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 – LAB 4

Họ và tên SV: Đặng Hồng Minh
MSSV: 20225740
Lớp: CNTT Việt Nhật – K67
GVHD: Lê Thị Hoa
HTGD: Đặng Mạnh Cường

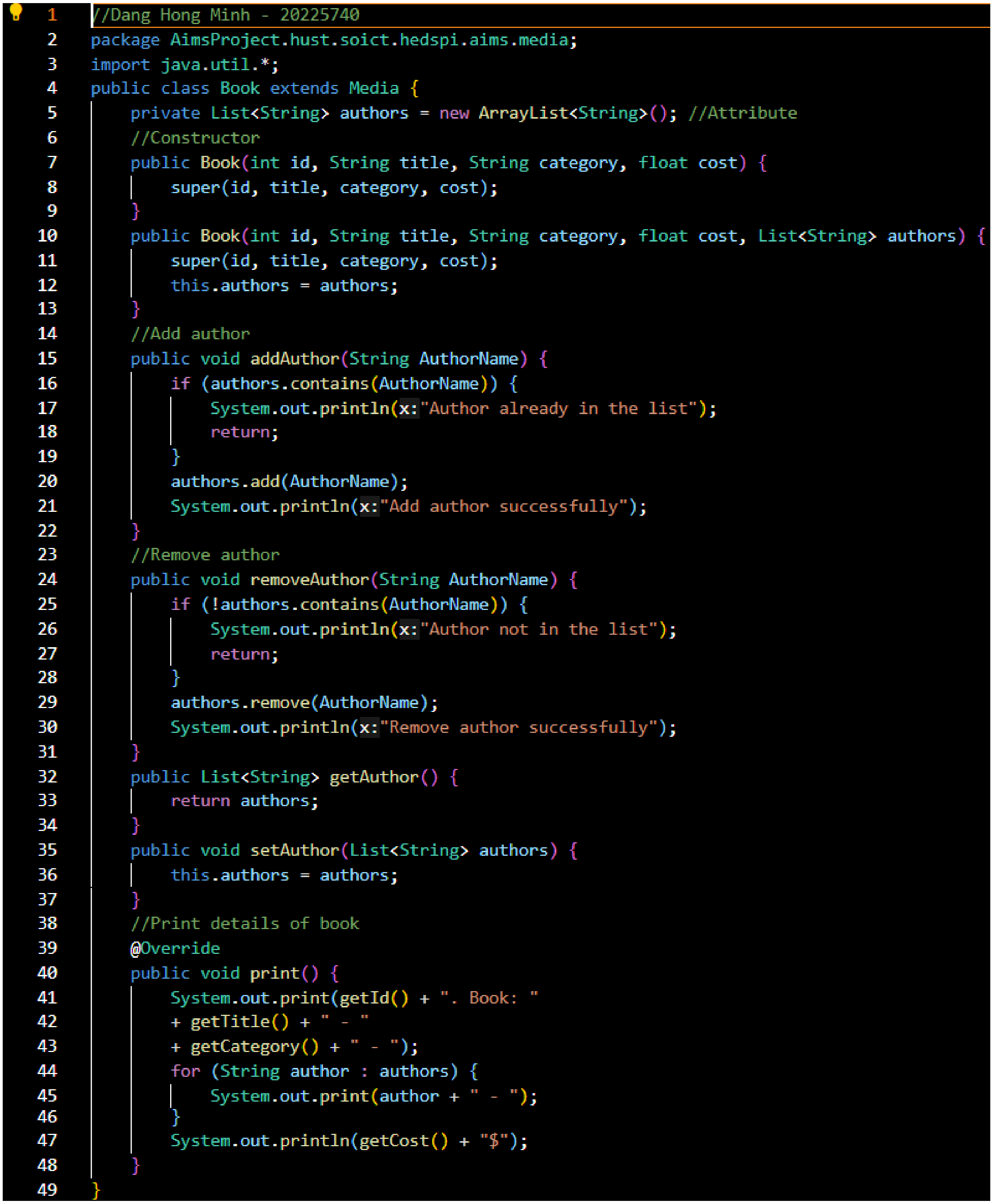
Hà Nội 11/2024

Mục lục

Danh mục hình ảnh

BÁO CÁO THỰC HÀNH LAB 4

1. Create the Book class

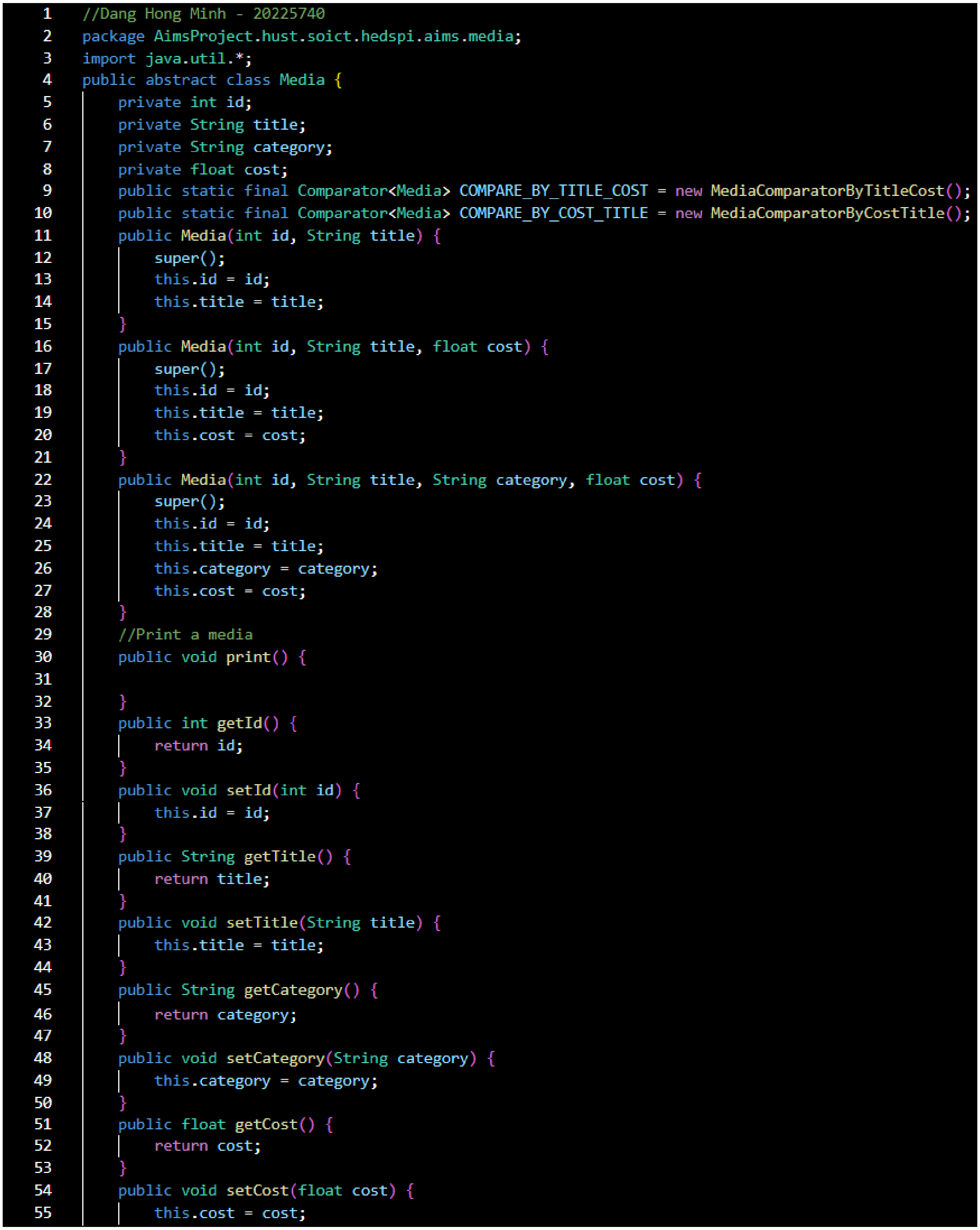


```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
import java.util.*;
public class Book extends Media {
    private List<String> authors = new ArrayList<String>(); //Attribute
    //Constructor
    public Book(int id, String title, String category, float cost) {
        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;
    }
    //Add author
    public void addAuthor(String AuthorName) {
        if (authors.contains(AuthorName)) {
            System.out.println(x:"Author already in the list");
            return;
        }
        authors.add(AuthorName);
        System.out.println(x:"Add author successfully");
    }
    //Remove author
    public void removeAuthor(String AuthorName) {
        if (!authors.contains(AuthorName)) {
            System.out.println(x:"Author not in the list");
            return;
        }
        authors.remove(AuthorName);
        System.out.println(x:"Remove author successfully");
    }
    public List<String> getAuthor() {
        return authors;
    }
    public void setAuthor(List<String> authors) {
        this.authors = authors;
    }
    //Print details of book
    @Override
    public void print() {
        System.out.print(getId() + ". Book: "
        + getTitle() + " - "
        + getCategory() + " - ");
        for (String author : authors) {
            System.out.print(author + " - ");
        }
        System.out.println(getCost() + "$");
    }
}
```

Figure 1: Book Class

2. Creating the abstract Media class

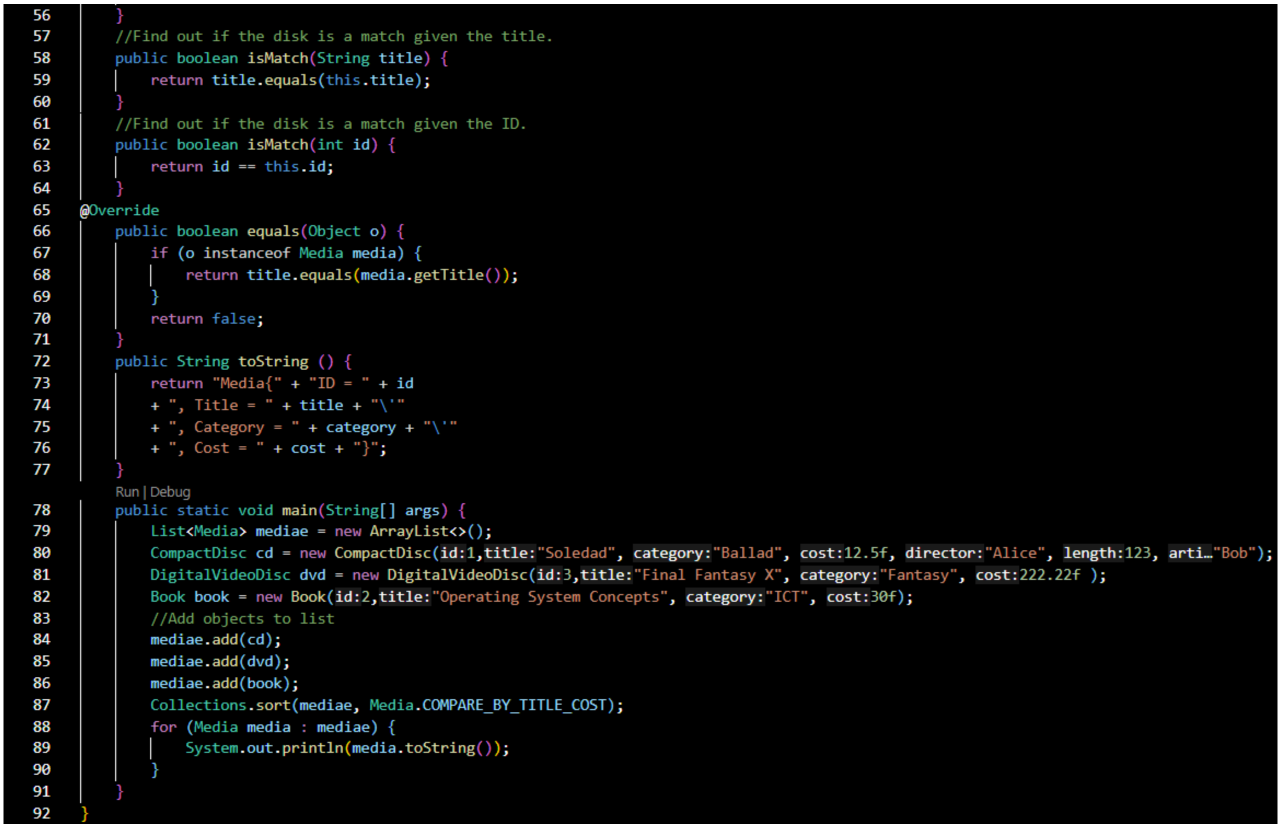
- Đây sẽ là lớp cha để các lớp DigitalVideoDisc, Book kế thừa.



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
import java.util.*;
public abstract class 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) {
        super();
        this.id = id;
        this.title = title;
    }
    public Media(int id, String title, float cost) {
        super();
        this.id = id;
        this.title = title;
        this.cost = cost;
    }
    public Media(int id, String title, String category, float cost) {
        super();
        this.id = id;
        this.title = title;
        this.category = category;
        this.cost = cost;
    }
    //Print a media
    public void print() {

    }
    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;
    }
    public float getCost() {
        return cost;
    }
    public void setCost(float cost) {
        this.cost = cost;
```

Figure 2: Media Class 1

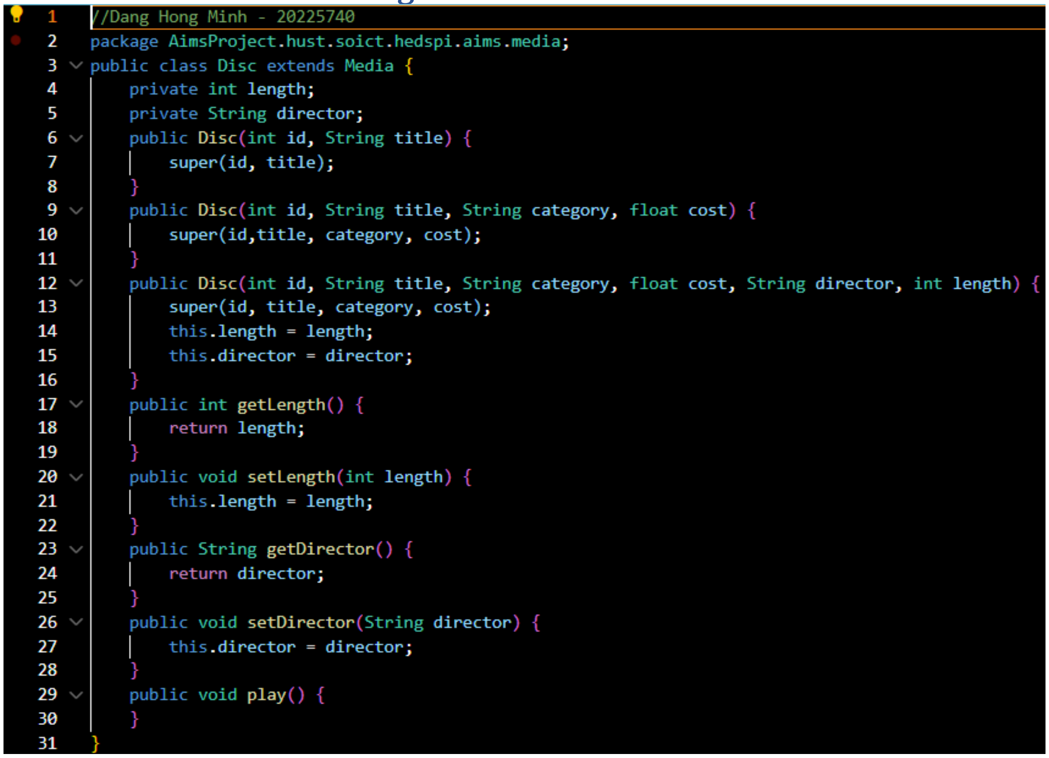


