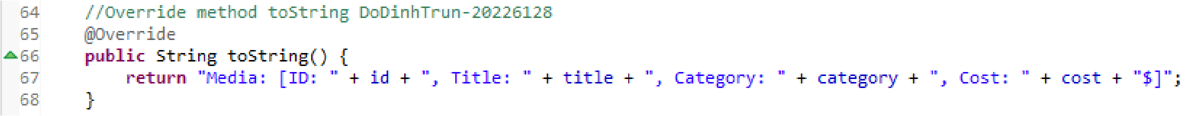
Figure 8.2: Override equals in Track Class

9 Polymorphism with toString() method



```java
package hust.soict.itep.aims.media;

import java.util.ArrayList;
import java.util.List;

public class TestPolymorphism {
    public static void main(String[] args) {
        List<Media> media = new ArrayList<Media>();
        DigitalVideoDisc dvd = new DigitalVideoDisc(1, "The Lion King", "Animation", 19.95f, 87, "Roger Allers");
        List<String> authors = new ArrayList<String>();
        authors.add("To Hoai");
        authors.add("Nguyen Huy Tuong");
        Book book = new Book(2, "Tuoi tre du doi", "Ki su", 6.5f, authors);

        ArrayList<Track> tracks = new ArrayList<Track>();
        tracks.add(new Track("End of the world", 3));
        tracks.add(new Track("Eternal sunshine", 4));
        CompactDisc cd = new CompactDisc(3, "Eternal sunshine", "Pop", 5.5f, "Ariana Grande", tracks);

        media.add(dvd);
        media.add(cd);
        media.add(book);

        for (Media item : media) {
            System.out.println(item.toString());
        }
    }
}
```

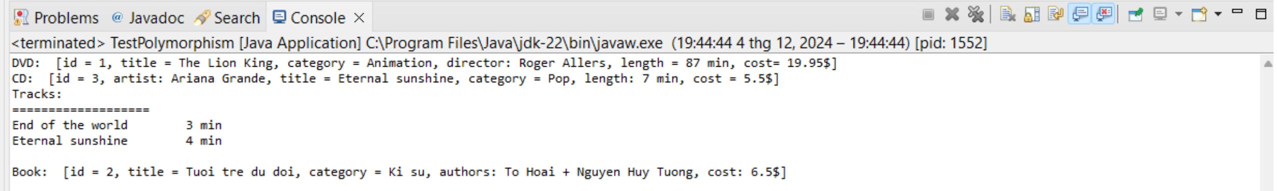
Figure 9.1: Code mô phỏng Polymorphism



```java
    //Override method toString DoDinhTrun-20226128
    @Override
    public String toString() {
        return "Media: [ID: " + id + ", Title: " + title + ", Category: " + category + ", Cost: " + cost + "$]";
    }
```

Figure 9.2: Override toString() in Media Class

Kết quả



```
<terminated> TestPolymorphism [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe  (19:44:44 4 thg 12, 2024 – 19:44:44) [pid: 1552]
DVD:  [id = 1, title = The Lion King, category = Animation, director: Roger Allers, length = 87 min, cost= 19.95$]
CD:  [id = 3, artist: Ariana Grande, title = Eternal sunshine, category = Pop, length: 7 min, cost = 5.5$]
Tracks:
===================
End of the world        3 min
Eternal sunshine        4 min

Book:  [id = 2, title = Tuoi tre du doi, category = Ki su, authors: To Hoai + Nguyen Huy Tuong, cost: 6.5$]
```

Figure 9.3: Result demo Polymorphism

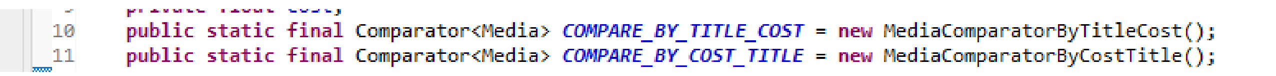
Lớp Media là lớp cơ sở được kế thừa bởi các lớp cụ thể hơn là CompactDisc, DigitalVideoDisc và Book. Khi khởi tạo các đối tượng cd, dvd, book thuộc lớp con rồi gán chúng cho biến kiểu Media, ta áp dụng kỹ thuật gọi là upcasting.

Việc thêm chúng vào danh sách media và duyệt danh sách để in ra thông tin mỗi phần tử bằng phương thức toString() là ví dụ điển hình cho tính đa hình động. Mỗi lớp con có thể cài đặt riêng toString() nên kết quả sẽ khác nhau dựa theo loại đối tượng, mà không cần quan tâm đến kiểu cụ thể của từng phần tử.

10 Sort media in the car

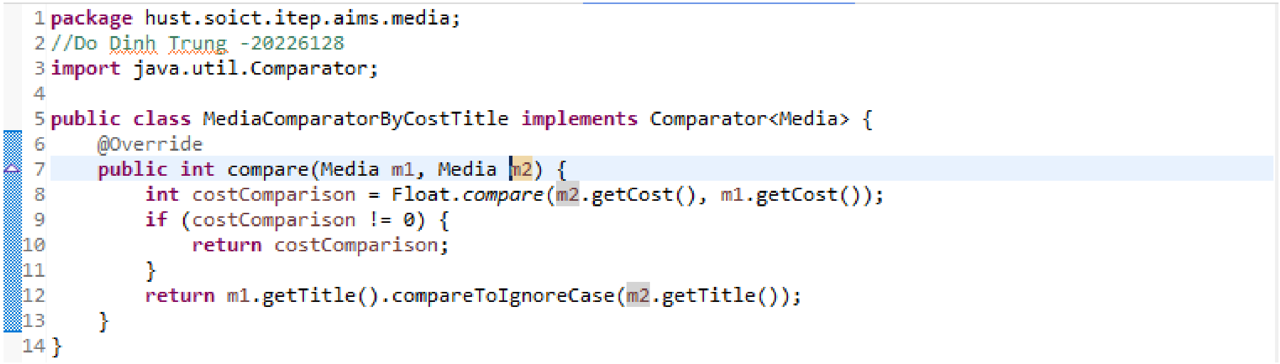
Sắp xếp các media trong giỏ hàng theo hai tiêu chí:

- Bằng title: Hiển thị tất cả các media theo thứ tự bảng chữ cái. Trong trường hợp cùng title, media có cost cao hơn sẽ được hiển thị trước.
- Bằng cost: Hiển thị theo thứ tự cost giảm dần. Trong trường hợp cost như nhau, sắp xếp media theo thứ tự bảng chữ cái



