ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

--- o0o --- -

BÁO CÁO THỰC HÀNH

IT3103-744527-2024.1

BÀI THỰC HÀNH – LAB04

Họ và tên: Lê Quang Khải

MSSV: 20225638

Lớp: VN03-K67

GVHD: Lê Thị Hoa

HTGD: Đặng Mạnh Cường

Contents

Table of Figures

1 Create the Book class

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

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

// Le Quang Khai 20225638
public class Book {  no usages
    private int id;  1 usage
    private String title;  1 usage
    private String category;  1 usage
    private float cost;  1 usage
    private List<String> authors = new ArrayList<String>();  7 usages

    public Book(int id, String title, String category, float cost, List<String> authors) {  no usages
        this.id = id;
        this.title = title;
        this.category = category;
        this.cost = cost;
        this.authors = authors;
    }

    public List<String> getAuthors() { return authors; }

    public void setAuthors(List<String> authors) { this.authors = authors; }
```

String title
AimsProject

Figure 1.1: Book Class 1

```java
public class Book {  no usages

    public void setAuthors(List<String> authors) { this.authors = authors; }

    // Method to add an author
    public void addAuthor(String authorName) {  no usages
        if (!authors.contains(authorName)) {
            authors.add(authorName);
            System.out.println(authorName + " added as an author.");
        } else {
            System.out.println(authorName + " is already in the list.");
        }
    }

    // Method to remove an author
    public void removeAuthor(String authorName) {  no usages
        if (authors.contains(authorName)) {
            authors.remove(authorName);
            System.out.println(authorName + " removed as an author.");
        } else {
            System.out.println(authorName + " is absent in the list.");
        }
    }
}
```

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
// Le Quang Khai 20225638
public abstract class Media {  2 usages  2 inheritors
    private int id;  5 usages
    private String title;  7 usages
    private String category;  5 usages
    private float cost;  5 usages

    public Media(String title) {  1 usage
        this.title = title;
    }

    public Media(int id, String title) {  no usages
        this.id = id;
        this.title = title;
    }

    public Media(String category, String title, float cost) {  3 usages
        this.category = category;
        this.title = title;
        this.cost = cost;
    }

    public Media(int id, String title, String category, float cost) {  1 usage
        this.id = id;
        this.title = title;
        this.category = category;
        this.cost = cost;
    }

    public int getId() {  3 usages
        return id;
    }
```

Figure 2.1: Media Class 1

```java
public abstract class Media {  2 usages  2 inheritors
    public int getId() {  1 usage
        return id;
    }

    public void setId(int id) {...}

    public String getTitle() {...}

    public void setTitle(String title) {...}

    public String getCategory() {...}

    public void setCategory(String category) {...}

    public float getCost() {...}

    public void setCost(float cost) {...}

    @Override  1 override
    public String toString() {
        return "Media{" +
                "id=" + id +
                ", title='" + title + '\'' +
                ", category='" + category + '\'' +
                ", cost=" + cost +
                '}';
    }

    public void print(){  no usages
    // lam o ham con
    }
}
```

Figure 2.2: Media Class 2

3 Creating the CompactDisc class

3.1 Create the Disc class extending the Media clas

```java
package hust.soict.hedspi.aims.media;
// Le Quang Khai 20225638
public class Disc extends Media{  1 usage

    private String director;  4 usages
    private int length;  3 usages

    public Disc(String title, String category, float cost, String director, int length) {  1 usage
        super(title, category, cost);
        this.director = director;
        this.length = length;
    }

    public Disc(String title) {  1 usage
        super(title);
    }

    public Disc(String category, String title, float cost) {  1 usage
        super(category, title, cost);
    }

    public Disc(String category, String title, float cost, String director) {  1 usage
        super(category, title, cost);
        this.director = director;
    }

    public String getDirector() { return director; }
    public void setDirector(String director) { this.director = director; }
    public int getLength() { return length; }
    public void setLength(int length) { this.length = length; }  no usages
}
```