```
    }
    //Find out if the disk is a match given the title.
    public boolean isMatch(String title) {
        return title.equals(this.title);
    }
    //Find out if the disk is a match given the ID.
    public boolean isMatch(int id) {
        return id == this.id;
    }
@Override
    public boolean equals(Object o) {
        if (o instanceof Media media) {
            return title.equals(media.getTitle());
        }
        return false;
    }
    public String toString () {
        return "Media{" + "ID = " + id
        + ", Title = " + title + "\'"
        + ", Category = " + category + "\'"
        + ", Cost = " + cost + "}";
    }
    Run | Debug
    public static void main(String[] args) {
        List<Media> mediae = new ArrayList<>();
        CompactDisc cd = new CompactDisc(id:1,title:"Soledad", category:"Ballad", cost:12.5f, director:"Alice", length:123, arti…"Bob");
        DigitalVideoDisc dvd = new DigitalVideoDisc(id:3,title:"Final Fantasy X", category:"Fantasy", cost:222.22f );
        Book book = new Book(id:2,title:"Operating System Concepts", category:"ICT", cost:30f);
        //Add objects to list
        mediae.add(cd);
        mediae.add(dvd);
        mediae.add(book);
        Collections.sort(mediae, Media.COMPARE_BY_TITLE_COST);
        for (Media media : mediae) {
            System.out.println(media.toString());
        }
    }
}
```

Figure 3: Media Class 2

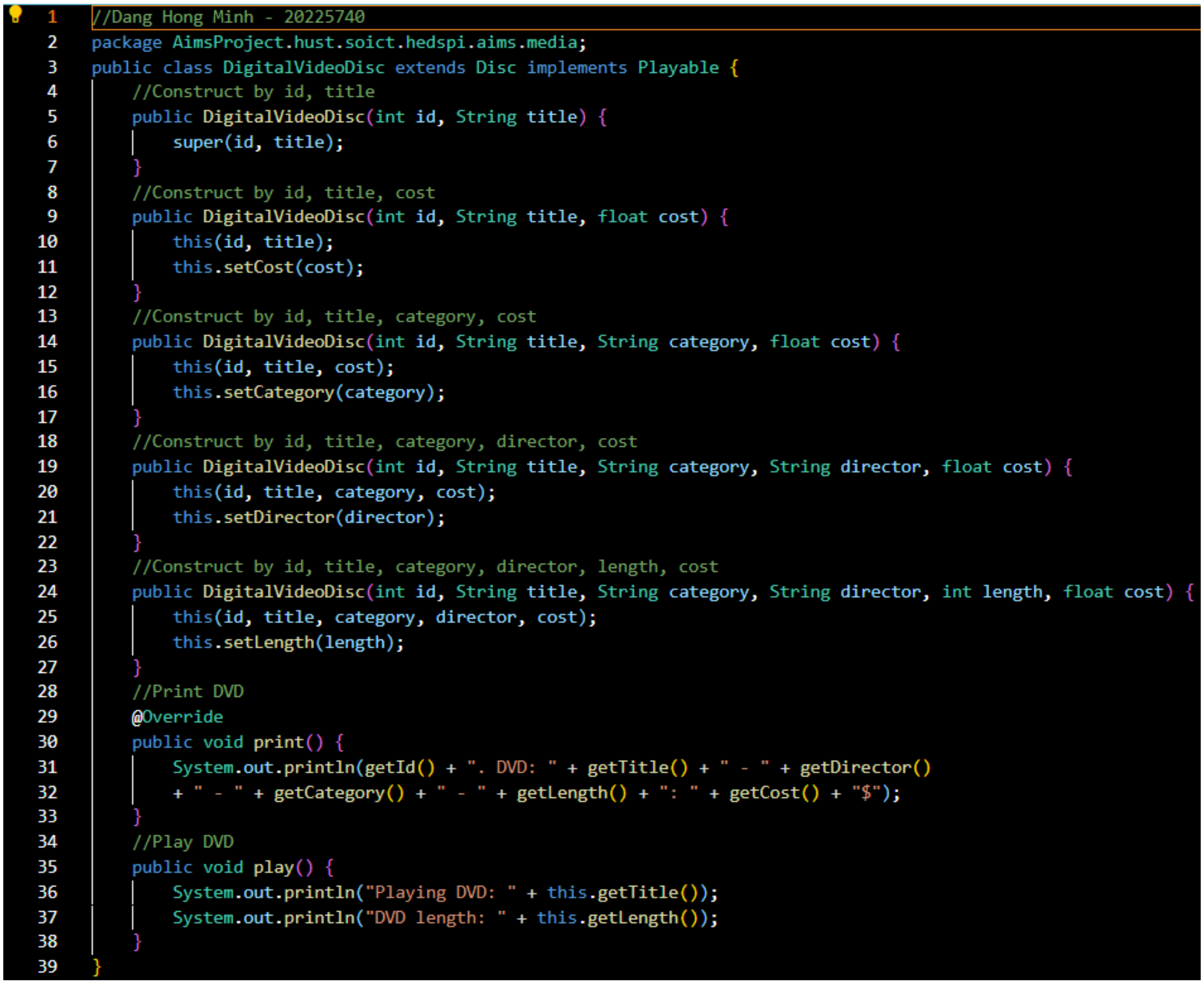
3. Creating the CompactDisc class

3.1 Create the Disc class extending the Media class



```
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
public class Disc extends Media {
    private int length;
    private String director;
    public Disc(int id, String title) {
        super(id, title);
    }
    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, String director, int length) {
        super(id, title, category, cost);
        this.length = length;
        this.director = 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 void play() {
    }
}
```

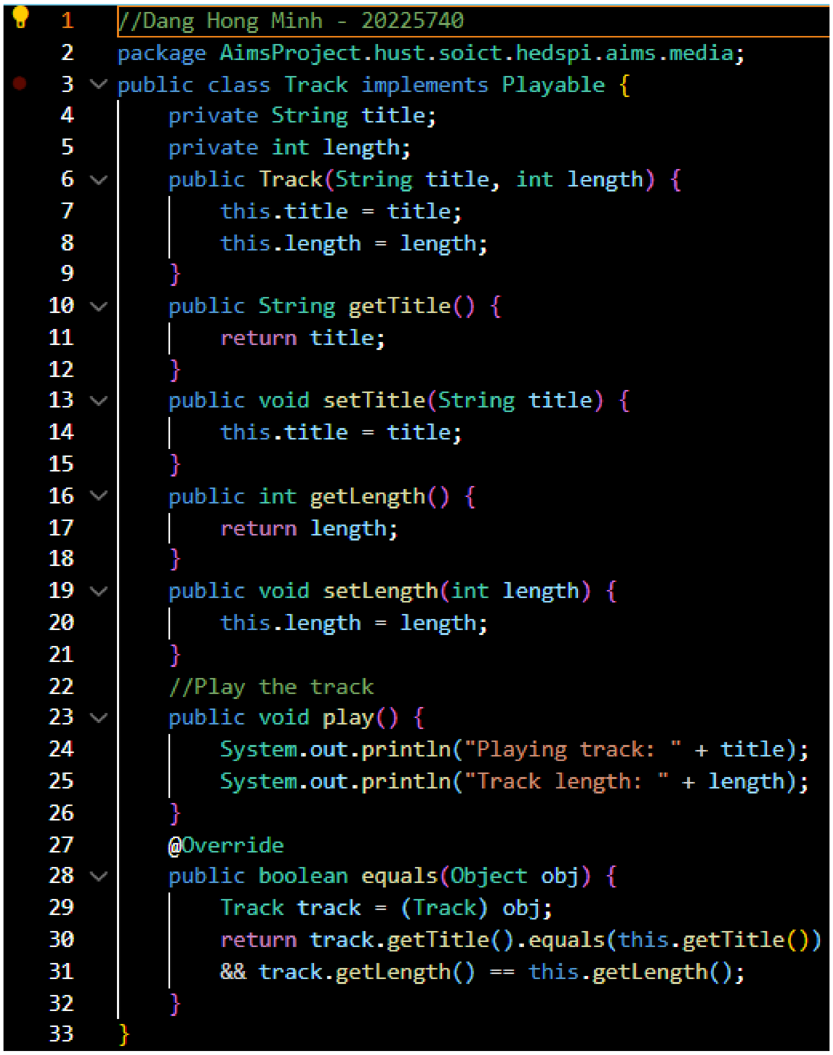
Figure 4: Disc Class



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
public class DigitalVideoDisc extends Disc implements Playable {
    //Construct by id, title
    public DigitalVideoDisc(int id, String title) {
        super(id, title);
    }
    //Construct by id, title, cost
    public DigitalVideoDisc(int id, String title, float cost) {
        this(id, title);
        this.setCost(cost);
    }
    //Construct by id, title, category, cost
    public DigitalVideoDisc(int id, String title, String category, float cost) {
        this(id, title, cost);
        this.setCategory(category);
    }
    //Construct by id, title, category, director, cost
    public DigitalVideoDisc(int id, String title, String category, String director, float cost) {
        this(id, title, category, cost);
        this.setDirector(director);
    }
    //Construct by id, title, category, director, length, cost
    public DigitalVideoDisc(int id, String title, String category, String director, int length, float cost) {
        this(id, title, category, director, cost);
        this.setLength(length);
    }
    //Print DVD
    @Override
    public void print() {
        System.out.println(getId() + ". DVD: " + getTitle() + " - " + getDirector()
        + " - " + getCategory() + " - " + getLength() + ": " + getCost() + "$");
    }
    //Play DVD
    public void play() {
        System.out.println("Playing DVD: " + this.getTitle());
        System.out.println("DVD length: " + this.getLength());
    }
}
```

Figure 5: DigitalVideoDisc Class

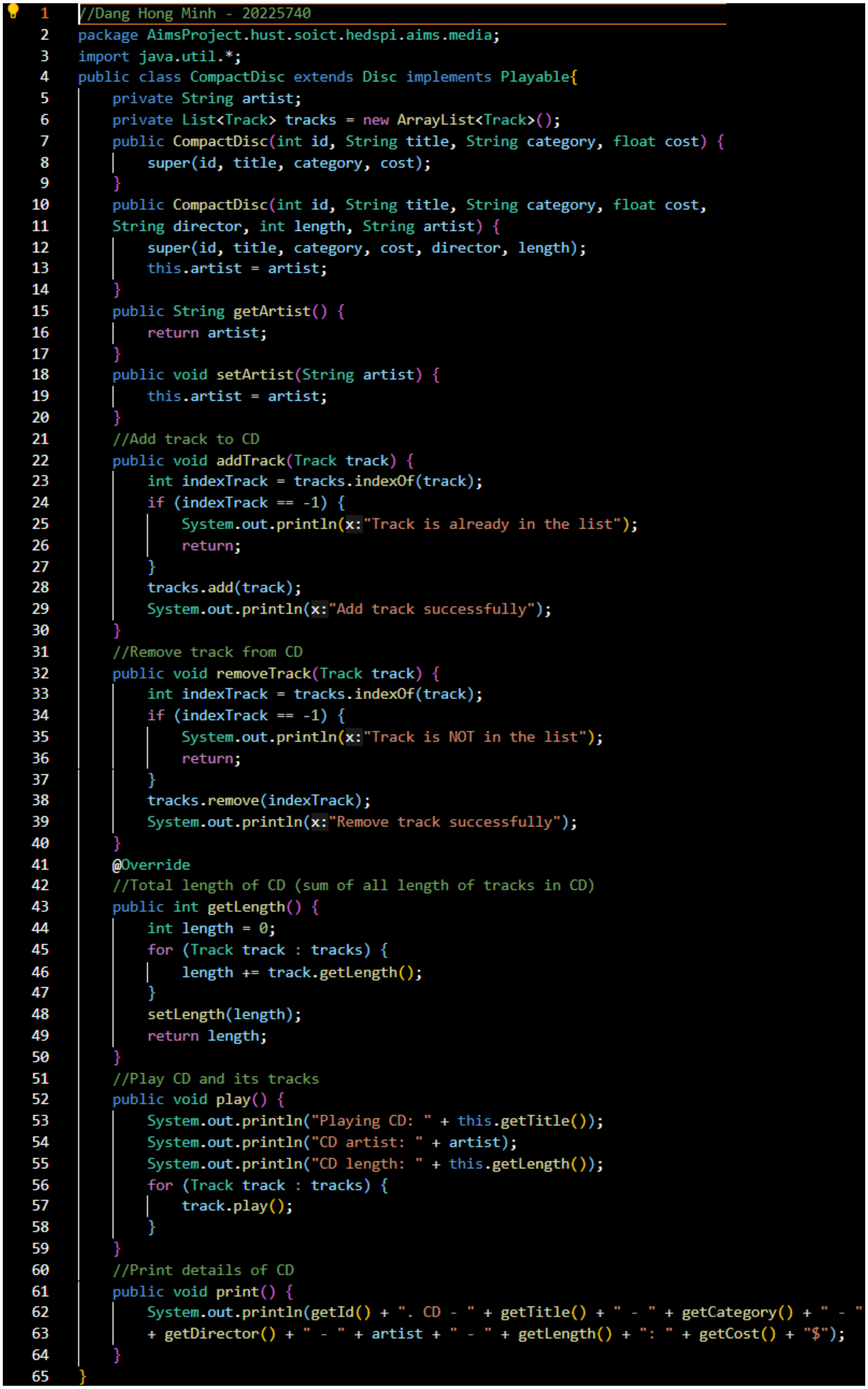
3.2 Create the Track class which models a track on a compact disc and will store information incuding the title and length of the track.



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
public class Track implements Playable {
    private String title;
    private int length;
    public Track(String title, int length) {
        this.title = title;
        this.length = 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;
    }
    //Play the track
    public void play() {
        System.out.println("Playing track: " + title);
        System.out.println("Track length: " + length);
    }
    @Override
    public boolean equals(Object obj) {
        Track track = (Track) obj;
        return track.getTitle().equals(this.getTitle())
        && track.getLength() == this.getLength();
    }
}
```

Figure 6: Track Class

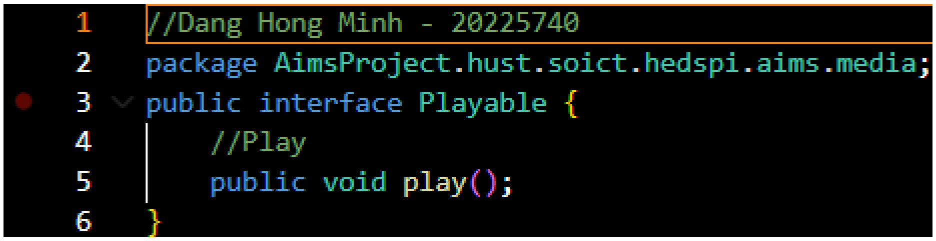
3.3 Open the CompactDisc class



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
import java.util.*;
public class CompactDisc extends Disc implements Playable{
    private String artist;
    private List<Track> tracks = new ArrayList<Track>();
    public CompactDisc(int id, String title, String category, float cost) {
        super(id, title, category, cost);
    }
    public CompactDisc(int id, String title, String category, float cost,
    String director, int length, String artist) {
        super(id, title, category, cost, director, length);
        this.artist = artist;
    }
    public String getArtist() {
        return artist;
    }
    public void setArtist(String artist) {
        this.artist = artist;
    }
    //Add track to CD
    public void addTrack(Track track) {
        int indexTrack = tracks.indexOf(track);
        if (indexTrack == -1) {
            System.out.println(x:"Track is already in the list");
            return;
        }
        tracks.add(track);
        System.out.println(x:"Add track successfully");
    }
    //Remove track from CD
    public void removeTrack(Track track) {
        int indexTrack = tracks.indexOf(track);
        if (indexTrack == -1) {
            System.out.println(x:"Track is NOT in the list");
            return;
        }
        tracks.remove(indexTrack);
        System.out.println(x:"Remove track successfully");
    }
    @Override
    //Total length of CD (sum of all length of tracks in CD)
    public int getLength() {
        int length = 0;
        for (Track track : tracks) {
            length += track.getLength();
        }
        setLength(length);
        return length;
    }
    //Play CD and its tracks
    public void play() {
        System.out.println("Playing CD: " + this.getTitle());
        System.out.println("CD artist: " + artist);
        System.out.println("CD length: " + this.getLength());
        for (Track track : tracks) {
            track.play();
        }
    }
    //Print details of CD
    public void print() {
        System.out.println(getId() + ". CD - " + getTitle() + " - " + getCategory() + " - "
        + getDirector() + " - " + artist + " - " + getLength() + ": " + getCost() + "$");
    }
}
```