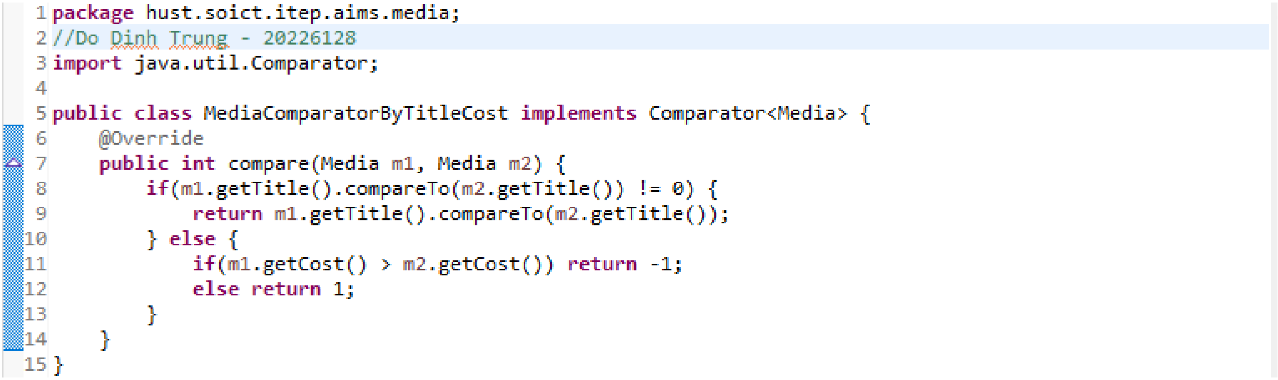
```
    private float cost;
    public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();
    public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();
```

Figure 10.1: Add the comparators as attributes of the Media class



```
package hust.soict.itep.aims.media;
//Do Dinh Trung -20226128
import java.util.Comparator;

public class MediaComparatorByCostTitle implements Comparator<Media> {
    @Override
    public int compare(Media m1, Media m2) {
        int costComparison = Float.compare(m2.getCost(), m1.getCost());
        if (costComparison != 0) {
            return costComparison;
        }
        return m1.getTitle().compareToIgnoreCase(m2.getTitle());
    }
}
```

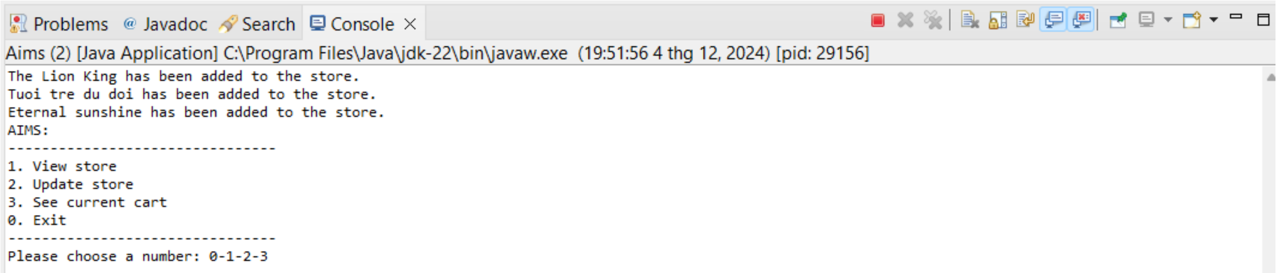
Figure 10.2: MediaComparatorByCostTitle Class



```java
package hust.soict.itep.aims.media;
//Do Dinh Trung - 20226128
import java.util.Comparator;

public class MediaComparatorByTitleCost implements Comparator<Media> {
    @Override
    public int compare(Media m1, Media m2) {
        if(m1.getTitle().compareTo(m2.getTitle()) != 0) {
            return m1.getTitle().compareTo(m2.getTitle());
        } else {
            if(m1.getCost() > m2.getCost()) return -1;
            else return 1;
        }
    }
}
```

Figure 10.3: MediaComparatorByTitleCost Class

11 Create a complete console application in the Aims class

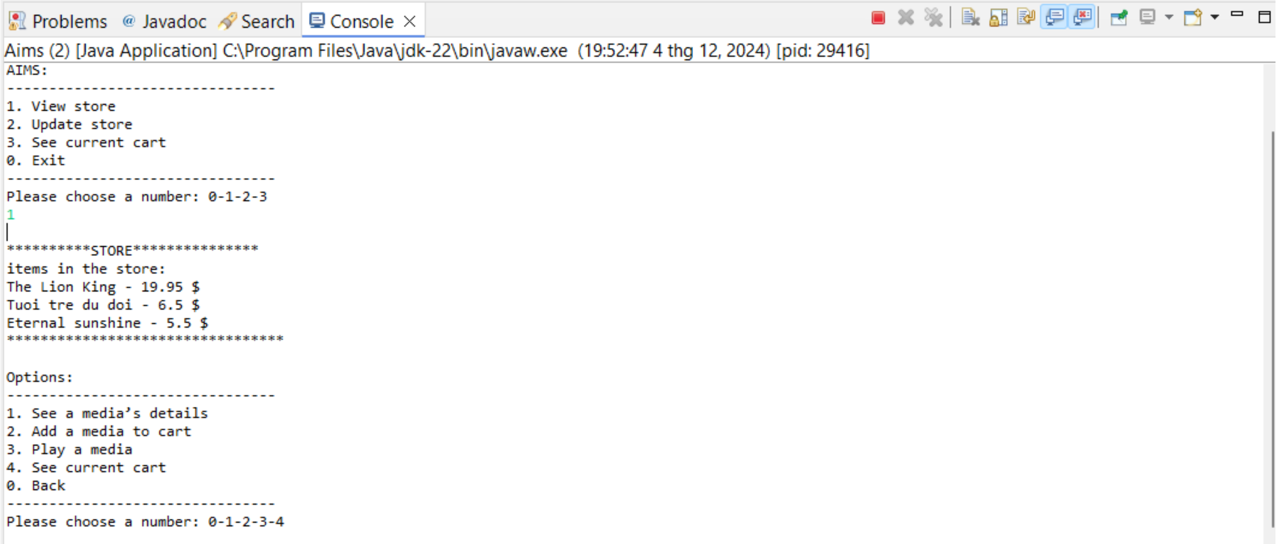


Aims (2) [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (19:51:56 4 thg 12, 2024) [pid: 29156]

```
The Lion King has been added to the store.
Tuoi tre du doi has been added to the store.
Eternal sunshine has been added to the store.
AIMS:
--------------------------------
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
```

Figure 11.1: Màn hình chính

11.1 Người dùng chọn 1: View store

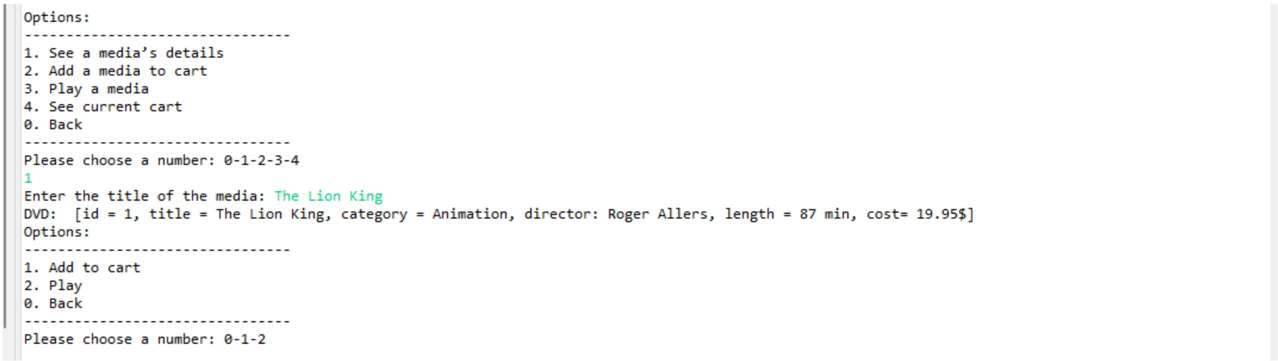


Aims (2) [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (19:52:47 4 thg 12, 2024) [pid: 29416]

```
AIMS:
--------------------------------
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1

**********STORE***************
items in the store:
The Lion King - 19.95 $
Tuoi tre du doi - 6.5 $
Eternal sunshine - 5.5 $
*********************************

Options:
--------------------------------
1. See a media's details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
```