Figure 3.1: Disc Class

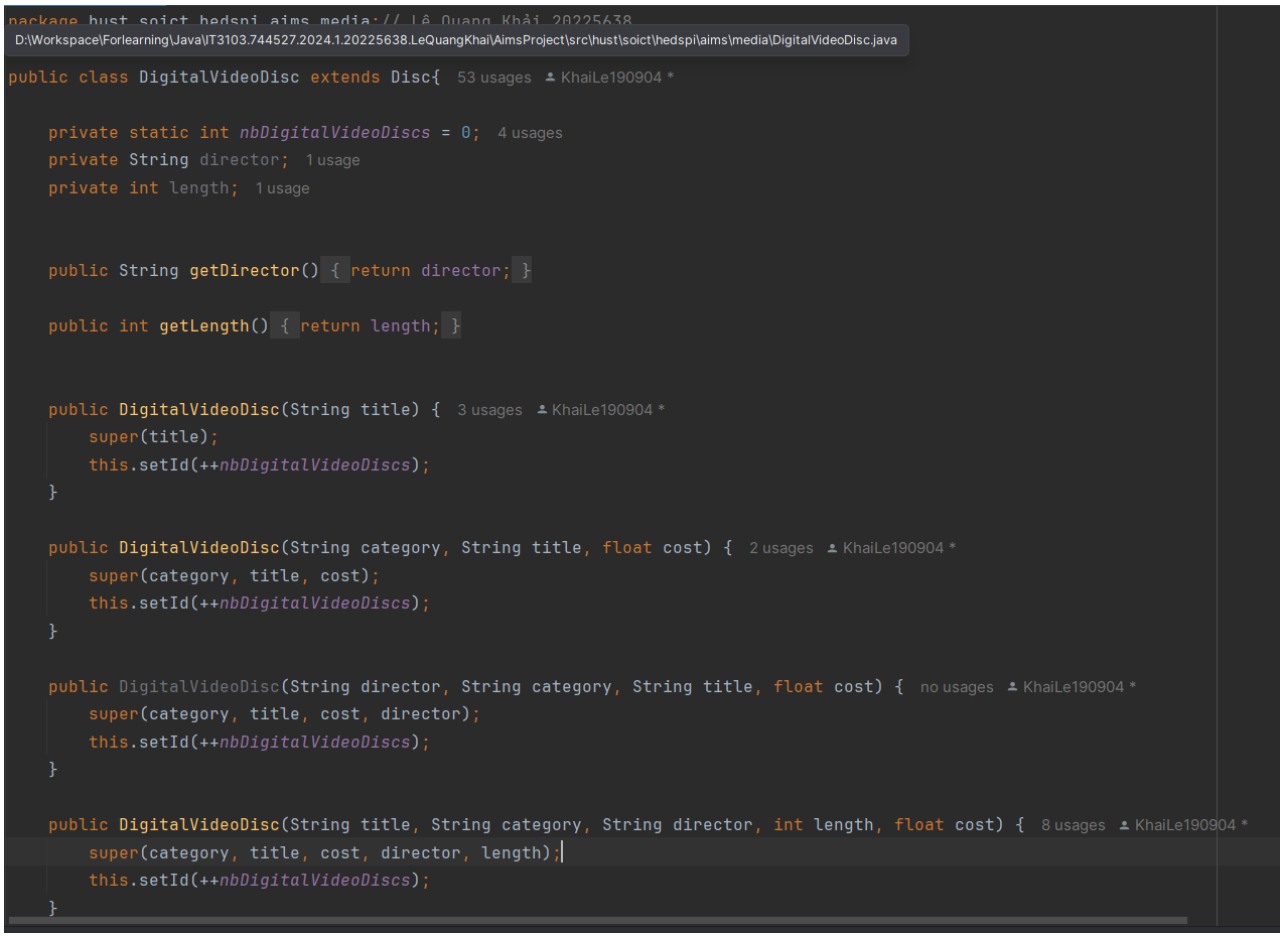


Figure 3.2: DigitalVideoDisc Class

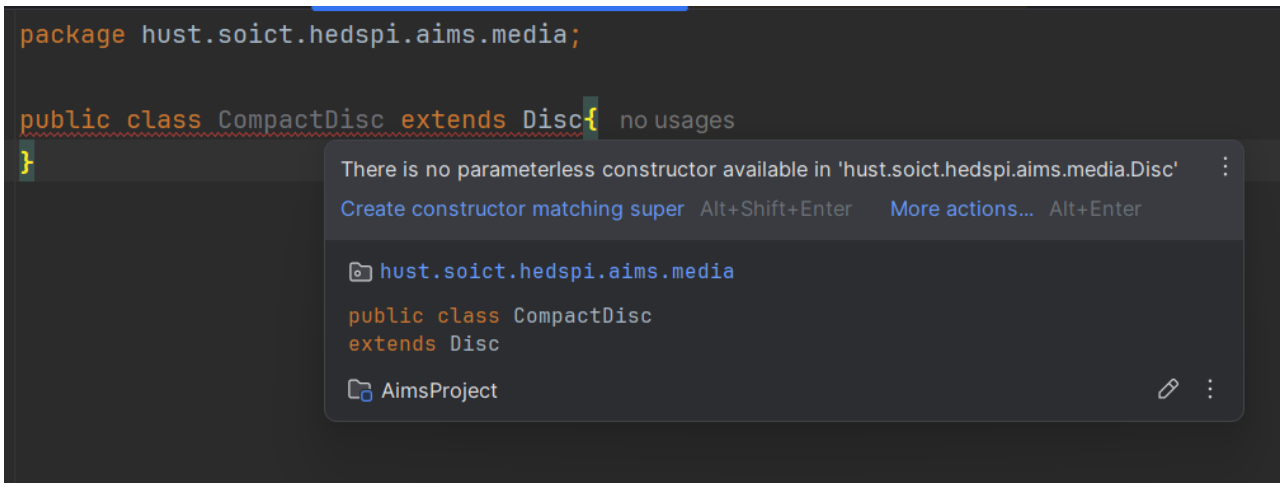


Figure 3.3: CompactDisc Class

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.hedspi.aims.media;

public class Track {  no usages
    private String title;
    private int length;

    public String getTitle() {
        return title;
    }

    public int getLength() {
        return length;
    }

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

Figure 3.4: Track Class

3.3 Open the CompactDisc class

D:\Workspace\Forlearning\Java\IT3103.744527.2024.1.20225638.LeQuangKhai\AimsProject\src\hust\soict\hedspi\aims\media\Track.java

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

import java.util.ArrayList;

public class CompactDisc extends Disc{  no usages
    private String artist;
    private ArrayList<Track> tracks;

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

    public CompactDisc(String title) {
        super(title);
    }

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

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

    public String getArtist() {
        return artist;
```

Figure 3.5: CompactDisc Class 1

```java
// Method to add a track
public void addTrack(Track track) {
    if (!tracks.contains(track)) {
        tracks.add(track);
        System.out.println(track.getTitle() + " added to the CD.");
    } else {
        System.out.println(track.getTitle() + " is already in the CD.");
    }
}

// Method to remove a track
public void removeTrack(Track track) {
    if (tracks.contains(track)) {
        tracks.remove(track);
        System.out.println(track.getTitle() + " removed from the CD.");