Figure 7: CompactDisc Class

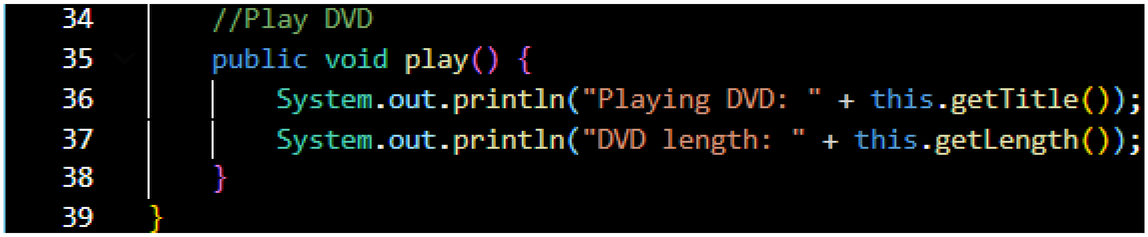
4. Create the Playable interface



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
public interface Playable {
    //Play
    public void play();
}
```

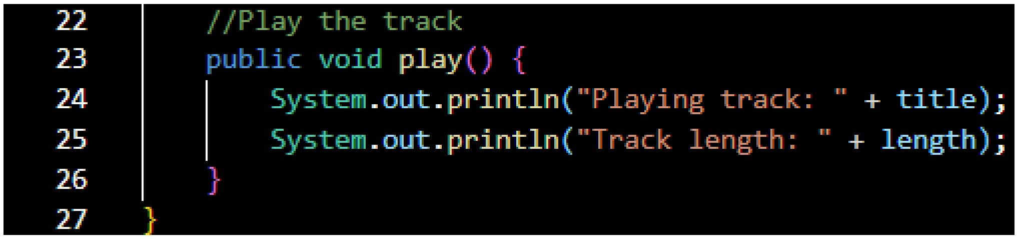
Figure 8: Playable interface

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



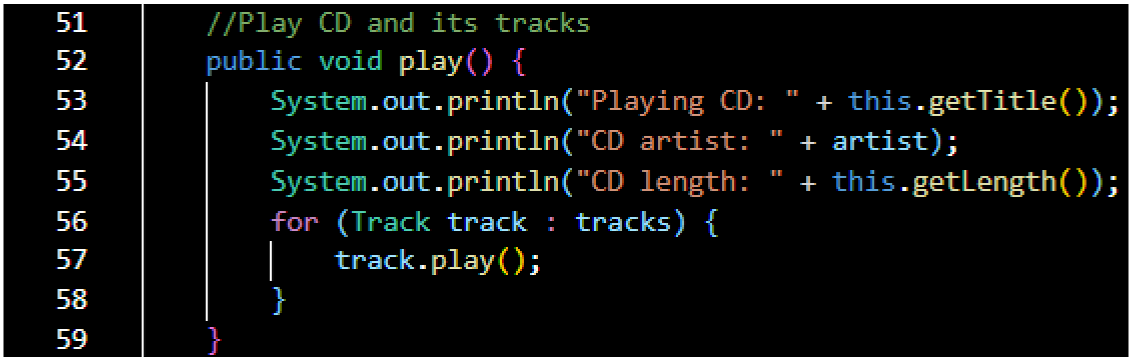
```java
    //Play DVD
    public void play() {
        System.out.println("Playing DVD: " + this.getTitle());
        System.out.println("DVD length: " + this.getLength());
    }
}
```

Figure 9: Method play() của DigitalVideoDisc



```java
    //Play the track
    public void play() {
        System.out.println("Playing track: " + title);
        System.out.println("Track length: " + length);
    }
}
```

Figure 10: Method play() của Track

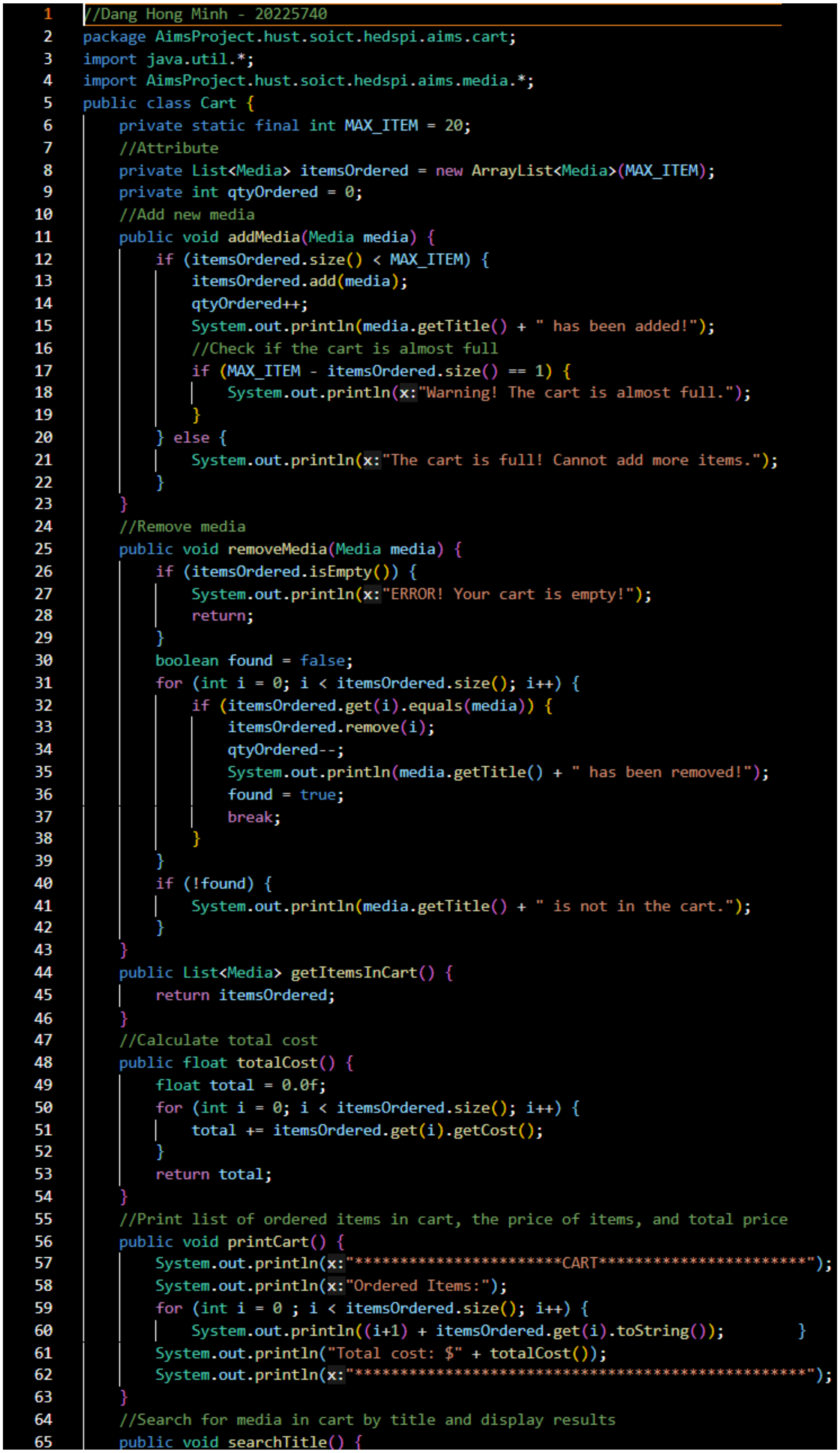


```java
    //Play CD and its tracks
    public void play() {
        System.out.println("Playing CD: " + this.getTitle());
        System.out.println("CD artist: " + artist);
        System.out.println("CD length: " + this.getLength());
        for (Track track : tracks) {
            track.play();
        }
    }
```

Figure 11: Method play() của CompactDisc

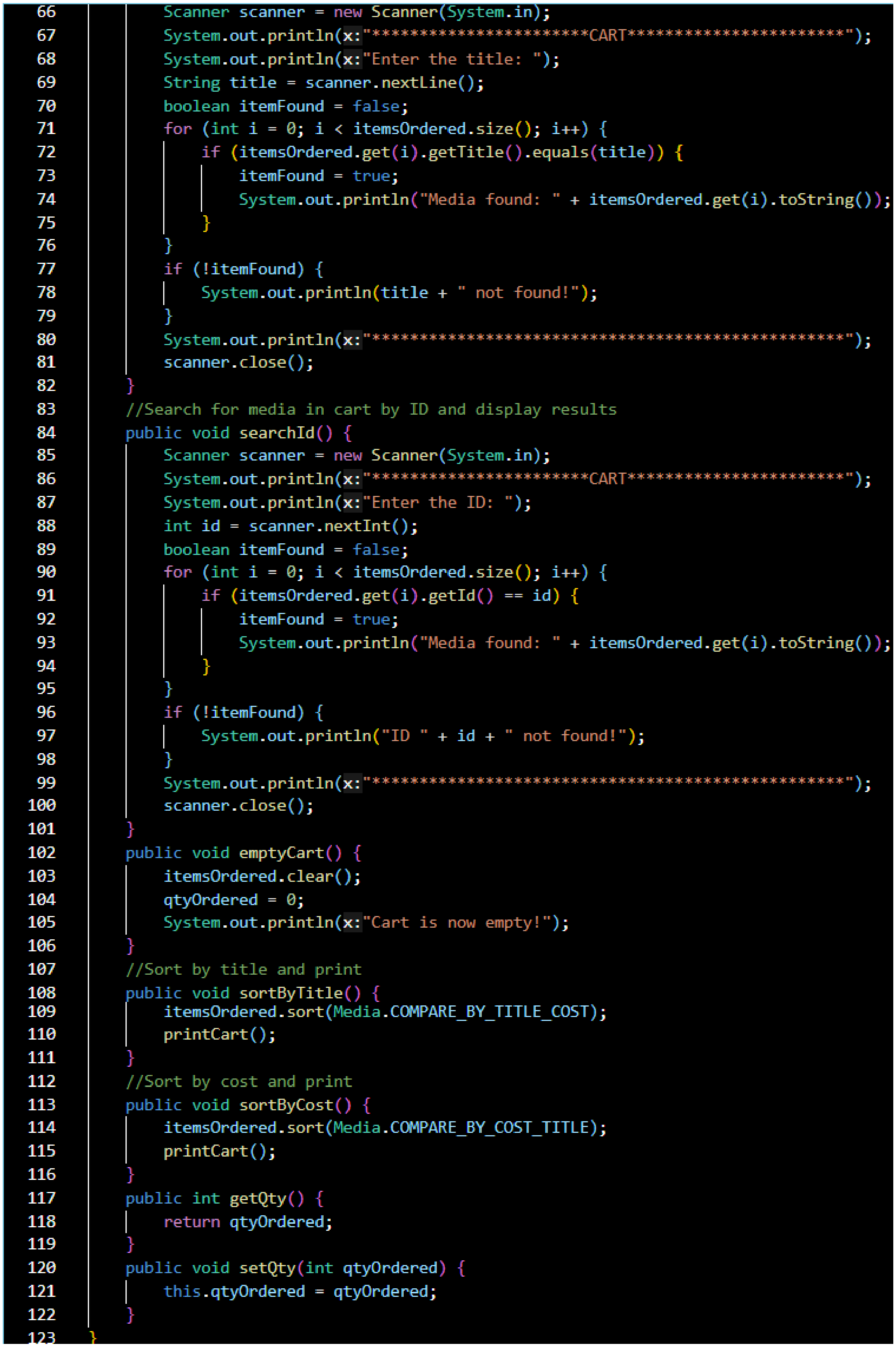
5. Update the Cart class to work with Media

- 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
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.cart;
import java.util.*;
import AimsProject.hust.soict.hedspi.aims.media.*;
public class Cart {
    private static final int MAX_ITEM = 20;
    //Attribute
    private List<Media> itemsOrdered = new ArrayList<Media>(MAX_ITEM);
    private int qtyOrdered = 0;
    //Add new media
    public void addMedia(Media media) {
        if (itemsOrdered.size() < MAX_ITEM) {
            itemsOrdered.add(media);
            qtyOrdered++;
            System.out.println(media.getTitle() + " has been added!");
            //Check if the cart is almost full
            if (MAX_ITEM - itemsOrdered.size() == 1) {
                System.out.println(x:"Warning! The cart is almost full.");
            }
        } else {
            System.out.println(x:"The cart is full! Cannot add more items.");
        }
    }
    //Remove media
    public void removeMedia(Media media) {
        if (itemsOrdered.isEmpty()) {
            System.out.println(x:"ERROR! Your cart is empty!");
            return;
        }
        boolean found = false;
        for (int i = 0; i < itemsOrdered.size(); i++) {
            if (itemsOrdered.get(i).equals(media)) {
                itemsOrdered.remove(i);
                qtyOrdered--;
                System.out.println(media.getTitle() + " has been removed!");
                found = true;
                break;
            }
        }
        if (!found) {
            System.out.println(media.getTitle() + " is not in the cart.");
        }
    }
    public List<Media> getItemsInCart() {
        return itemsOrdered;
    }
    //Calculate total cost
    public float totalCost() {
        float total = 0.0f;
        for (int i = 0; i < itemsOrdered.size(); i++) {
            total += itemsOrdered.get(i).getCost();
        }
        return total;
    }
    //Print list of ordered items in cart, the price of items, and total price
    public void printCart() {
        System.out.println(x:"**********************CART***********************");
        System.out.println(x:"Ordered Items:");
        for (int i = 0 ; i < itemsOrdered.size(); i++) {
            System.out.println((i+1) + itemsOrdered.get(i).toString());        }
        System.out.println("Total cost: $" + totalCost());
        System.out.println(x:"***************************************************");
    }
    //Search for media in cart by title and display results
    public void searchTitle() {
```