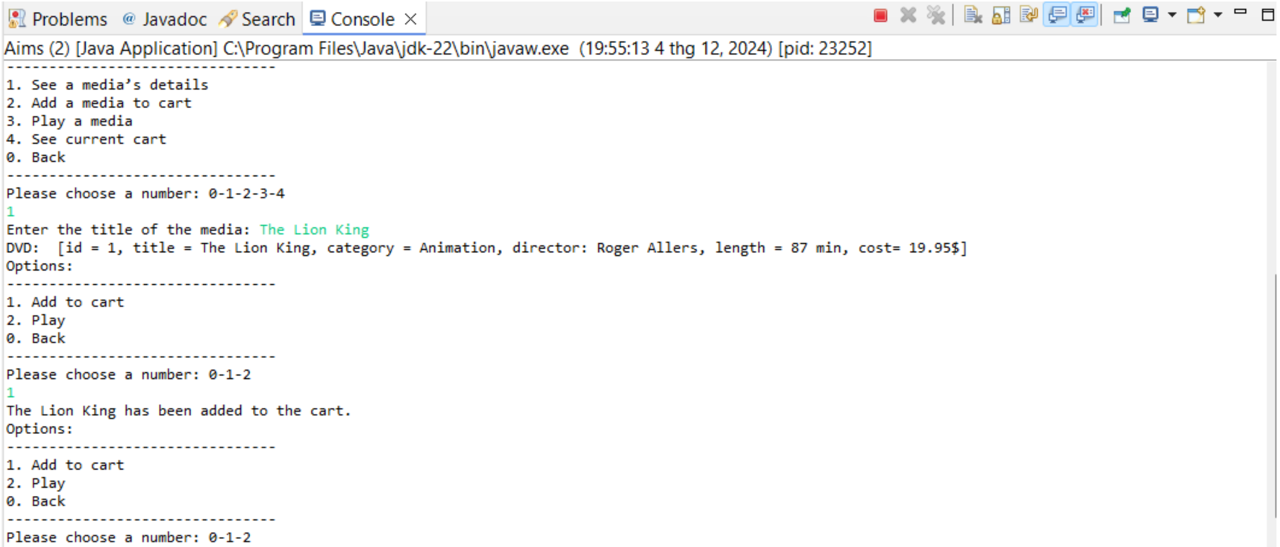
Figure 11.2: Vào Trang View Store

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



```
Options:
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
1
Enter the title of the media: The Lion King
DVD:  [id = 1, title = The Lion King, category = Animation, director: Roger Allers, length = 87 min, cost= 19.95$]
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
```

Figure 11.3: See a media's details

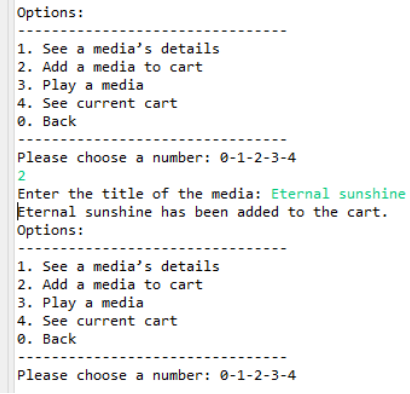


Aims (2) [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (19:55:13 4 thg 12, 2024) [pid: 23252]

```
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
1
Enter the title of the media: The Lion King
DVD:  [id = 1, title = The Lion King, category = Animation, director: Roger Allers, length = 87 min, cost= 19.95$]
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
1
The Lion King has been added to the cart.
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
```

Figure 11.4: Thêm vào Cart

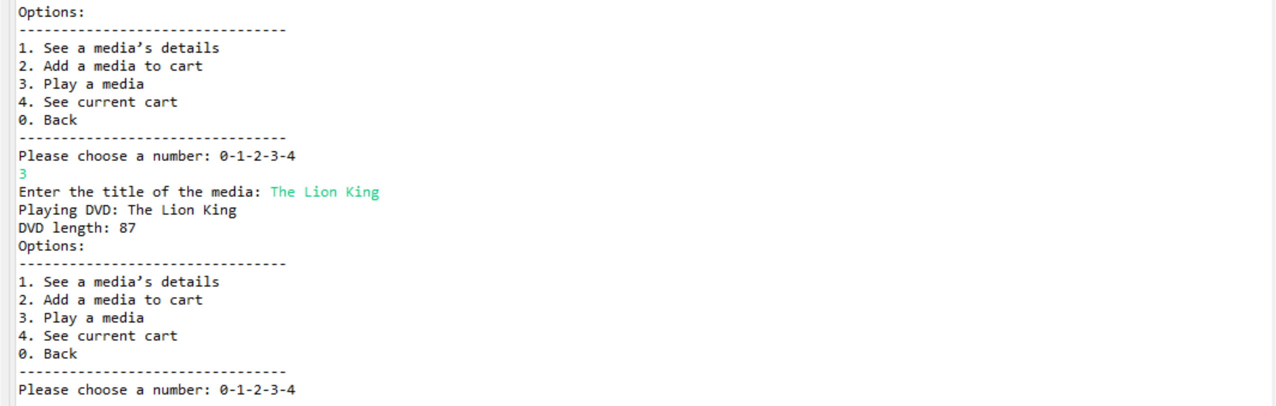
11.1.2 Người dùng chọn 2: Add a media to the cart



```
Options:
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
2
Enter the title of the media: Eternal sunshine
Eternal sunshine has been added to the cart.
Options:
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
```

Figure 11.5: Thêm media vào Cart

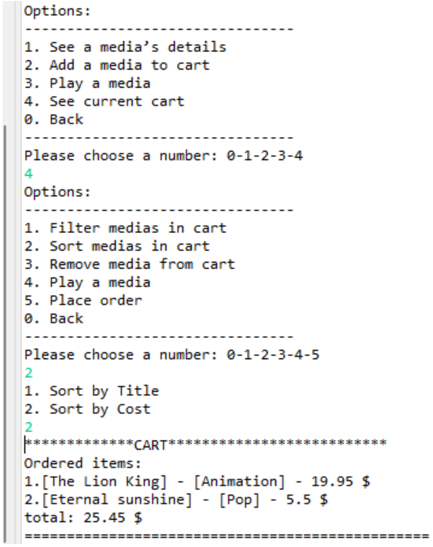
11.1.3 Người dùng chọn 3: Play a media



```
Options:
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
3
Enter the title of the media: The Lion King
Playing DVD: The Lion King
DVD length: 87
Options:
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
```