    } else {
        System.out.println(track.getTitle() + " is not in the CD.");
    }
}

// Method to get the total length of the CD
public int getLength() {
    int totalLength = 0;
    for (Track track : tracks) {
        totalLength += track.getLength();
    }
    return totalLength;
}
```

Figure 3.6: CompactDisc Class 2

4 Create the Playable interface

```java
package hust.soict.hedspi.aims.media;
// Le Quang Khai - 20225638
public interface Playable {
    public void play();
}
```

Figure 4.1: Playable interface

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

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

Figure 4.3: Method play() của Track

```java
@Override
public void play() {
    System.out.println("Information of Compact Disc : \n");
    System.out.println("CD artist: " + getArtist());
    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.hedspi.aims.cart;// Lê Quang Khải 20225638
// Le Quang Khai 20225638
import hust.soict.hedspi.aims.media.DigitalVideoDisc;
import hust.soict.hedspi.aims.media.Media;

import java.util.ArrayList;

public class Cart {
    public static final int MAX_NUMBERS_ORDERED = 20;
    private ArrayList<Media> itemsOrdered = new ArrayList<Media>(); // collection of Media objects

    // Method to add any type of Media to the cart
    public void addMedia(Media media) {
        if (itemsOrdered.size() < MAX_NUMBERS_ORDERED) {
            itemsOrdered.add(media);
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " has been
            System.out.println("Number of Medias in current cart: "+itemsOrdered.size());
        } else {
            System.out.println("The cart is full!!!");
        }
    }