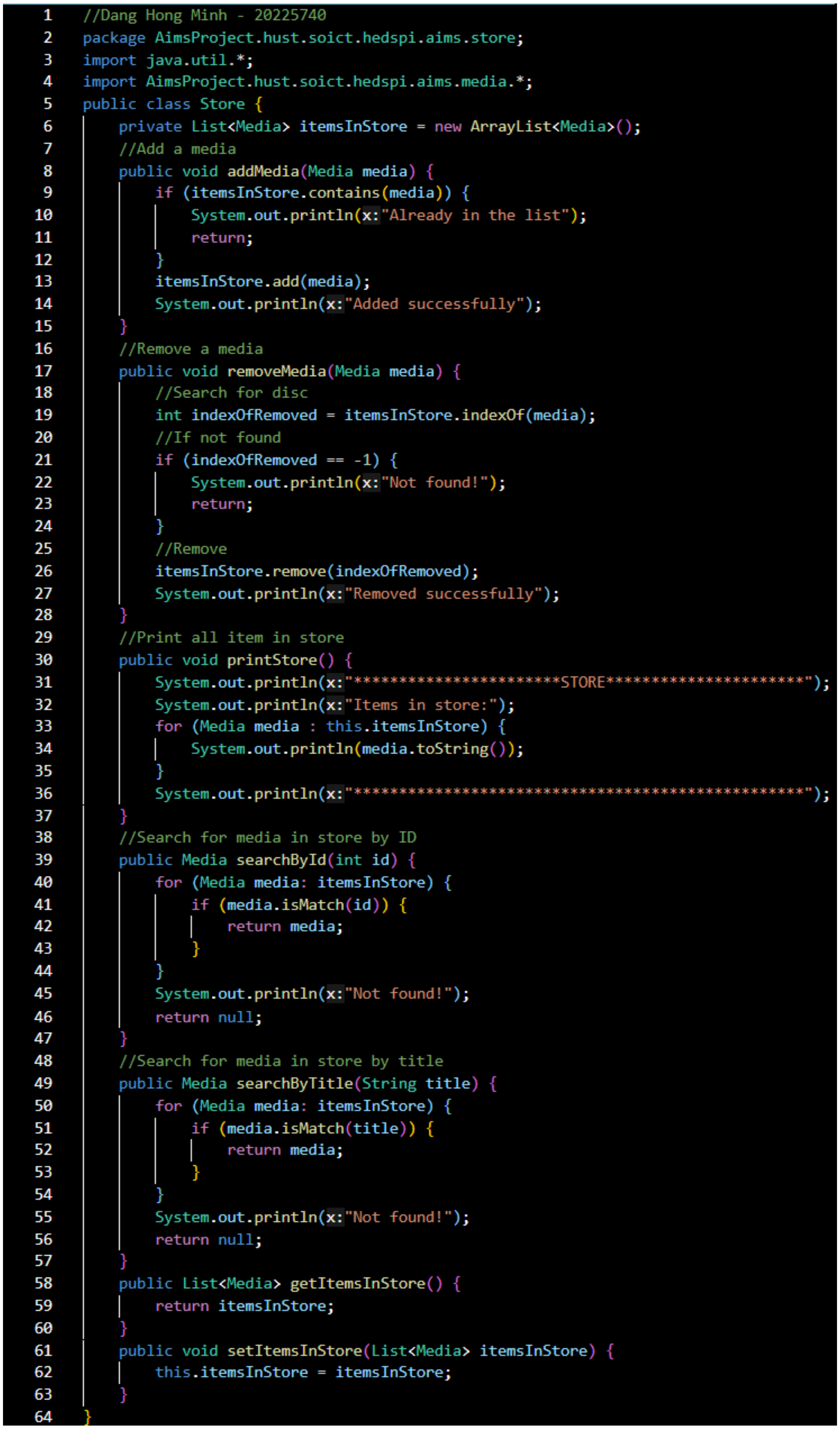
Figure 12: Cart Class 1



```java
        Scanner scanner = new Scanner(System.in);
        System.out.println(x:"***********************CART************************");
        System.out.println(x:"Enter the title: ");
        String title = scanner.nextLine();
        boolean itemFound = false;
        for (int i = 0; i < itemsOrdered.size(); i++) {
            if (itemsOrdered.get(i).getTitle().equals(title)) {
                itemFound = true;
                System.out.println("Media found: " + itemsOrdered.get(i).toString());
            }
        }
        if (!itemFound) {
            System.out.println(title + " not found!");
        }
        System.out.println(x:"***************************************************");
        scanner.close();
    }
    //Search for media in cart by ID and display results
    public void searchId() {
        Scanner scanner = new Scanner(System.in);
        System.out.println(x:"***********************CART************************");
        System.out.println(x:"Enter the ID: ");
        int id = scanner.nextInt();
        boolean itemFound = false;
        for (int i = 0; i < itemsOrdered.size(); i++) {
            if (itemsOrdered.get(i).getId() == id) {
                itemFound = true;
                System.out.println("Media found: " + itemsOrdered.get(i).toString());
            }
        }
        if (!itemFound) {
            System.out.println("ID " + id + " not found!");
        }
        System.out.println(x:"***************************************************");
        scanner.close();
    }
    public void emptyCart() {
        itemsOrdered.clear();
        qtyOrdered = 0;
        System.out.println(x:"Cart is now empty!");
    }
    //Sort by title and print
    public void sortByTitle() {
        itemsOrdered.sort(Media.COMPARE_BY_TITLE_COST);
        printCart();
    }
    //Sort by cost and print
    public void sortByCost() {
        itemsOrdered.sort(Media.COMPARE_BY_COST_TITLE);
        printCart();
    }
    public int getQty() {
        return qtyOrdered;
    }
    public void setQty(int qtyOrdered) {
        this.qtyOrdered = qtyOrdered;
    }
}
```

Figure 13: Cart Class 2

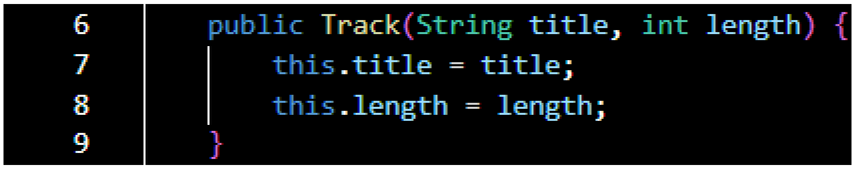
6. Update the Store class to work with Media



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.store;
import java.util.*;
import AimsProject.hust.soict.hedspi.aims.media.*;
public class Store {
    private List<Media> itemsInStore = new ArrayList<Media>();
    //Add a media
    public void addMedia(Media media) {
        if (itemsInStore.contains(media)) {
            System.out.println(x:"Already in the list");
            return;
        }
        itemsInStore.add(media);
        System.out.println(x:"Added successfully");
    }
    //Remove a media
    public void removeMedia(Media media) {
        //Search for disc
        int indexOfRemoved = itemsInStore.indexOf(media);
        //If not found
        if (indexOfRemoved == -1) {
            System.out.println(x:"Not found!");
            return;
        }
        //Remove
        itemsInStore.remove(indexOfRemoved);
        System.out.println(x:"Removed successfully");
    }
    //Print all item in store
    public void printStore() {
        System.out.println(x:"**********************STORE**********************");
        System.out.println(x:"Items in store:");
        for (Media media : this.itemsInStore) {
            System.out.println(media.toString());
        }
        System.out.println(x:"*************************************************");
    }
    //Search for media in store by ID
    public Media searchById(int id) {
        for (Media media: itemsInStore) {
            if (media.isMatch(id)) {
                return media;
            }
        }
        System.out.println(x:"Not found!");
        return null;
    }
    //Search for media in store by title
    public Media searchByTitle(String title) {
        for (Media media: itemsInStore) {
            if (media.isMatch(title)) {
                return media;
            }
        }
        System.out.println(x:"Not found!");
        return null;
    }
    public List<Media> getItemsInStore() {
        return itemsInStore;
    }
    public void setItemsInStore(List<Media> itemsInStore) {
        this.itemsInStore = itemsInStore;
    }
}
```

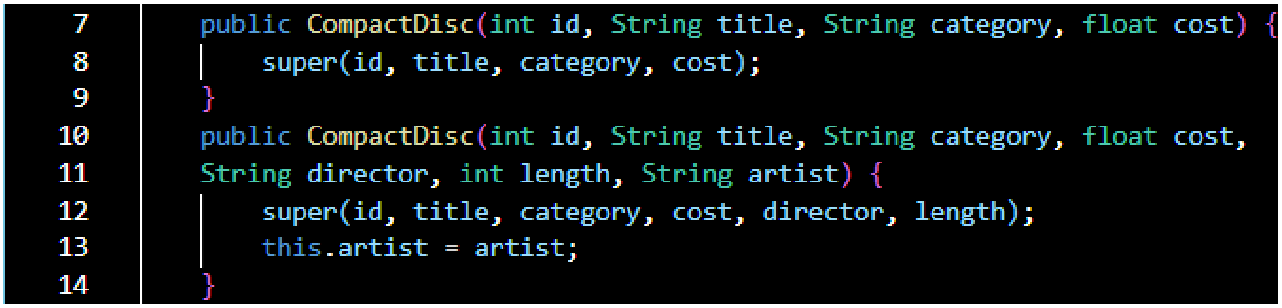
Figure 14: Store Class

7. Constructors of whole classes and parent classes



```java
public Track(String title, int length) {
    this.title = title;
    this.length = length;
}
```

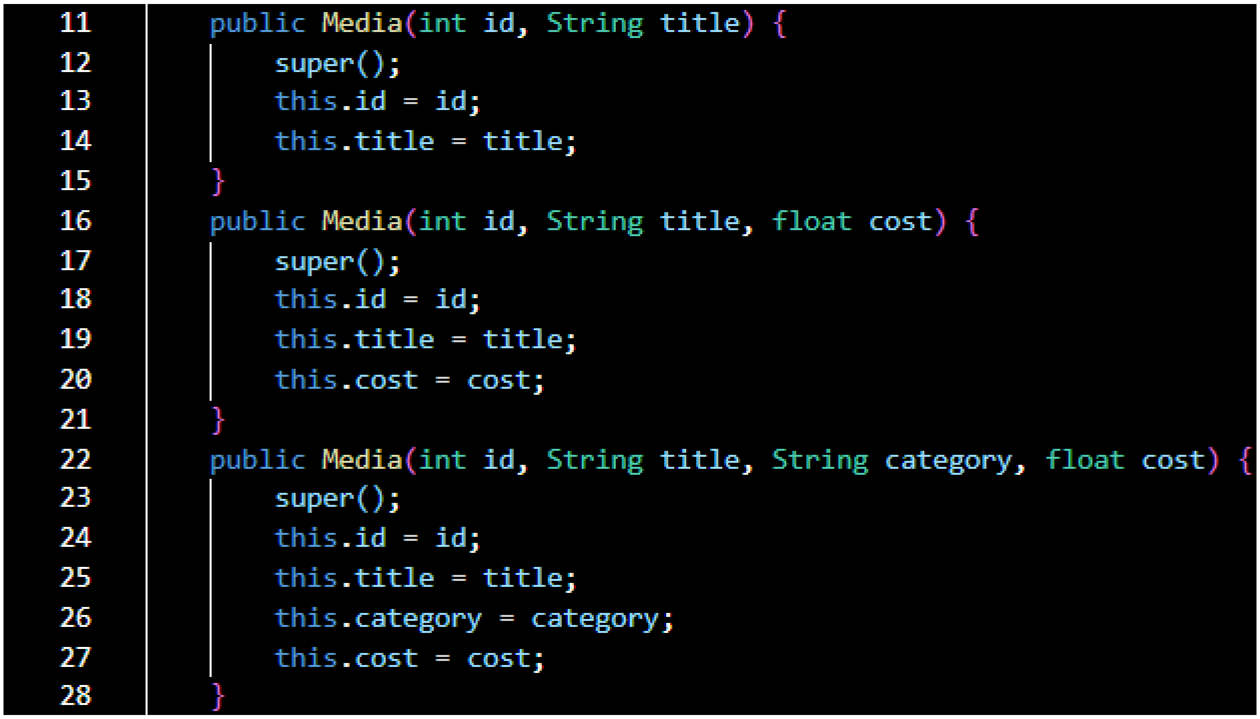
Figure 15: Constructor của Track Class



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

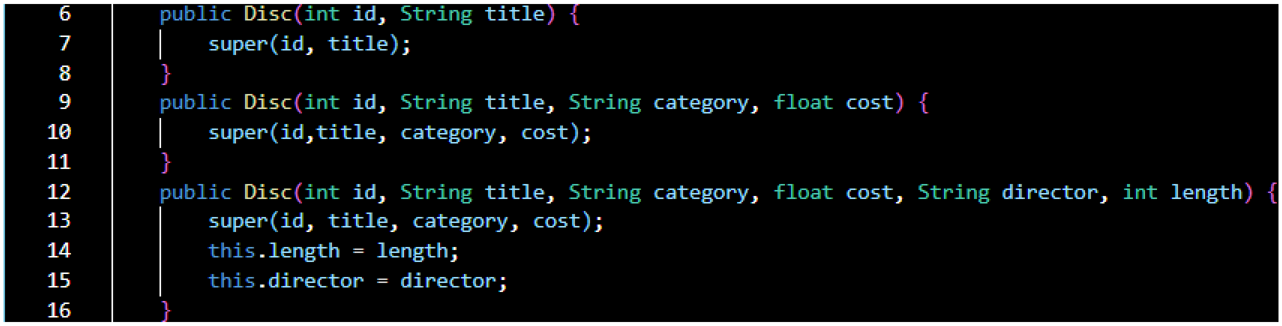
Figure 16: Constructor của CompactDisc Class

- 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) {
    super();
    this.id = id;
    this.title = title;
}
public Media(int id, String title, float cost) {
    super();
    this.id = id;
    this.title = title;
    this.cost = cost;
}
public Media(int id, String title, String category, float cost) {
    super();
    this.id = id;
    this.title = title;
    this.category = category;
    this.cost = cost;
}
```

Figure 17: Constructor của Media Class

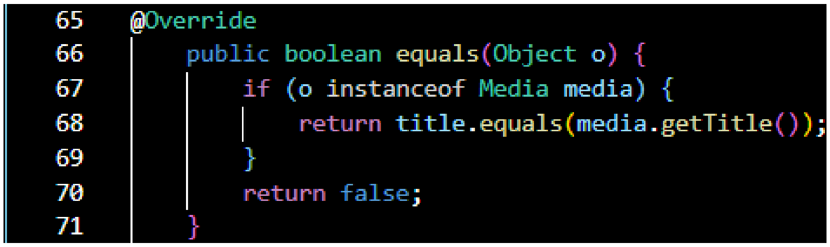


```java
    public Disc(int id, String title) {
        super(id, title);
    }
    public Disc(int id, String title, String category, float cost) {
        super(id,title, category, cost);
    }
    public Disc(int id, String title, String category, float cost, String director, int length) {
      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

Figure 18: Constructor của Disc Class

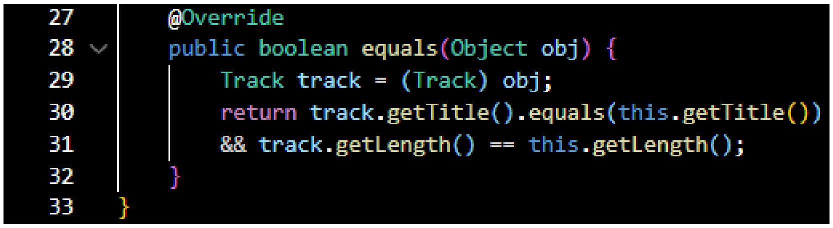
8. Unique item in a list

- 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 lặp vào danh sách.



```java
@Override
    public boolean equals(Object o) {
        if (o instanceof Media media) {
            return title.equals(media.getTitle());
        }
        return false;
    }
```

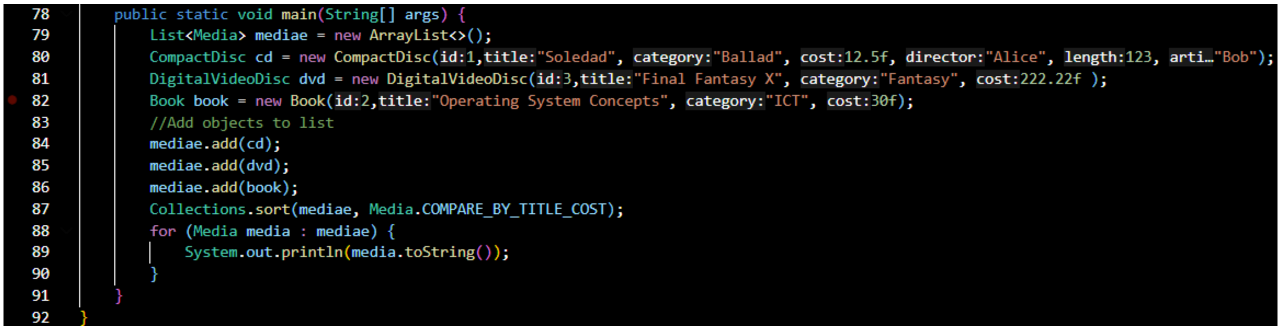
Figure 19: Override equals() trong Media Class



```java
    @Override
    public boolean equals(Object obj) {
        Track track = (Track) obj;
        return track.getTitle().equals(this.getTitle())
        && track.getLength() == this.getLength();
    }
}
```

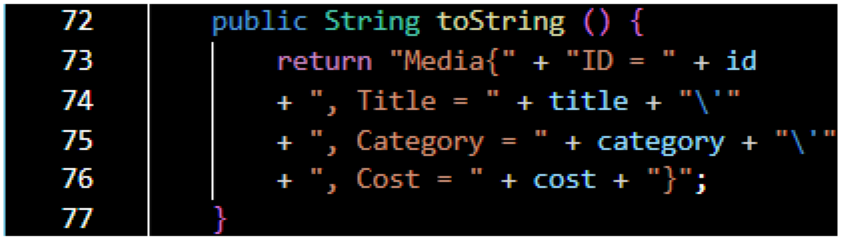
Figure 20: Override equals() trong Track Class

9. Polymorphism with toString() method



```java
    public static void main(String[] args) {
        List<Media> mediae = new ArrayList<>();
        CompactDisc cd = new CompactDisc(id:1,title:"Soledad", category:"Ballad", cost:12.5f, director:"Alice", length:123, arti…"Bob");
        DigitalVideoDisc dvd = new DigitalVideoDisc(id:3,title:"Final Fantasy X", category:"Fantasy", cost:222.22f );
        Book book = new Book(id:2,title:"Operating System Concepts", category:"ICT", cost:30f);
        //Add objects to list
        mediae.add(cd);
        mediae.add(dvd);
        mediae.add(book);
        Collections.sort(mediae, Media.COMPARE_BY_TITLE_COST);
        for (Media media : mediae) {
            System.out.println(media.toString());
        }
    }
}
```

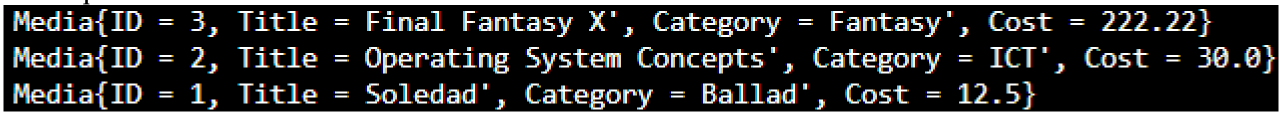
Figure 21: Code mô phỏng Polymorphism