Figure 11.6: Play a media

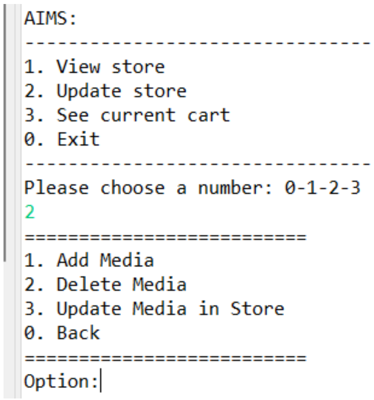
11.1.4 Người dùng chọn 4: See current cart



```
Options:
--------------------------------
1. See a media’s details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
4
Options:
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
2
1. Sort by Title
2. Sort by Cost
2
*************CART**************************
Ordered items:
1.[The Lion King] - [Animation] - 19.95 $
2.[Eternal sunshine] - [Pop] - 5.5 $
total: 25.45 $
================================================
```

Figure 11.7: See current cart after sort

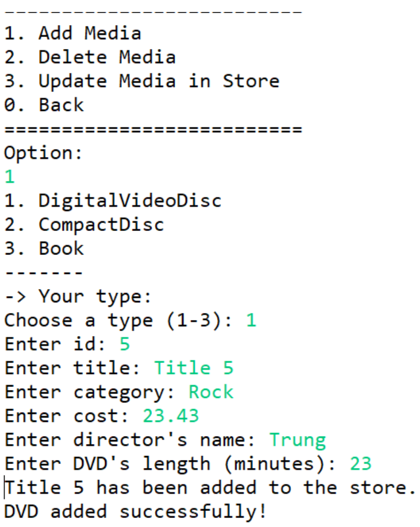
11.2 Người dùng chọn 2: Update store



```
AIMS:
--------------------------------
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
2
==========================
1. Add Media
2. Delete Media
3. Update Media in Store
0. Back
==========================
Option:
```

Figure 11.8: Vào Trang Update Store

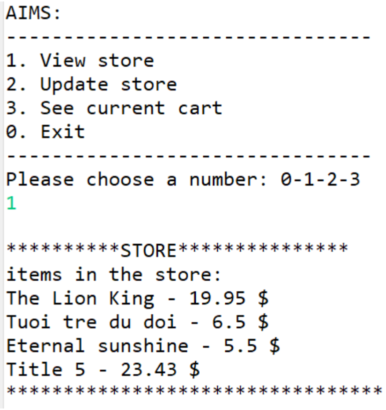
11.2.1 Người dùng chọn 1: Add a media to the store



```
--------------------------
1. Add Media
2. Delete Media
3. Update Media in Store
0. Back
==========================
Option:
1
1. DigitalVideoDisc
2. CompactDisc
3. Book
-------
-> Your type:
Choose a type (1-3): 1
Enter id: 5
Enter title: Title 5
Enter category: Rock
Enter cost: 23.43
Enter director's name: Trung
Enter DVD's length (minutes): 23
Title 5 has been added to the store.
DVD added successfully!
```

Figure 11.9: Add a media to store

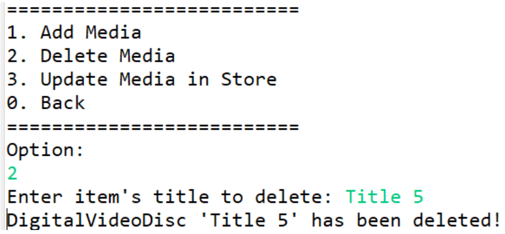
=> Kết quả sau khi thêm



```
AIMS:
--------------------------------
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1

*********STORE**************
items in the store:
The Lion King - 19.95 $
Tuoi tre du doi - 6.5 $
Eternal sunshine - 5.5 $
Title 5 - 23.43 $
*********************************
```

Figure 11.10: Result after add media to store

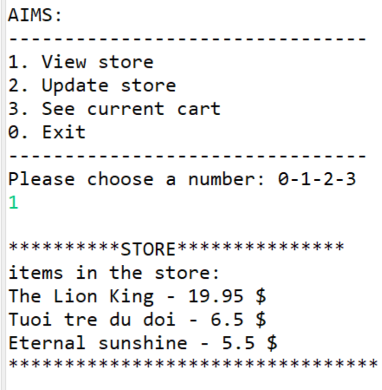
11.2.2 Người dùng chọn 2: Remove a media from the store



```
==========================
1. Add Media
2. Delete Media
3. Update Media in Store
0. Back
==========================
Option:
2
Enter item's title to delete: Title 5
DigitalVideoDisc 'Title 5' has been deleted!
```

Figure 11.11: Remove a media from the store

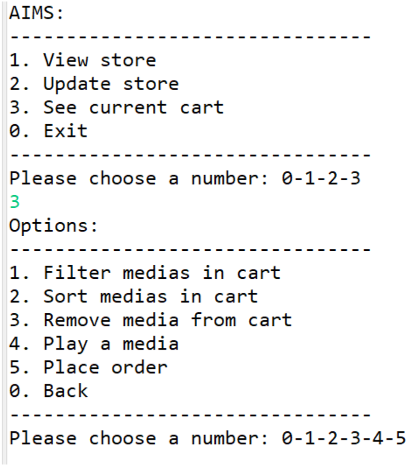
=> Kết quả sau khi remove



```
AIMS:
--------------------------------
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1

*********STORE**************
items in the store:
The Lion King - 19.95 $
Tuoi tre du doi - 6.5 $
Eternal sunshine - 5.5 $
*********************************
```

Figure 11.12: Result after remove a media

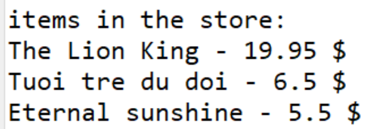
11.3 Người dùng chọn 3: See current cart



```
AIMS:
--------------------------------
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
3
Options:
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
```

Figure 11.13: Vào trang See current cart

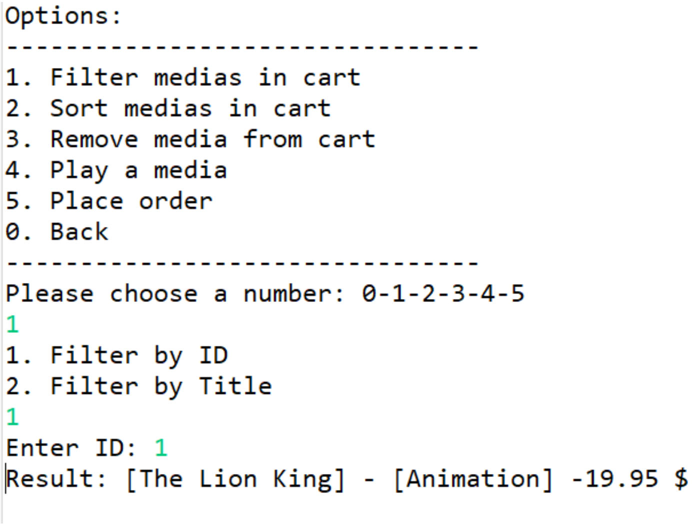
Giả sử lúc này trong Cart sẽ có các Media sau



```
items in the store:
The Lion King - 19.95 $
Tuoi tre du doi - 6.5 $
Eternal sunshine - 5.5 $
```