    // Method to remove any type of Media from the cart
    public void removeMedia(Media media) {
        if (itemsOrdered.remove(media)) {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " has been
```

Figure 5.1: Cart Class 1

```java
public class Cart {                                                                    ⚠ 12
    // Method to remove any type of Media from the cart
    public void removeMedia(Media media) {
        if (itemsOrdered.remove(media)) {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " has b
        } else {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " doesn
        }
    }

    // Method to calculate the total cost of all items in the cart
    public float totalCost() {
        float cost = 0f;
        for (Media media : itemsOrdered) {
            cost += media.getCost();
        }
        return cost;
    }

    // Method to print all information about items in the cart
    public void printAllMedia() {
        System.out.println("\n**********************CART**********************");
        System.out.println("Ordered Items:");
        for (int i = 0; i < itemsOrdered.size(); i++) {
            System.out.println((i + 1) + ". " + itemsOrdered.get(i).toString());
        }
```

Figure 5.2: Cart Class 2

```java
// Method to print all information about items in the cart
public void printAllMedia() {
    System.out.println("\n**********CART**********");
    System.out.println("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("**************************************************");
}

public boolean checkInCart(int id){
    for(Media item : itemsOrdered)
        if(id==item.getId()) return true;
    return false;
}

public boolean checkInCart(String title){
    for(Media item : itemsOrdered)
        if(title.compareTo(item.getTitle())==0) return true;
    return false;
}
```

Figure 5.3: Cart Class 3

```java
// Method to search for a Media item by ID
public Media searchByID(int id) {
    for (Media media : itemsOrdered) {
        if (media.getId() == id) {
            return media;
        }
    }
    return null;
}

// Method to search for a Media item by title
public Media searchByTitle(String title) {
    for (Media media : itemsOrdered) {
        if (media.getTitle().equals(title)) {
            return media;
        }
    }
    return null;
}

public void cartEmpty(){
    for(Media item:itemsOrdered)
        removeMedia(item);
}
```

Figure 5.4: Cart Class 4

6 Update the Store class to work with Media

```java
package hust.soict.hedspi.aims.store;// Le Quang Khai 20225638
// Le Quang Khai 20225638
import hust.soict.hedspi.aims.media.DigitalVideoDisc;
import hust.soict.hedspi.aims.media.Media;

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

public class Store {  3 usages  KhaiLe190904
    public static final int MAX_CAPACITY = 1000;
    private ArrayList<Media> itemsInStore = new ArrayList<Media>(MAX_CAPACITY);

    public void addMedia(Media media) {
        if (itemsInStore.size() < MAX_CAPACITY) {
            itemsInStore.add(media);
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " has been added to store");
        } else {
            System.out.println("The store is full!!!");
        }
    }

    public void removeMedia(Media media) {
        if (itemsInStore.remove(media)) {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " has been removed from the store");
        } else {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " doesn't exist in the store");
        }
    }
    public boolean checkInStore(String title){
        for(Media item : itemsInStore){
            if (title.compareTo(item.getTitle())==0)
                return true;
        }
```

Figure 6.1: Store Class 1

```java
public class Store {  3 usages  KhaiLe190904
    public void addMedia(Media media) {
    }

    public void removeMedia(Media media) {
        if (itemsInStore.remove(media)) {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " has been removed from the store");
        } else {
            System.out.println("The " + media.getClass().getSimpleName() + " " + media.getTitle() + " doesn't exist in the store");
        }
    }
    public boolean checkInStore(String title){
        for(Media item : itemsInStore){
            if (title.compareTo(item.getTitle())==0)
                return true;
        }
        return false;
    }
    public Media searchItem(String title){

        for(Media item : itemsInStore){
            if(title.compareTo(item.getTitle())==0) return item;
        }
        return null;
    }
```

Figure 6.2: Store Class 2

7 Constructors of whole classes and parent classes

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

Figure 7.1: Constructor Track Class

```
// Constructor
public CompactDisc( String title, String category, float cost, String director, int length, String artist) {
    super(title, category, cost, director, length);
    this.artist = artist;
    this.tracks = new ArrayList<>();
}
```

Figure 7.2: Constructor CompactDisc Class

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