```
public String toString () {
    return "Media{" + "ID = " + id
    + ", Title = " + title + "\'"
    + ", Category = " + category + "\'"
    + ", Cost = " + cost + "}";
}
```

Figure 22: Override toString() trong Media Class

- Kết quả:



```
Media{ID = 3, Title = Final Fantasy X', Category = Fantasy', Cost = 222.22}
Media{ID = 2, Title = Operating System Concepts', Category = ICT', Cost = 30.0}
Media{ID = 1, Title = Soledad', Category = Ballad', Cost = 12.5}
```

Figure 23: Kết quả demo Polymorphism

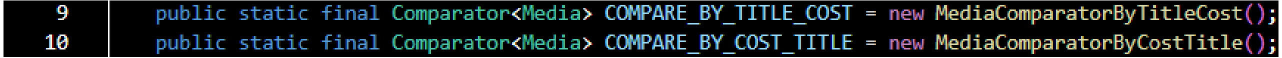
- Lớp Media là lớp cơ sở được kế thừa bởi các lớp cụ thể hơn là CompactDisc, DigitalVideoDisc và Book. Khi khởi tạo các đối tượng cd, dvd, book thuộc lớp con rồi gán chúng cho biến kiểu Media, ta áp dụng kỹ thuật gọi là upcasting.

- 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 cart

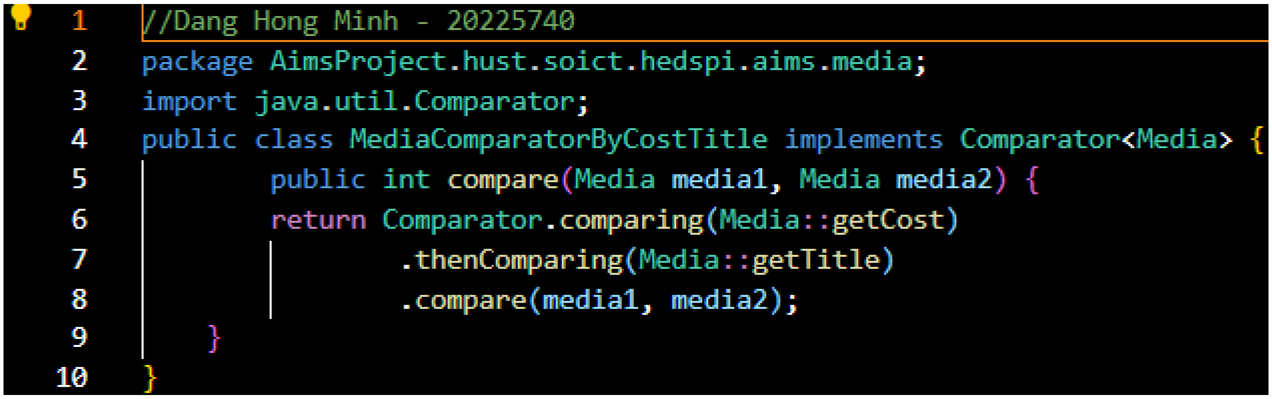
- Sắp xếp các media trong giỏ hàng theo hai tiêu chí:

+ 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



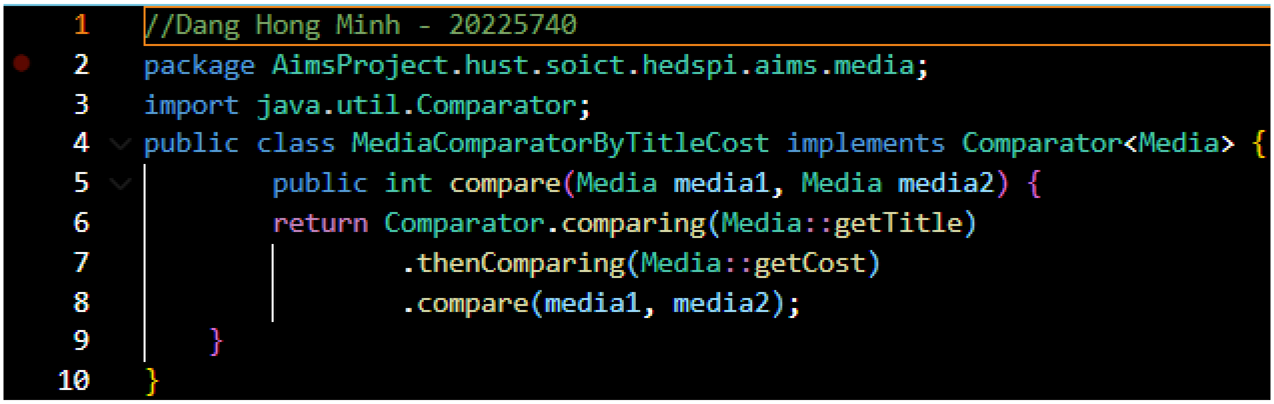
```
    public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();
    public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();
```

Figure 24: Thêm các biến so sánh thành thuộc tính trong Media Class



```
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
import java.util.Comparator;
public class MediaComparatorByCostTitle implements Comparator<Media> {
        public int compare(Media media1, Media media2) {
        return Comparator.comparing(Media::getCost)
                .thenComparing(Media::getTitle)
                .compare(media1, media2);
    }
}
```

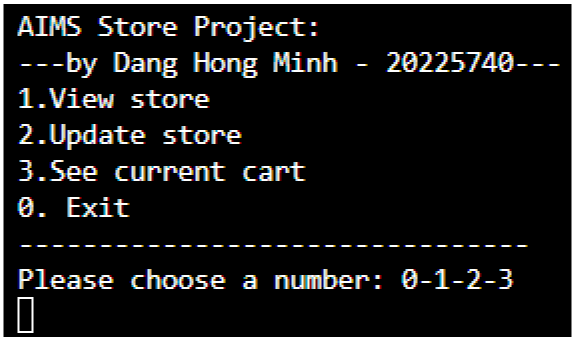
Figure 25: MediaComparatorByCostTitle Class



```java
//Dang Hong Minh - 20225740
package AimsProject.hust.soict.hedspi.aims.media;
import java.util.Comparator;
public class MediaComparatorByTitleCost implements Comparator<Media> {
        public int compare(Media media1, Media media2) {
        return Comparator.comparing(Media::getTitle)
                .thenComparing(Media::getCost)
                .compare(media1, media2);
    }
}
```

Figure 26: *MediaComparatorByTitleCost Class*

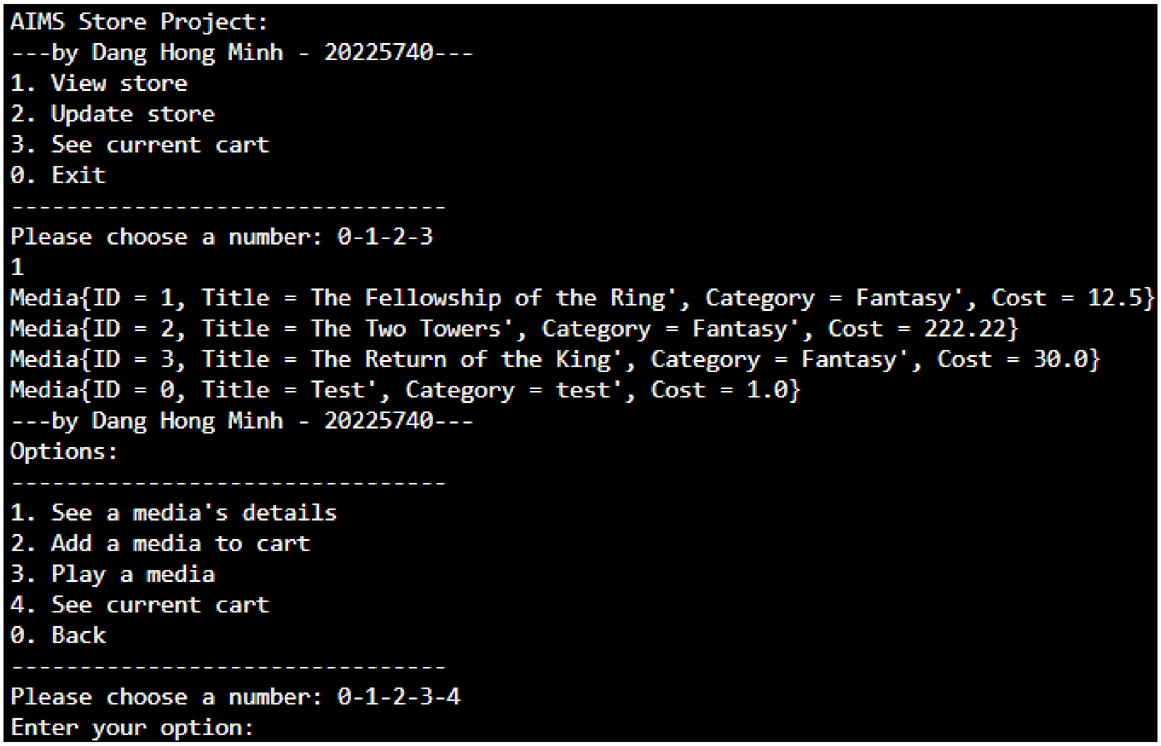
11. Create a complete console application in the Aims class



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1.View store
2.Update store
3.See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
```

Figure 27: Màn hình chính

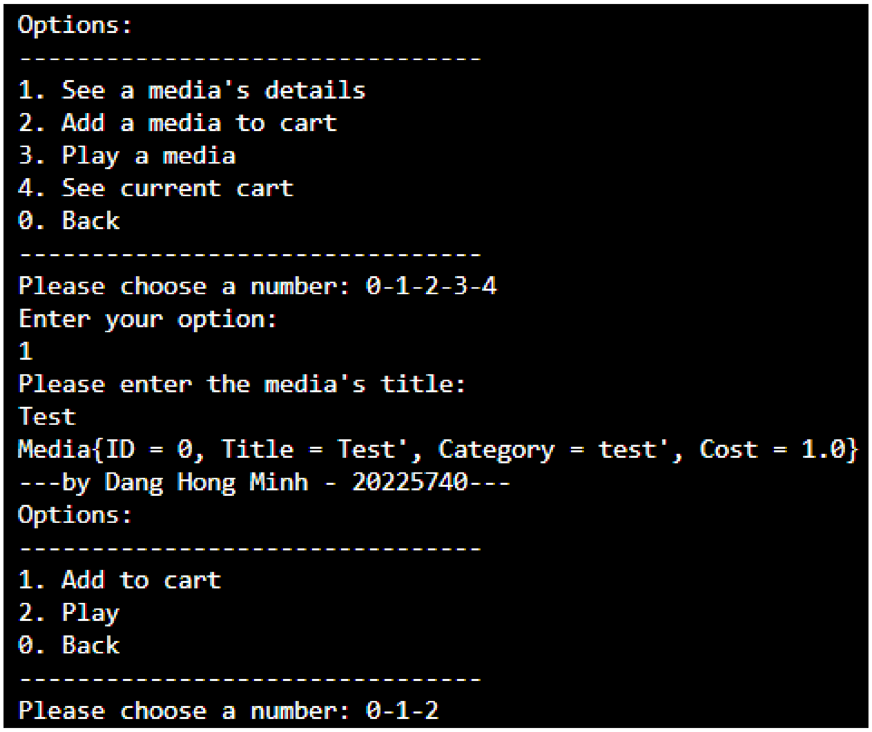
11.1 Người dùng chọn 1: View store



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1
Media{ID = 1, Title = The Fellowship of the Ring', Category = Fantasy', Cost = 12.5}
Media{ID = 2, Title = The Two Towers', Category = Fantasy', Cost = 222.22}
Media{ID = 3, Title = The Return of the King', Category = Fantasy', Cost = 30.0}
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
---by Dang Hong Minh - 20225740---
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
Enter your option:
```

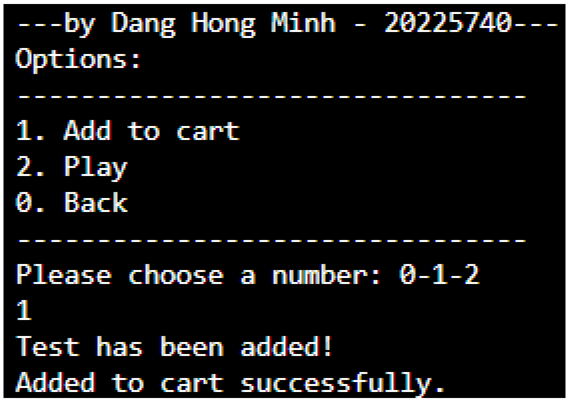
Figure 28: Vào menu View Store

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



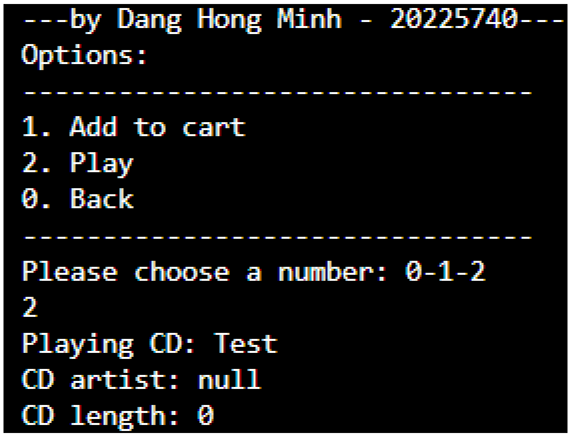
```
Options:
--------------------------------
1. See a media's details
2. Add a media to cart
3. Play a media
4. See current cart
0. Back
--------------------------------
Please choose a number: 0-1-2-3-4
Enter your option:
1
Please enter the media's title:
Test
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
---by Dang Hong Minh - 20225740---
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
```

Figure 29: See a media's details



```
---by Dang Hong Minh - 20225740---
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
1
Test has been added!
Added to cart successfully.
```

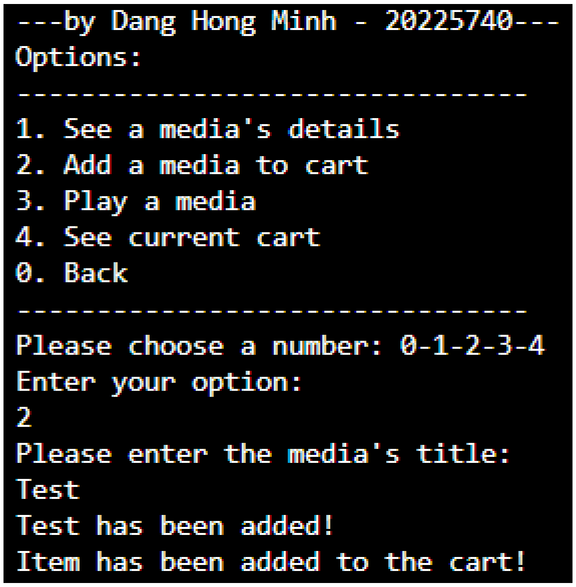
Figure 30: Add to cart



```
---by Dang Hong Minh - 20225740---
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
2
Playing CD: Test
CD artist: null
CD length: 0
```

Figure 31: Play

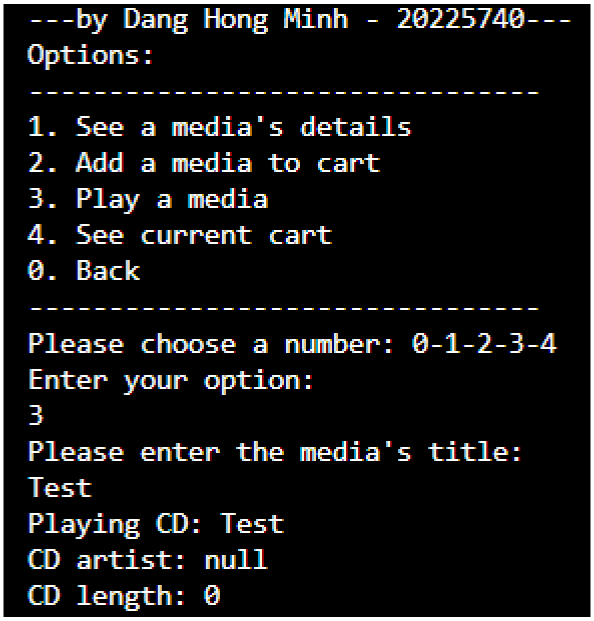
11.1.2 Người dùng chọn 2: Add a media to the cart



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
2
Please enter the media's title:
Test
Test has been added!
Item has been added to the cart!
```