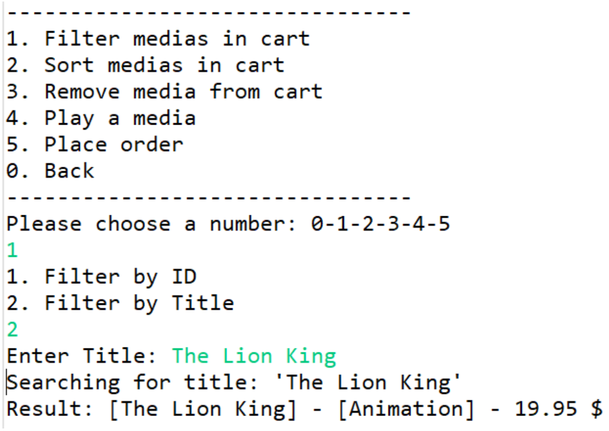
Figure 11.14: Media in Cart

11.3.1 Người dùng chọn 1: Filter medias in cart



```
Options:
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
1
1. Filter by ID
2. Filter by Title
1
Enter ID: 1
Result: [The Lion King] - [Animation] -19.95 $
```

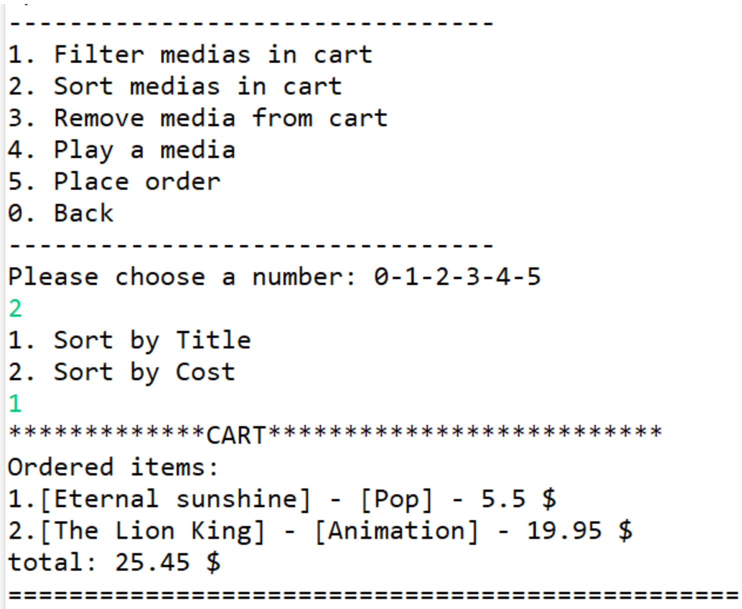
Figure 11.15: Filter Cart By id



```
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
1
1. Filter by ID
2. Filter by Title
2
Enter Title: The Lion King
Searching for title: 'The Lion King'
Result: [The Lion King] - [Animation] - 19.95 $
```

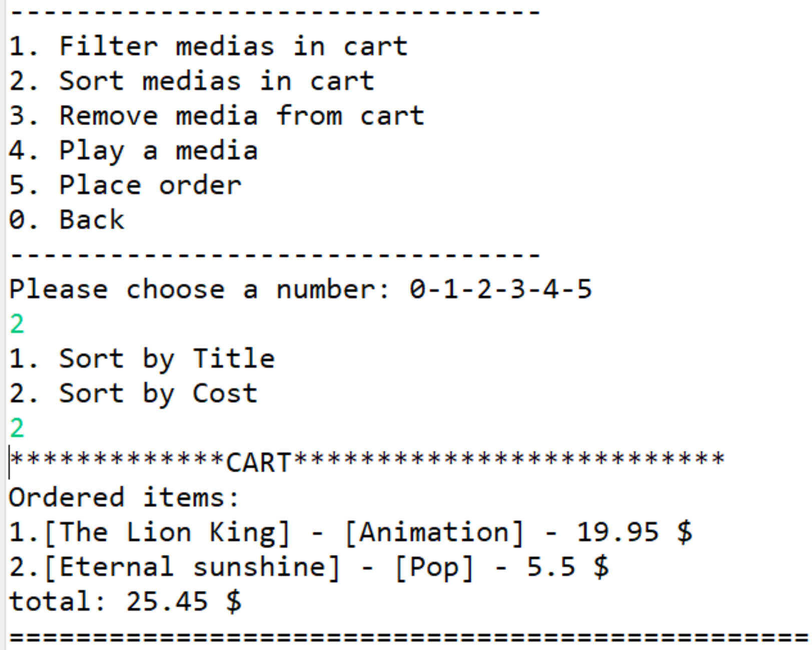
Figure 11.16: Filter Cart By Title

11.3.2 Người dùng chọn 2: Sort medias in cart



```
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
2
1. Sort by Title
2. Sort by Cost
1
*************CART**************************
Ordered items:
1.[Eternal sunshine] - [Pop] - 5.5 $
2.[The Lion King] - [Animation] - 19.95 $
total: 25.45 $
================================================
```

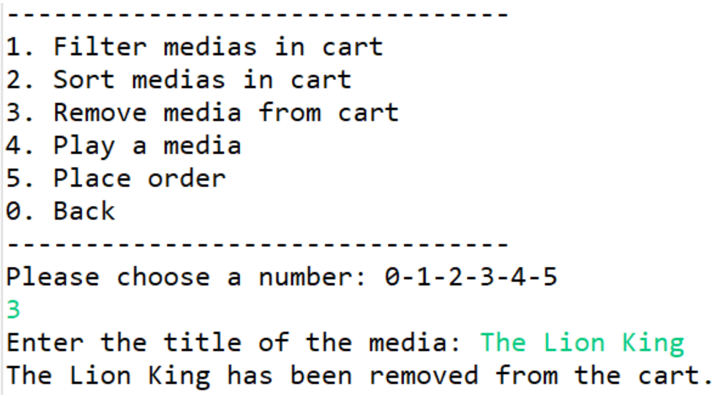
Figure 11.17: Sort Cart By Title



```
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
2
1. Sort by Title
2. Sort by Cost
2
*************CART**************************
Ordered items:
1.[The Lion King] - [Animation] - 19.95 $
2.[Eternal sunshine] - [Pop] - 5.5 $
total: 25.45 $
================================================
```

Figure 11.18: Sort Cart By Cost

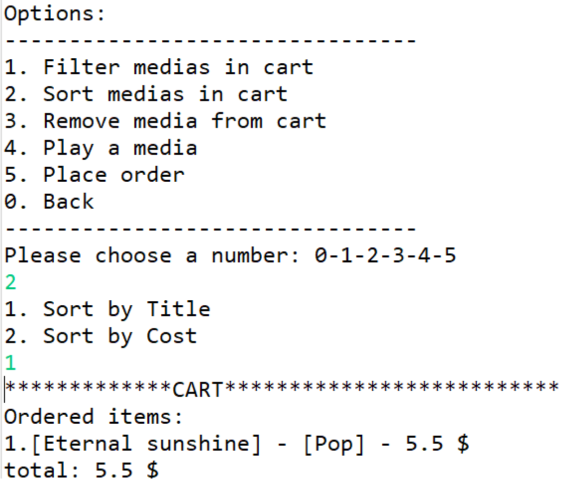
11.3.3 Người dùng chọn 3: Remove media from cart



```
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
3
Enter the title of the media: The Lion King
The Lion King has been removed from the cart.
```

Figure 11.19: Remove media by title

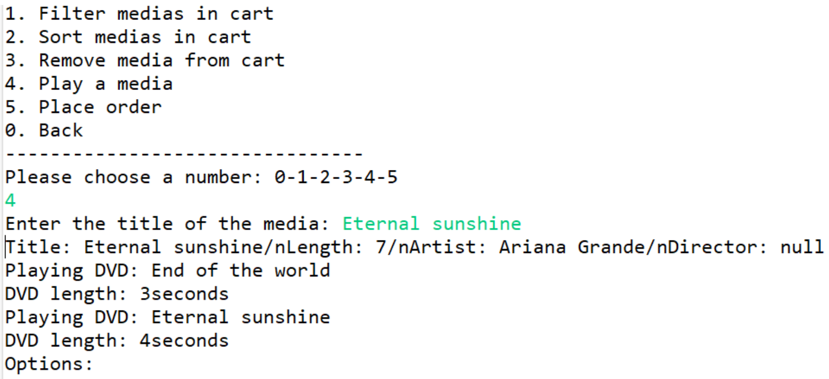
=> Kết quả



```
Options:
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
2
1. Sort by Title
2. Sort by Cost
1
*************CART**************************
Ordered items:
1.[Eternal sunshine] - [Pop] - 5.5 $
total: 5.5 $
```

Figure 11.20: Result after remove media in cart by id

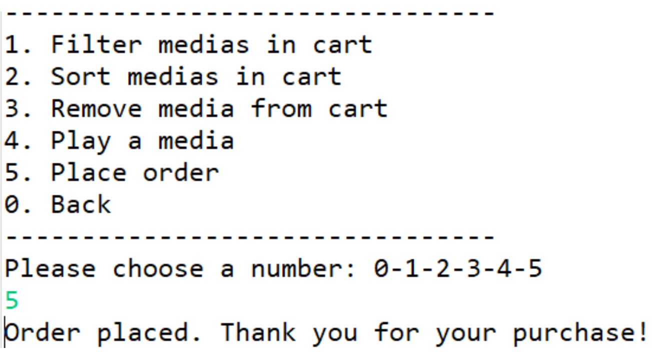
11.3.4 Người dùng chọn 4: Play a media



```
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
4
Enter the title of the media: Eternal sunshine
Title: Eternal sunshine/nLength: 7/nArtist: Ariana Grande/nDirector: null
Playing DVD: End of the world
DVD length: 3seconds
Playing DVD: Eternal sunshine
DVD length: 4seconds
Options:
```

Figure 11.21: Play a media in cart

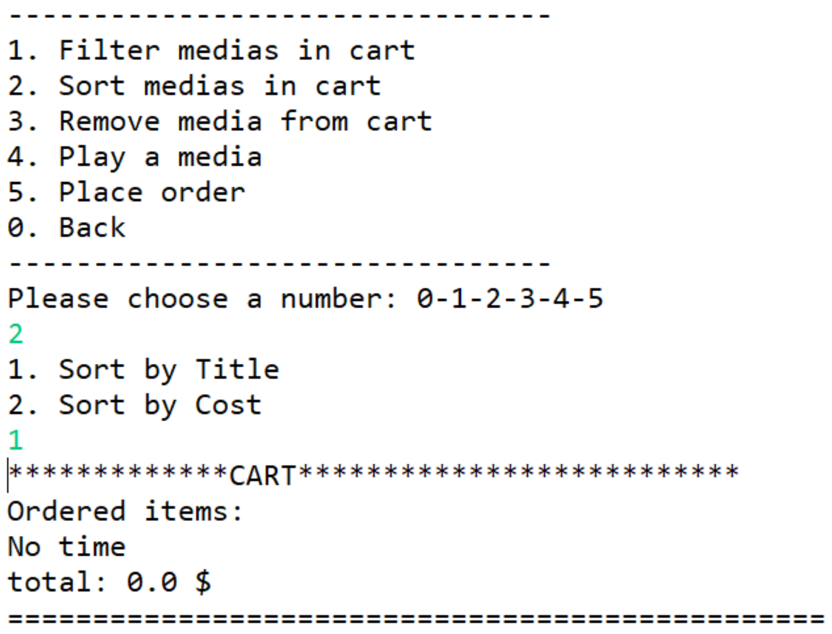
11.3.5 Người dùng chọn 5: Place order



```
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
5
Order placed. Thank you for your purchase!
```

Figure 11.22: Order

=> Kết quả sau khi order



```
--------------------------------
1. Filter medias in cart
2. Sort medias in cart
3. Remove media from cart
4. Play a media
5. Place order
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4-5
2
1. Sort by Title
2. Sort by Cost
1
*************CART**************************
Ordered items:
No time
total: 0.0 $
================================================
```