Figure 32: Add a media to cart

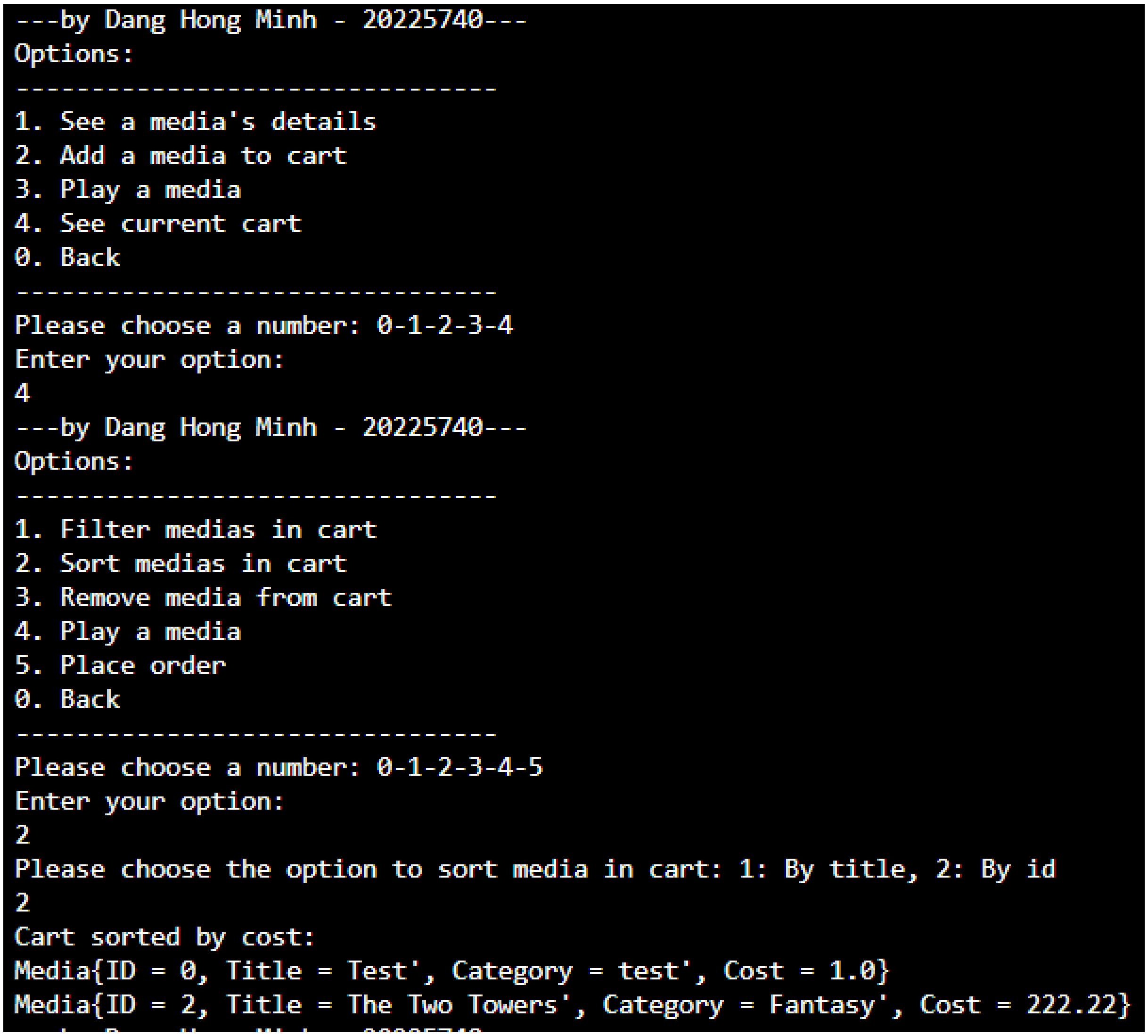
11.1.3 Người dùng chọn 3: Play a media



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
3
Please enter the media's title:
Test
Playing CD: Test
CD artist: null
CD length: 0
```

Figure 33: Play a media

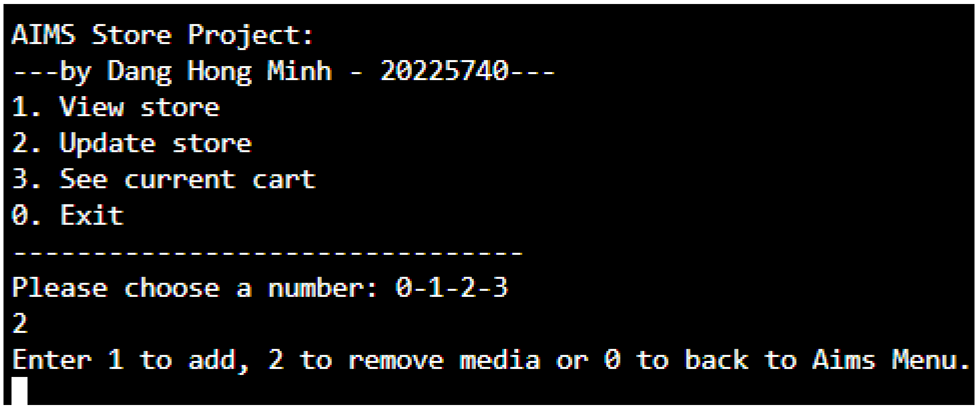
11.1.4 Người dùng chọn 4: See current cart



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
4
---by Dang Hong Minh - 20225740---
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
Enter your option:
2
Please choose the option to sort media in cart: 1: By title, 2: By id
2
Cart sorted by cost:
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
Media{ID = 2, Title = The Two Towers', Category = Fantasy', Cost = 222.22}
```

Figure 34: See current cart after sort

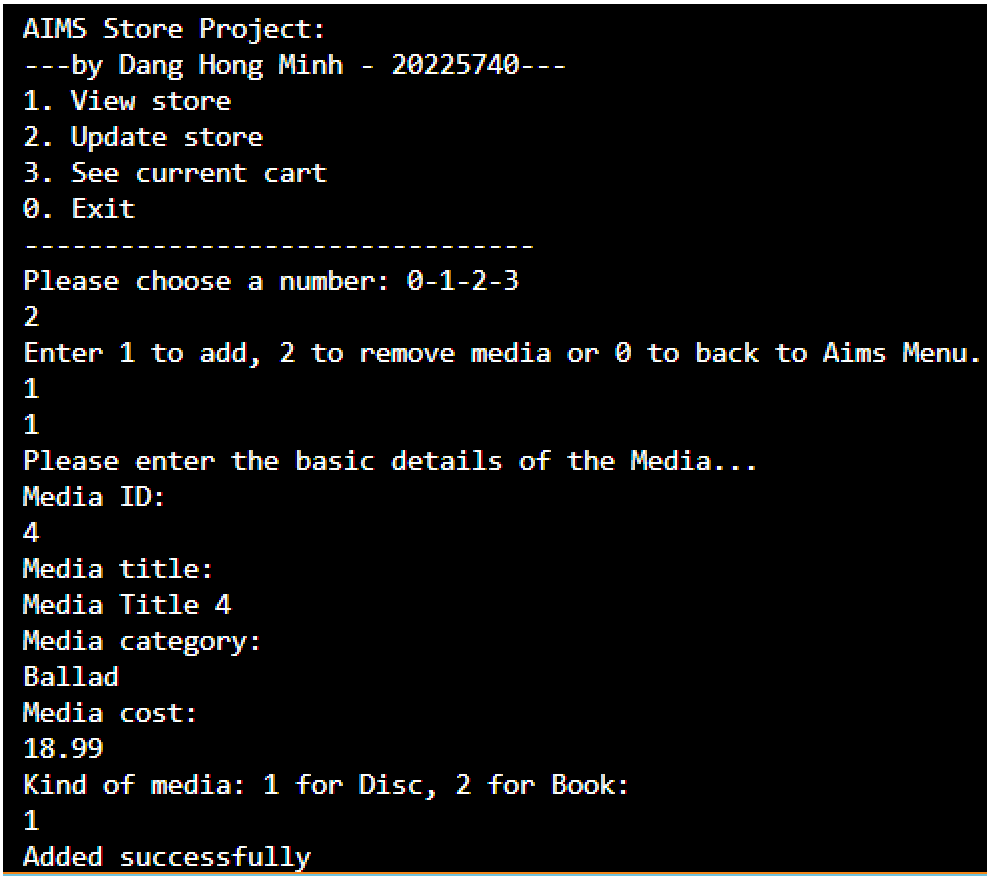
11.2 Người dùng chọn 2: Update store



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
2
Enter 1 to add, 2 to remove media or 0 to back to Aims Menu.
```

Figure 35: Vào menu Update Store

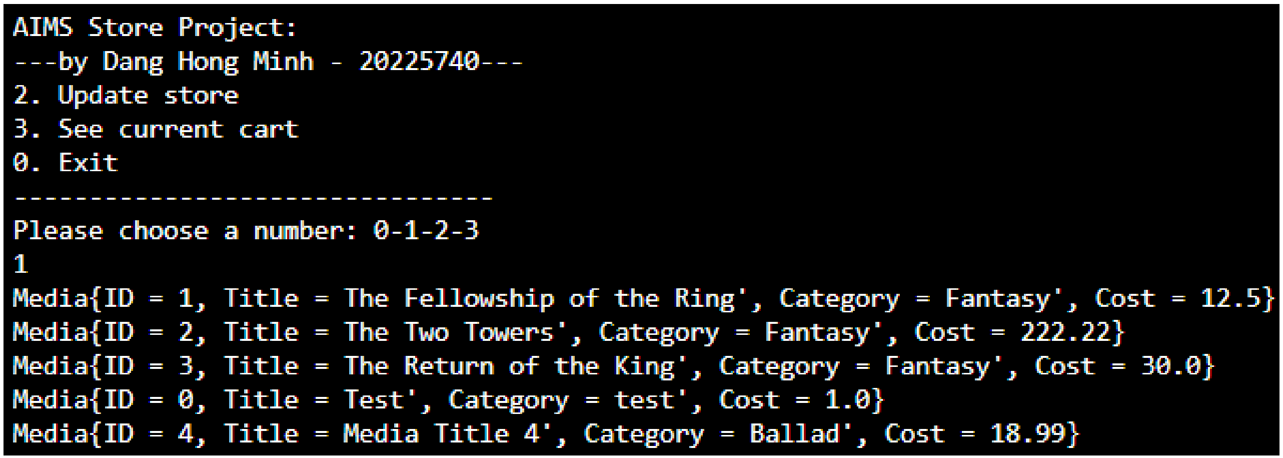
11.2.1 Người dùng chọn 1: Add a media to the store



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
2
Enter 1 to add, 2 to remove media or 0 to back to Aims Menu.
1
1
Please enter the basic details of the Media...
Media ID:
4
Media title:
Media Title 4
Media category:
Ballad
Media cost:
18.99
Kind of media: 1 for Disc, 2 for Book:
1
Added successfully
```

Figure 36: Add a media to store

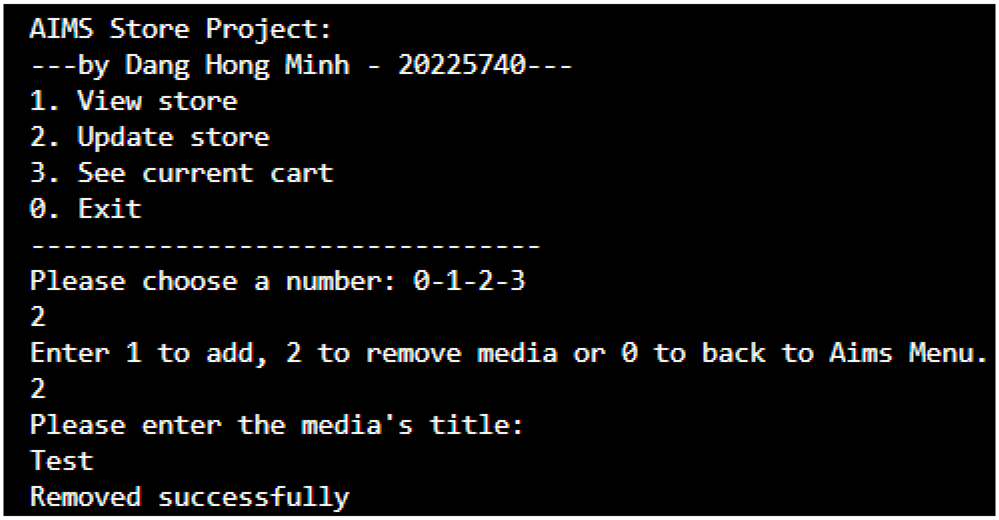
=> Kết quả sau khi thêm:



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1
Media{ID = 1, Title = The Fellowship of the Ring', Category = Fantasy', Cost = 12.5}
Media{ID = 2, Title = The Two Towers', Category = Fantasy', Cost = 222.22}
Media{ID = 3, Title = The Return of the King', Category = Fantasy', Cost = 30.0}
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
Media{ID = 4, Title = Media Title 4', Category = Ballad', Cost = 18.99}
```

Figure 37: Kết quả sau khi thêm

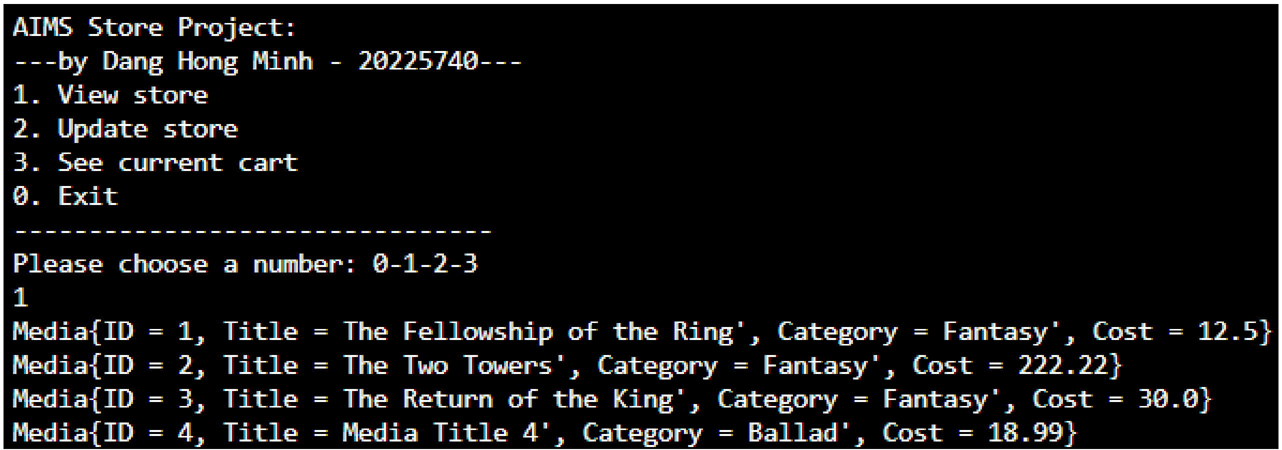
11.2.2 Người dùng chọn 2: Remove a media from the store



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
2
Enter 1 to add, 2 to remove media or 0 to back to Aims Menu.
2
Please enter the media's title:
Test
Removed successfully
```