Figure 11.23: Result after order

12 Class Diagram

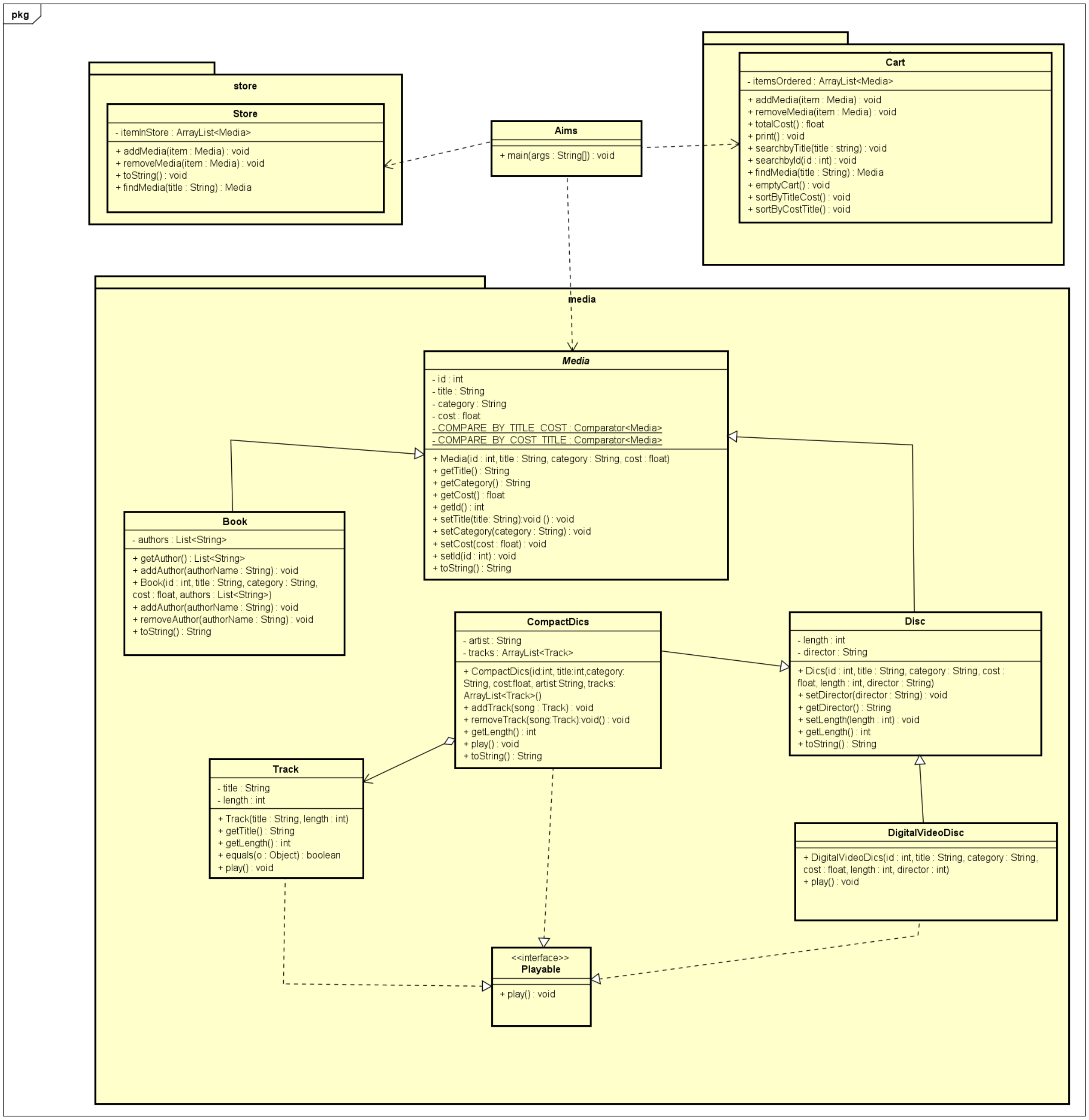
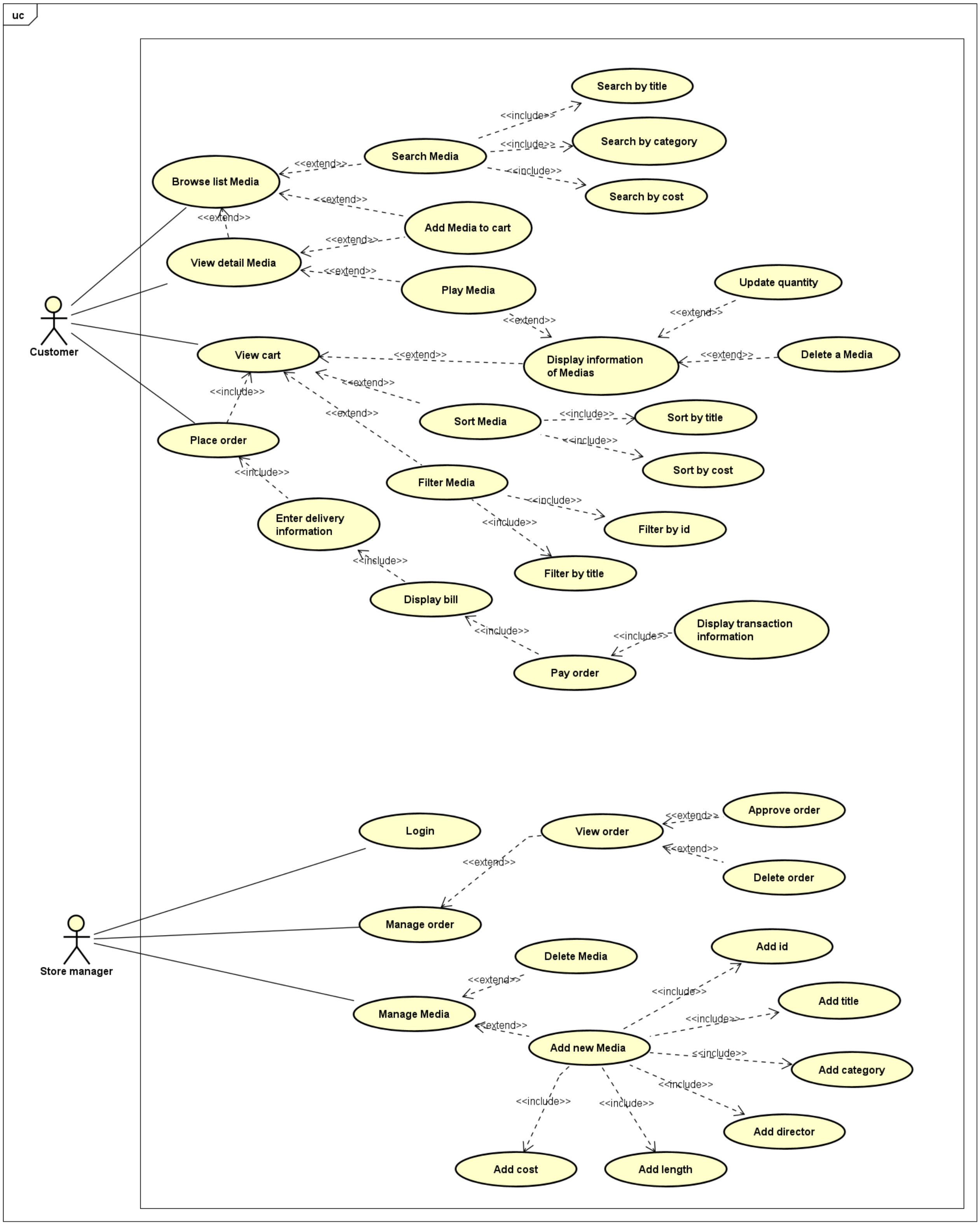

pkg
store
Store
- itemInStore : ArrayList<Media>
+ addMedia(item : Media) : void
+ removeMedia(item : Media) : void
+ toString() : void
+ findMedia(title : String) : Media
Aims
+ main(args : String[]) : void
Cart
- itemsOrdered : ArrayList<Media>
+ addMedia(item : Media) : void
+ removeMedia(item : Media) : void
+ totalCost() : float
+ print() : void
+ searchbyTitle(title : string) : void
+ searchbyId(id : int) : void
+ findMedia(title : String) : Media
+ emptyCart() : void
+ sortByTitleCost() : void
+ sortByCostTitle() : void
media
Media
- id : int
- title : String
- category : String
- cost : float
- COMPARE_BY_TITLE_COST : Comparator<Media>
- COMPARE_BY_COST_TITLE : Comparator<Media>
+ Media(id : int, title : String, category : String, cost : float)
+ getTitle() : String
+ getCategory() : String
+ getCost() : float
+ getId() : int
+ setTitle(title: String):void () : void
+ setCategory(category : String) : void
+ setCost(cost : float) : void
+ setId(id : int) : void
+ toString() : String
Book
- authors : List<String>
+ getAuthor() : List<String>
+ addAuthor(authorName : String) : void
+ Book(id : int, title : String, category : String, cost : float, authors : List<String>)
+ addAuthor(authorName : String) : void
+ removeAuthor(authorName : String) : void
+ toString() : String
CompactDics
- artist : String
- tracks : ArrayList<Track>
+ CompactDics(id:int, title:int,category: String, cost:float, artist:String, tracks: ArrayList<Track>()
+ addTrack(song : Track) : void
+ removeTrack(song:Track):void() : void
+ getLength() : int
+ play() : void
+ toString() : String
Disc
- length : int
- director : String
+ Dics(id : int, title : String, category : String, cost : float, length : int, director : String)
+ setDirector(director : String) : void
+ getDirector() : String
+ setLength(length : int) : void
+ getLength() : int
+ toString() : String
Track
- title : String
- length : int
+ Track(title : String, length : int)
+ getTitle() : String
+ getLength() : int
+ equals(o : Object) : boolean
+ play() : void
DigitalVideoDisc
+ DigitalVideoDics(id : int, title : String, category : String, cost : float, length : int, director : int)
+ play() : void
<<interface>>
Playable
+ play() : void

Figure 12.1: Class Diagram

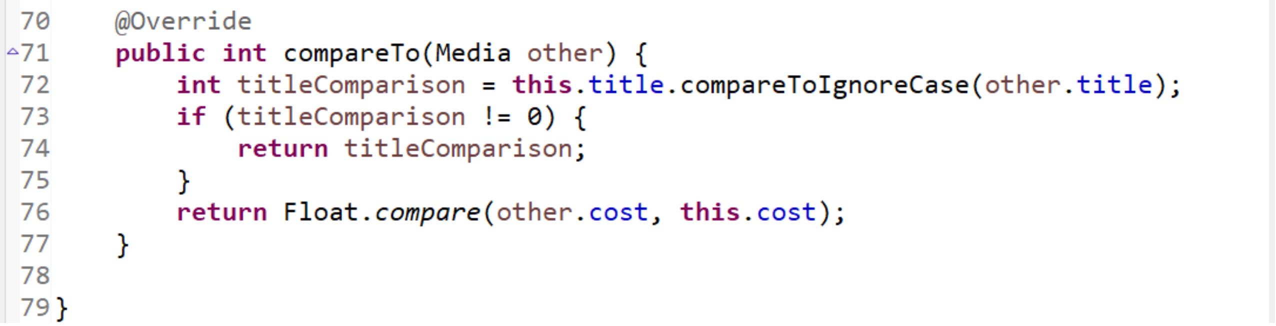
13 UseCase Diagram



Answer Questions

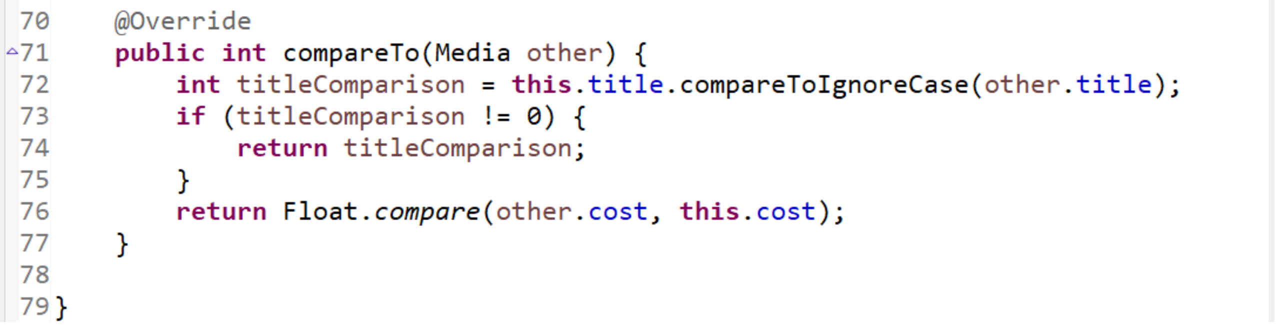
Figure 13.1: UseCase Diagram

Trong trường hợp muốn so sánh các đối tượng Media với nhau bằng cách sử dụng Comparable thay vì Comparator, thì thay vì tạo ra các lớp riêng cho từng Comparator, chúng ta cần để lớp Media triển khai interface Comparable.



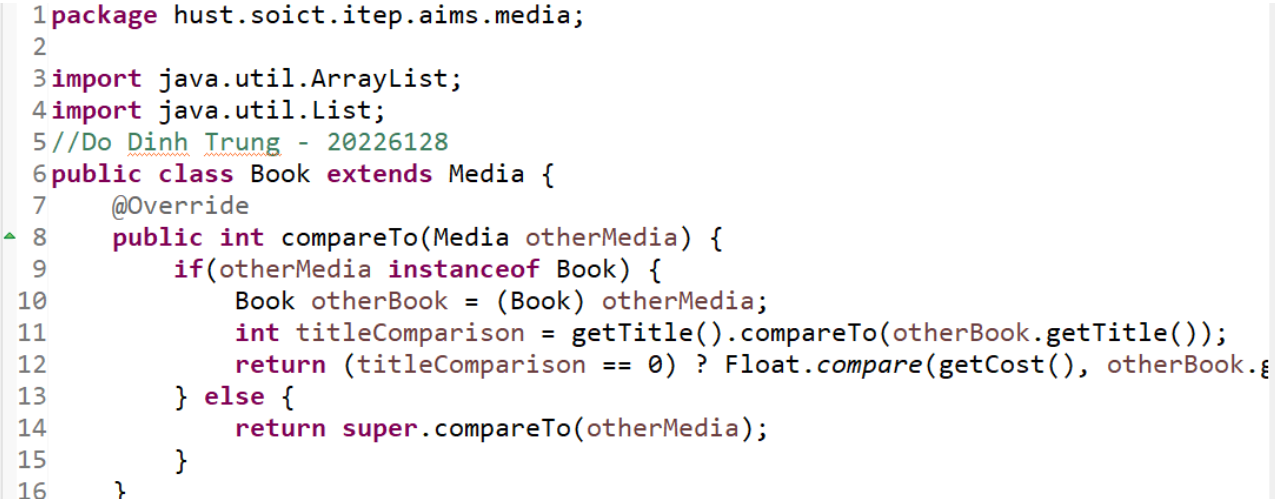
```
    @Override
    public int compareTo(Media other) {
        int titleComparison = this.title.compareToIgnoreCase(other.title);
        if (titleComparison != 0) {
            return titleComparison;
        }
        return Float.compare(other.cost, this.cost);
    }

}
```

Figure 14.1: Triển khai Comparable trong lớp trừu tượng Media



```
    @Override
    public int compareTo(Media other) {
        int titleComparison = this.title.compareToIgnoreCase(other.title);
        if (titleComparison != 0) {
            return titleComparison;
        }
        return Float.compare(other.cost, this.cost);
    }

}
```

Figure 14.2: Mở rộng để so sánh nhiều thuộc tính hơn



```
package hust.soict.itep.aims.media;

import java.util.ArrayList;
import java.util.List;
//Do Dinh Trung - 20226128
public class Book extends Media {
    @Override
    public int compareTo(Media otherMedia) {
        if(otherMedia instanceof Book) {
            Book otherBook = (Book) otherMedia;
            int titleComparison = getTitle().compareTo(otherBook.getTitle());
            return (titleComparison == 0) ? Float.compare(getCost(), otherBook.g
        } else {
            return super.compareTo(otherMedia);
        }
    }
```

Figure 14.3: Triển khai tại lớp con

Cách triển khai này giúp chúng ta linh hoạt hơn khi so sánh các đối tượng Media và cung cấp khả năng mở rộng cho các lớp con khác nếu cần thiết.