Figure 38: Remove a media from store

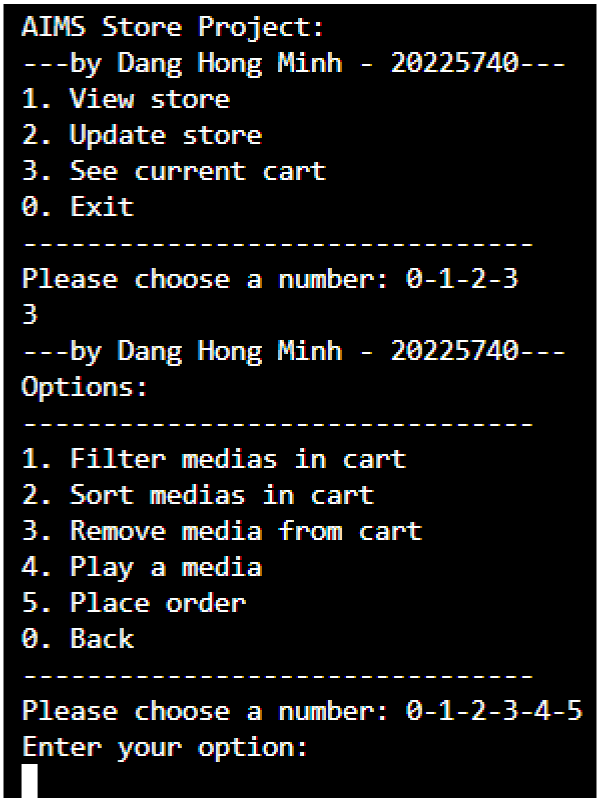
=> Kết quả sau khi xóa:



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1
Media{ID = 1, Title = The Fellowship of the Ring', Category = Fantasy', Cost = 12.5}
Media{ID = 2, Title = The Two Towers', Category = Fantasy', Cost = 222.22}
Media{ID = 3, Title = The Return of the King', Category = Fantasy', Cost = 30.0}
Media{ID = 4, Title = Media Title 4', Category = Ballad', Cost = 18.99}
```

Figure 39: Kết quả sau khi xóa

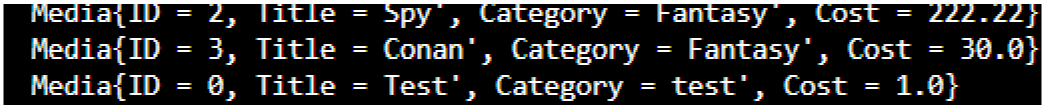
11.3 Người dùng chọn 3: See current cart



```
AIMS Store Project:
---by Dang Hong Minh - 20225740---
1. View store
2. Update store
3. See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
3
---by Dang Hong Minh - 20225740---
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
Enter your option:
```

Figure 40: Vào menu See current cart

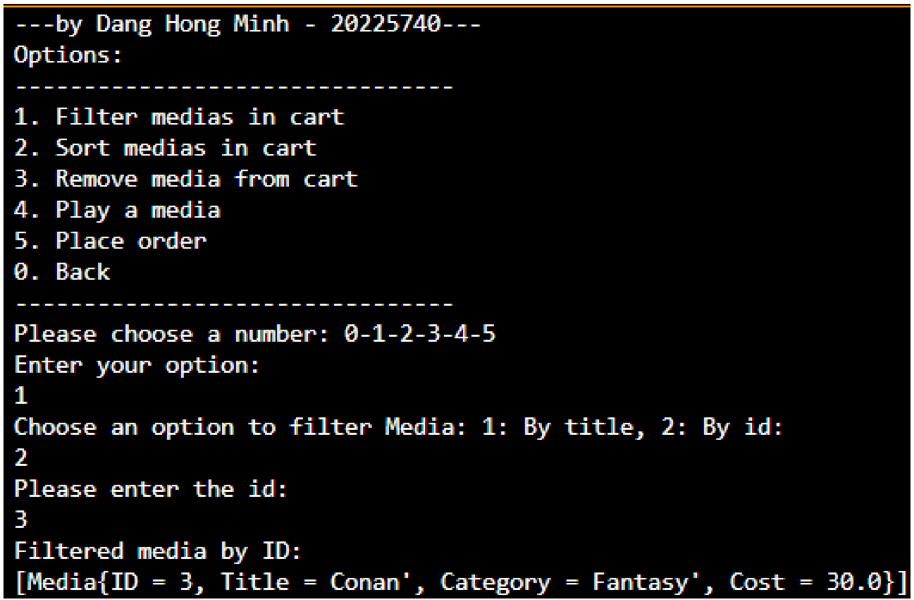
- Giả sử lúc này trong Cart sẽ có các Media sau:



```
Media{ID = 2, Title = Spy', Category = Fantasy', Cost = 222.22}
Media{ID = 3, Title = Conan', Category = Fantasy', Cost = 30.0}
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
```

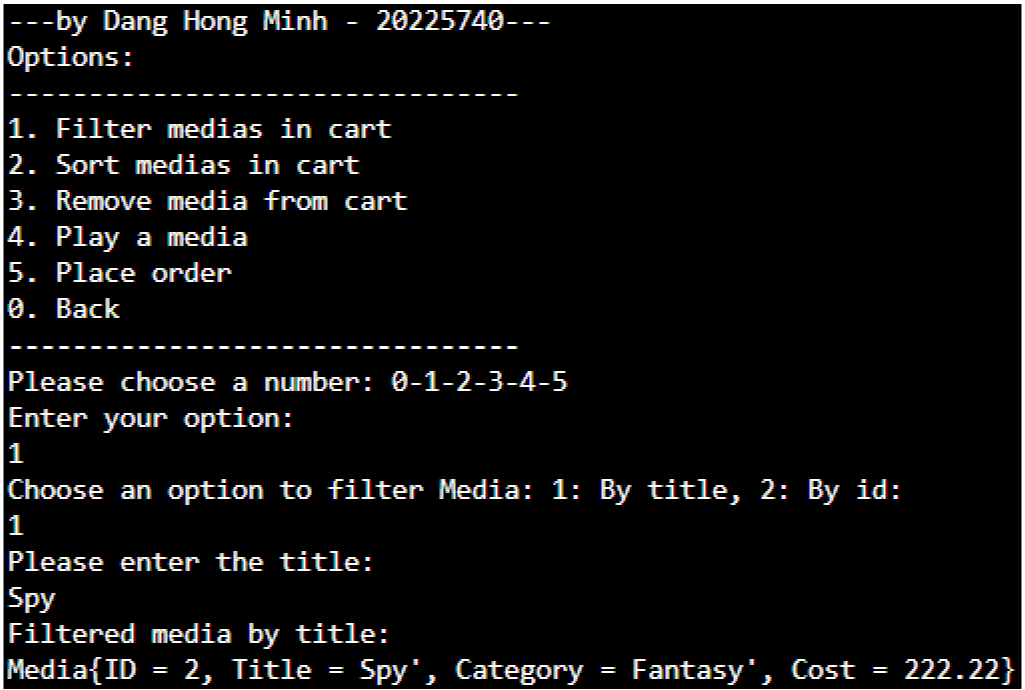
Figure 41: Media trong Cart

11.3.1 Người dùng chọn 1: Filter medias in cart



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
1
Choose an option to filter Media: 1: By title, 2: By id:
2
Please enter the id:
3
Filtered media by ID:
[Media{ID = 3, Title = Conan', Category = Fantasy', Cost = 30.0}]
```

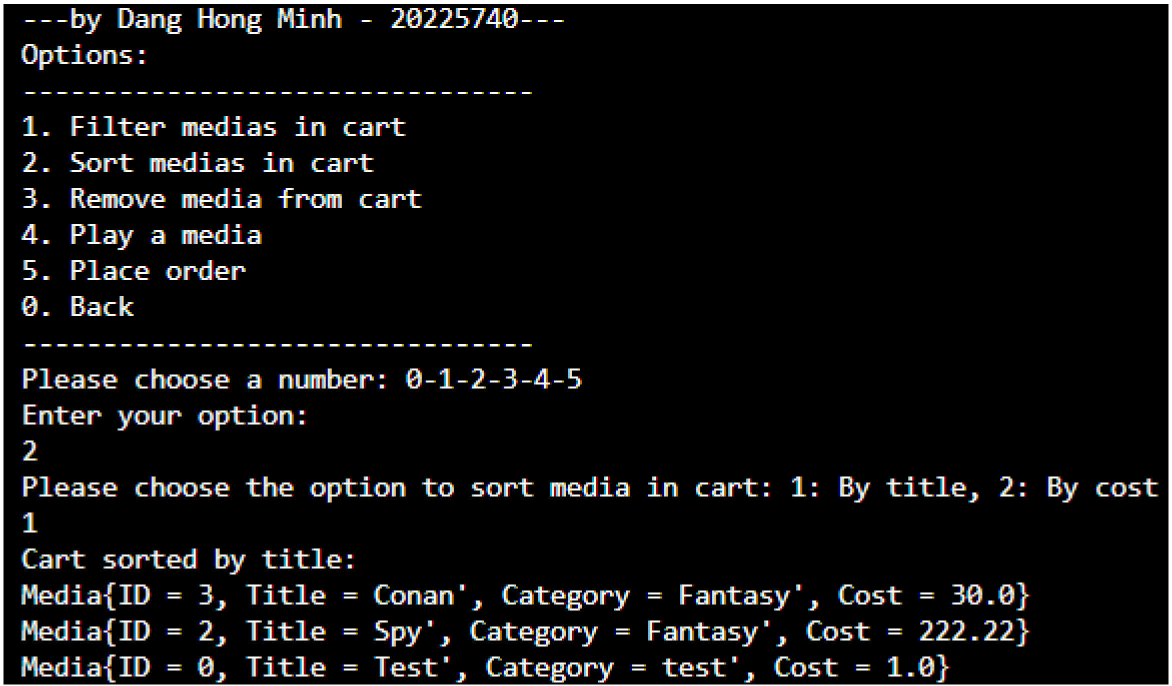
Figure 42: Lọc media trong Cart bằng ID



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
1
Choose an option to filter Media: 1: By title, 2: By id:
1
Please enter the title:
Spy
Filtered media by title:
Media{ID = 2, Title = Spy', Category = Fantasy', Cost = 222.22}
```

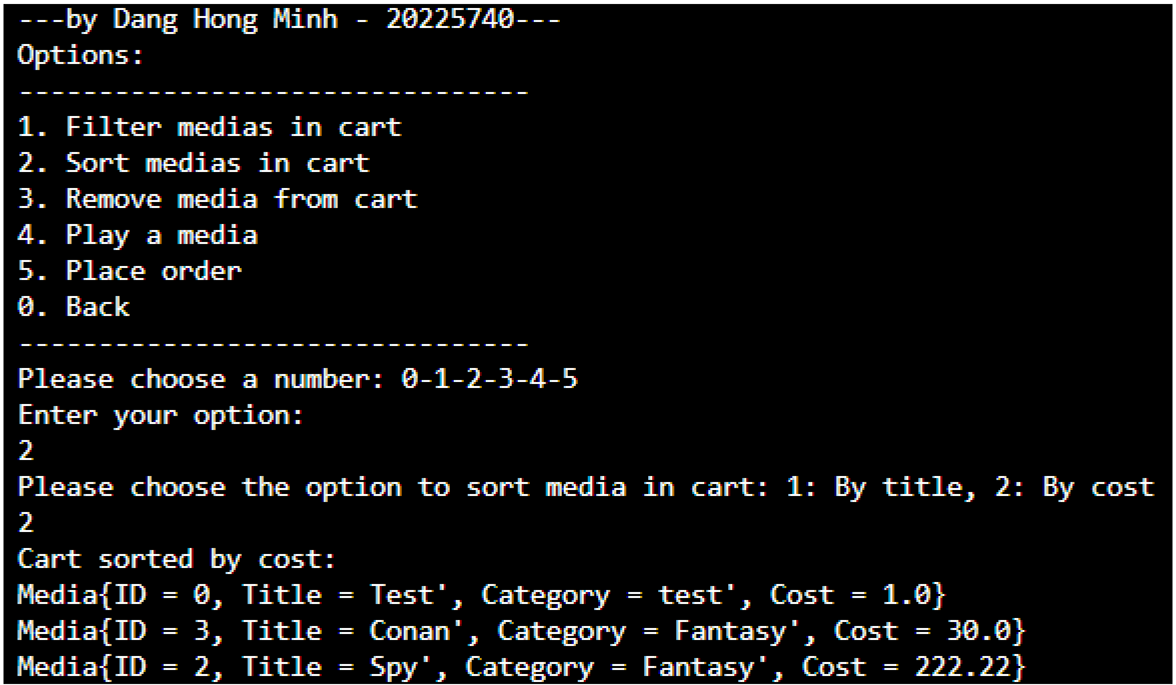
Figure 43: Lọc media trong Cart bằng title

11.3.2 Người dùng chọn 2: Sort medias in cart



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
2
Please choose the option to sort media in cart: 1: By title, 2: By cost
1
Cart sorted by title:
Media{ID = 3, Title = Conan', Category = Fantasy', Cost = 30.0}
Media{ID = 2, Title = Spy', Category = Fantasy', Cost = 222.22}
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
```

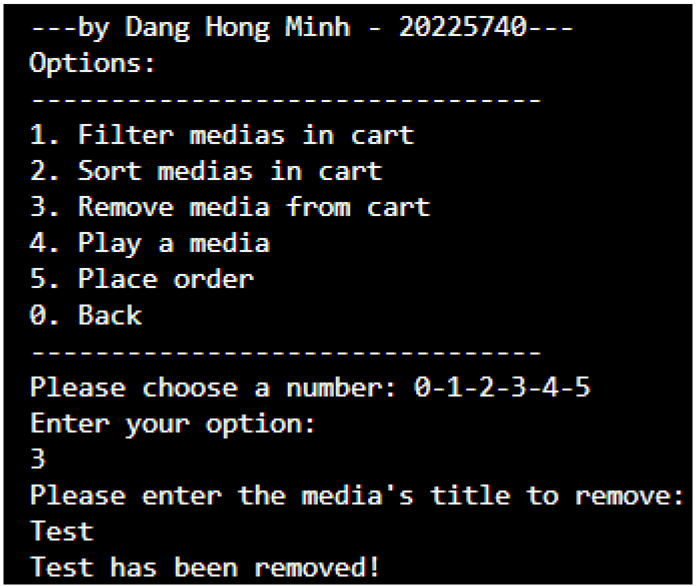
Figure 44: Sắp xếp media trong Cart bằng title



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
2
Please choose the option to sort media in cart: 1: By title, 2: By cost
2
Cart sorted by cost:
Media{ID = 0, Title = Test', Category = test', Cost = 1.0}
Media{ID = 3, Title = Conan', Category = Fantasy', Cost = 30.0}
Media{ID = 2, Title = Spy', Category = Fantasy', Cost = 222.22}
```

Figure 45: Sắp xếp media trong Cart bằng title

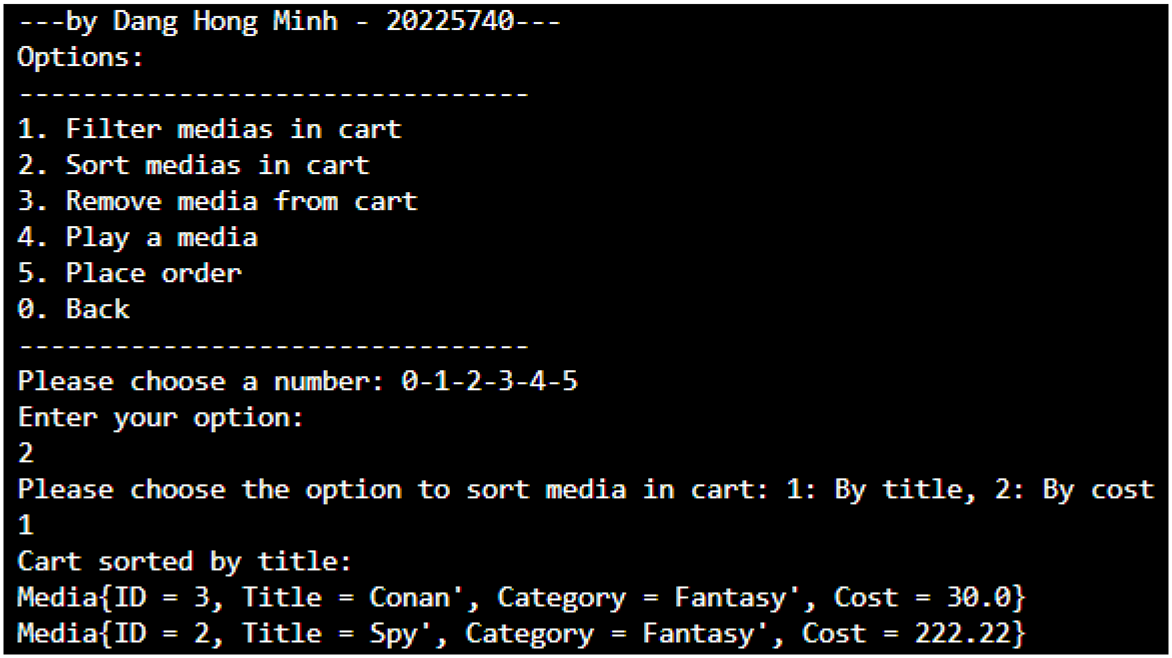
11.3.3 Người dùng chọn 3: Remove media from cart



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
3
Please enter the media's title to remove:
Test
Test has been removed!
```

Figure 46: Remove media bằng title

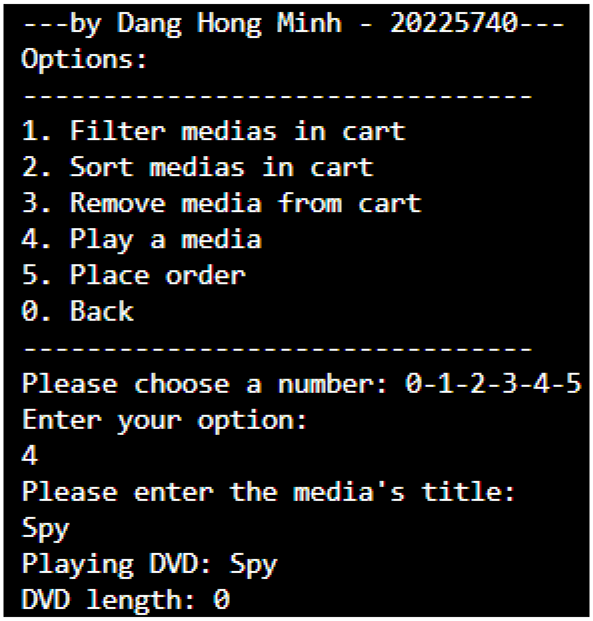
=> Kết quả sau khi xóa:



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
2
Please choose the option to sort media in cart: 1: By title, 2: By cost
1
Cart sorted by title:
Media{ID = 3, Title = Conan', Category = Fantasy', Cost = 30.0}
Media{ID = 2, Title = Spy', Category = Fantasy', Cost = 222.22}
```

Figure 47: Kết quả sau khi xóa

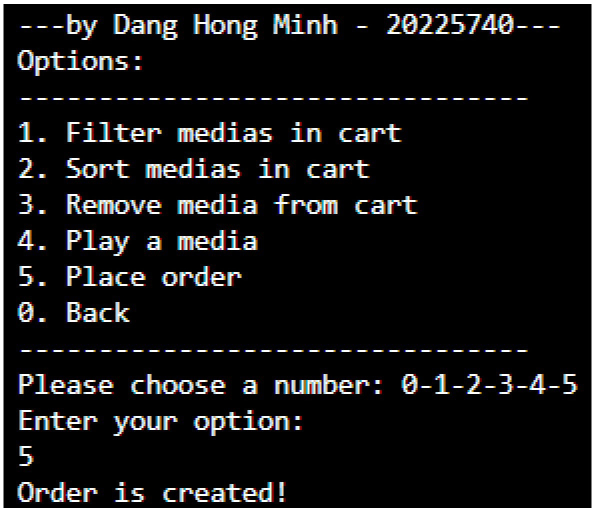
11.3.4 Người dùng chọn 4: Play a media



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
4
Please enter the media's title:
Spy
Playing DVD: Spy
DVD length: 0
```

Figure 48: Play a media in cart

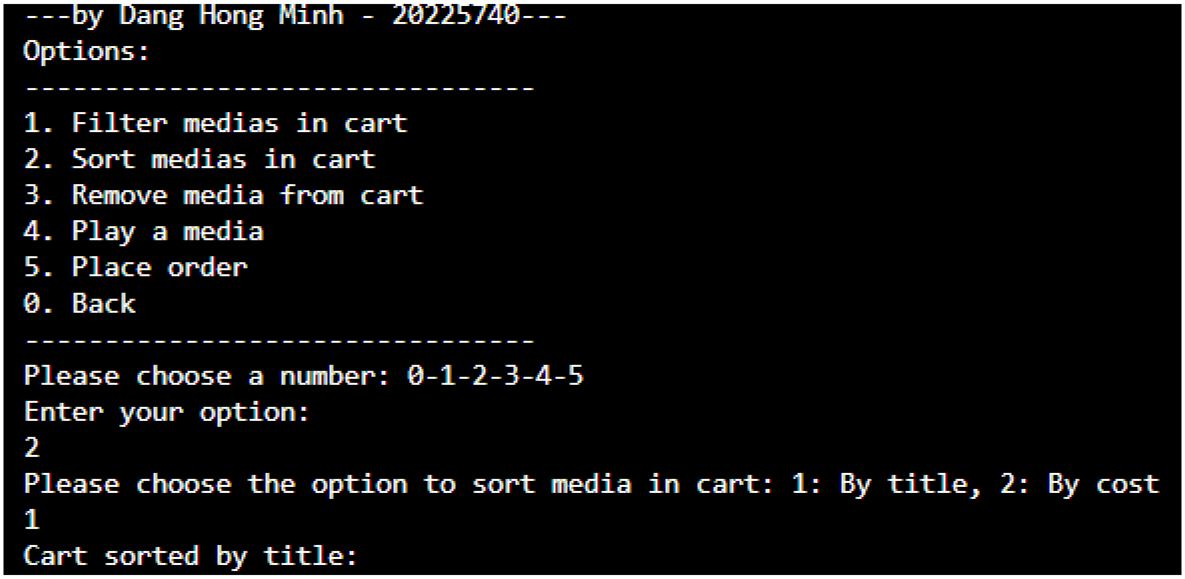
11.3.5 Người dùng chọn 5: Place order



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
5
Order is created!
```

Figure 49: Place order

=> Kết quả sau khi order:



```
---by Dang Hong Minh - 20225740---
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
Enter your option:
2
Please choose the option to sort media in cart: 1: By title, 2: By cost
1
Cart sorted by title:
```

Figure 50: Result after order

12. Class Diagram

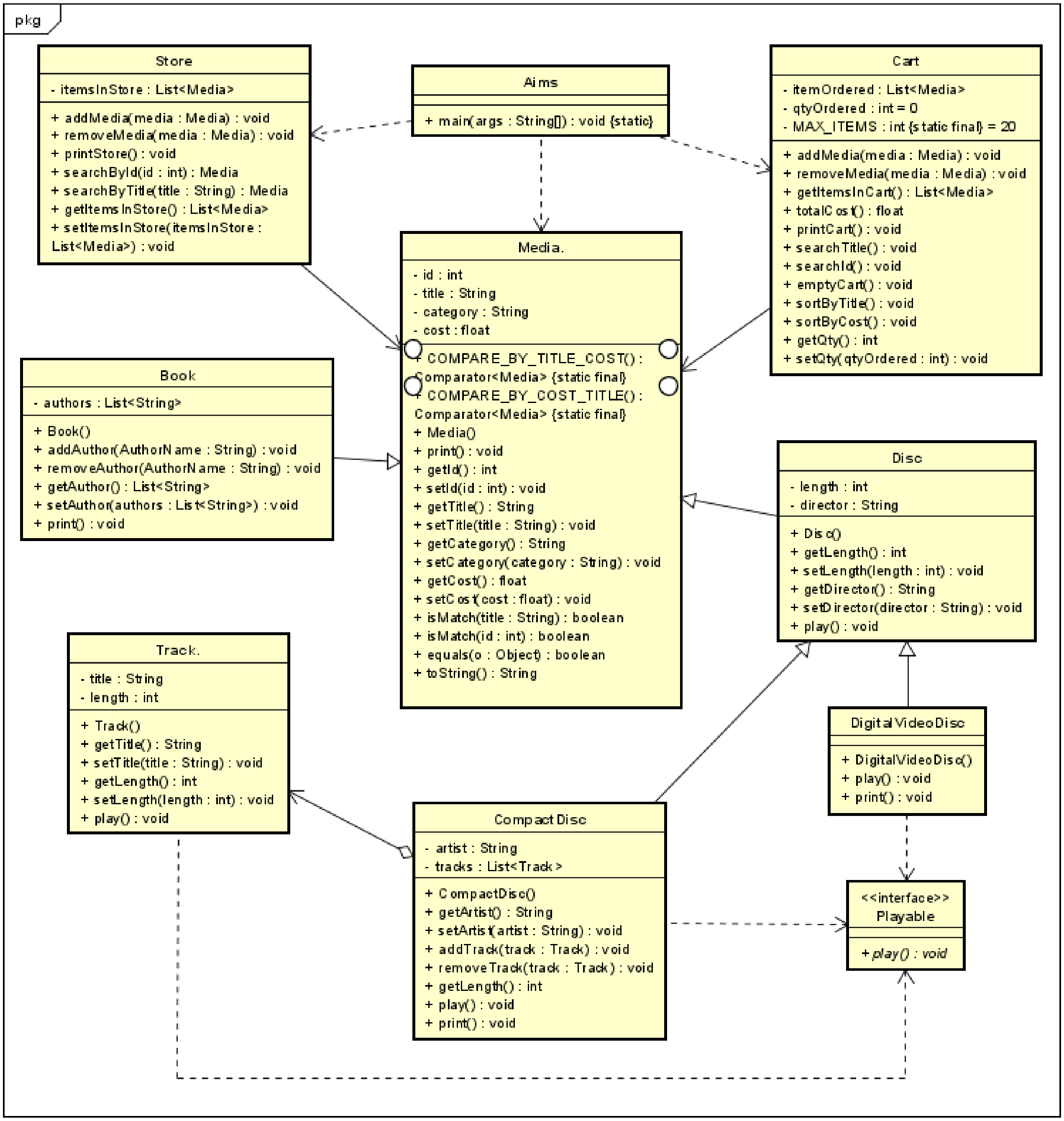

pkg
Store
- itemsInStore : List<Media>
+ addMedia(media : Media) : void
+ removeMedia(media : Media) : void
+ printStore() : void
+ searchById(id : int) : Media
+ searchByTitle(title : String) : Media
+ getItemsInStore() : List<Media>
+ setItemsInStore(itemsInStore : List<Media>) : void
Aims
+ main(args : String[]) : void {static}
Cart
- itemOrdered : List<Media>
- qtyOrdered : int = 0
- MAX_ITEMS : int {static final} = 20
+ addMedia(media : Media) : void
+ removeMedia(media : Media) : void
+ getItemsInCart() : List<Media>
+ totalCost() : float
+ printCart() : void
+ searchTitle() : void
+ searchId() : void
+ emptyCart() : void
+ sortByTitle() : void
+ sortByCost() : void
+ getQty() : int
+ setQty(qtyOrdered : int) : void
Media.
- id : int
- title : String
- category : String
- cost : float
+ COMPARE_BY_TITLE_COST() : Comparator<Media> {static final}
+ COMPARE_BY_COST_TITLE() : Comparator<Media> {static final}
+ Media()
+ print() : void
+ getId() : int
+ setId(id : int) : void
+ getTitle() : String
+ setTitle(title : String) : void
+ getCategory() : String
+ setCategory(category : String) : void
+ getCost() : float
+ setCost(cost : float) : void
+ isMatch(title : String) : boolean
+ isMatch(id : int) : boolean
+ equals(o : Object) : boolean
+ toString() : String
Book
- authors : List<String>
+ Book()
+ addAuthor(AuthorName : String) : void
+ removeAuthor(AuthorName : String) : void
+ getAuthor() : List<String>
+ setAuthor(authors : List<String>) : void
+ print() : void
Disc
- length : int
- director : String
+ Disc()
+ getLength() : int
+ setLength(length : int) : void
+ getDirector() : String
+ setDirector(director : String) : void
+ play() : void
Track.
- title : String
- length : int
+ Track()
+ getTitle() : String
+ setTitle(title : String) : void
+ getLength() : int
+ setLength(length : int) : void
+ play() : void
DigitalVideoDisc
+ DigitalVideoDisc()
+ play() : void
+ print() : void
Compact Disc
- artist : String
- tracks : List<Track>
+ CompactDisc()
+ getArtist() : String
+ setArtist(artist : String) : void
+ addTrack(track : Track) : void
+ removeTrack(track : Track) : void
+ getLength() : int
+ play() : void
+ print() : void
<<interface>>
Playable
+ play() : void

Figure 51: Updated Class Diagram

13. UseCase Diagram

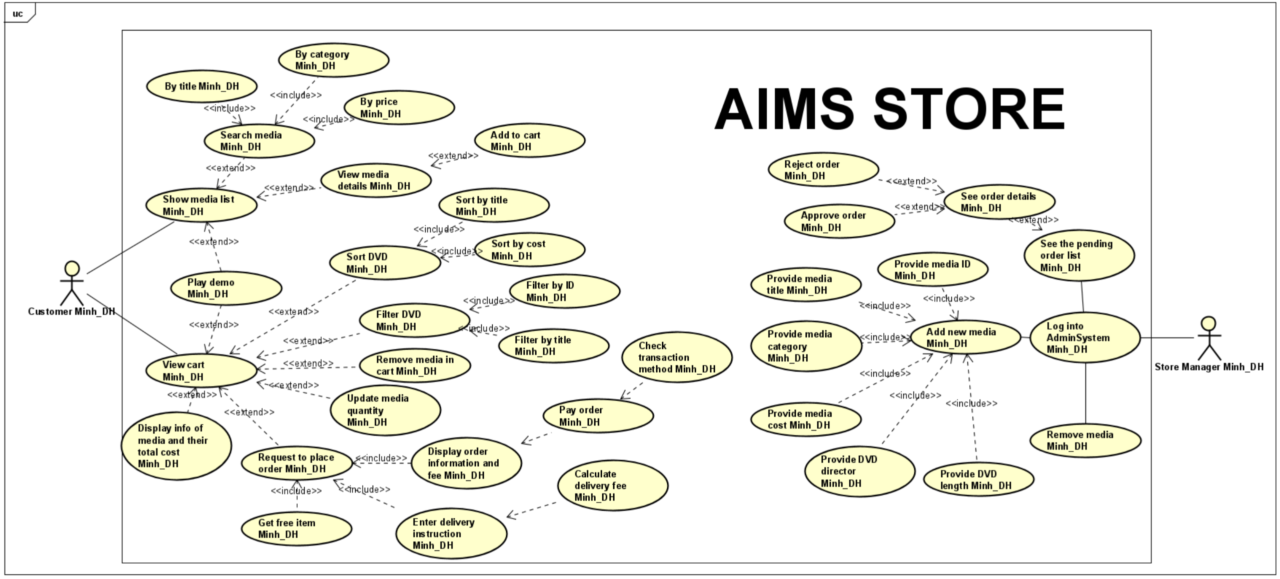

uc
AIMS STORE
By category Minh_DH
By title Minh_DH
By price Minh_DH
Search media Minh_DH
Add to cart Minh_DH
View media details Minh_DH
Show media list Minh_DH
Sort by title Minh_DH
Sort by cost Minh_DH
Sort DVD Minh_DH
Filter by ID Minh_DH
Filter DVD Minh_DH
Filter by title Minh_DH
Play demo Minh_DH
Customer Minh_DH
View cart Minh_DH
Remove media in cart Minh_DH
Update media quantity Minh_DH
Check transaction method Minh_DH
Pay order Minh_DH
Display info of media and their total cost Minh_DH
Request to place order Minh_DH
Display order information and fee Minh_DH
Calculate delivery fee Minh_DH
Get free item Minh_DH
Enter delivery instruction Minh_DH
Reject order Minh_DH
Approve order Minh_DH
See order details Minh_DH
See the pending order list Minh_DH
Provide media ID Minh_DH
Provide media title Minh_DH
Provide media category Minh_DH
Add new media Minh_DH
Log into AdminSystem Minh_DH
Store Manager Minh_DH
Provide media cost Minh_DH
Provide DVD director Minh_DH
Provide DVD length Minh_DH
Remove media Minh_DH
<<include>>
<<extend>>

Figure 52: Updated UseCase Diagram

14. Answer questions

- 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.

- 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.