```
public Media(String title) {  1 usage
    this.title = title;
}

public Media(String category, String title, float cost) {  3 usages
    this.category = category;
    this.title = title;
    this.cost = cost;
}

public Media(int id, String title, String category, float cost) {  1 usage
    this.id = id;
    this.title = title;
    this.category = category;
    this.cost = cost;
}
```

Figure 7.3: Constructor Media Class

```
public Disc(String title, String category, float cost, String director, int length) {  1 usag
    super(title, category, cost);
    this.director = director;
    this.length = length;
}

public Disc(String title) {  1 usage
    super(title);
}

public Disc(String category, String title, float cost) {  1 usage
    super(category, title, cost);
}

public Disc(String category, String title, float cost, String director) {  1 usage
    super(category, title, cost);
    this.director = director;
}
```

Figure 7.4: Constructor Disc Class

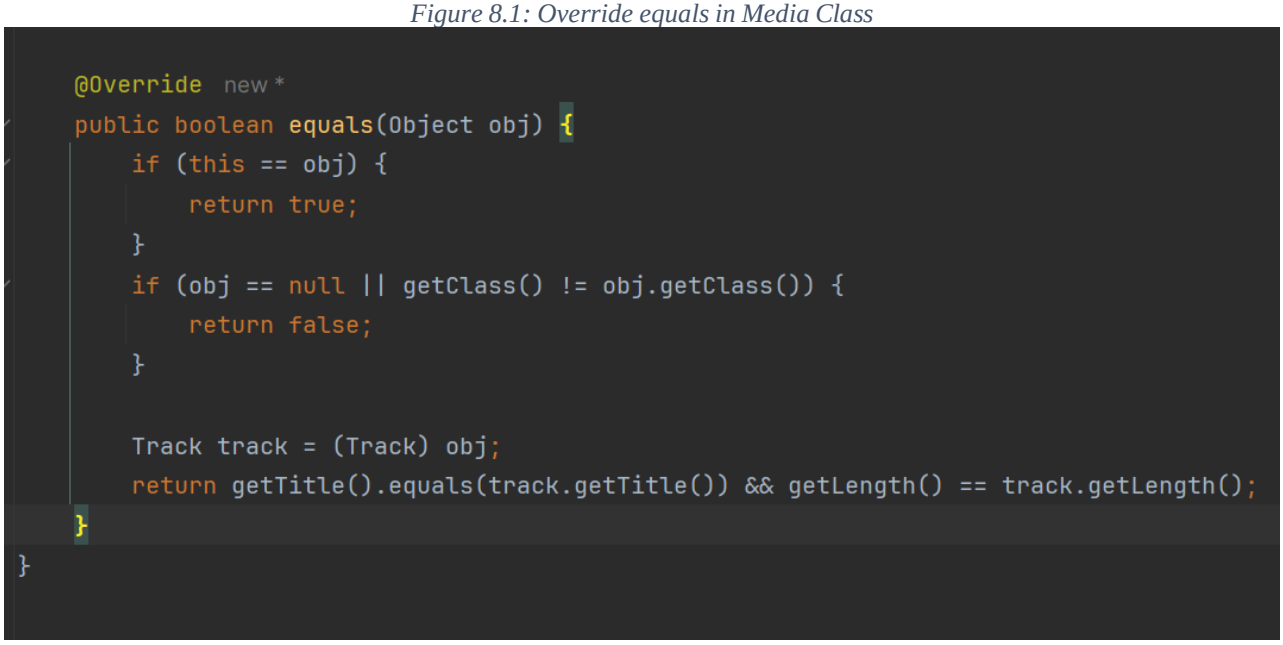
8 Unique item in a list

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.

```
    @Override  new *
    public boolean equals(Object obj) {
        if (this == obj) {
            return true;
        }
        if (obj == null || getClass() != obj.getClass()) {
            return false;
        }

        Media media = (Media) obj;
        return getTitle().equals(media.getTitle());
    }
}
```

Figure 8.1: Override equals in Media Class



```
    @Override  new *
    public boolean equals(Object obj) {
        if (this == obj) {
            return true;
        }
        if (obj == null || getClass() != obj.getClass()) {
            return false;
        }

        Track track = (Track) obj;
        return getTitle().equals(track.getTitle()) && getLength() == track.getLength();
    }
}
```

Figure 8.2: Override equals in Track Class

9 Polymorphism with toString() method

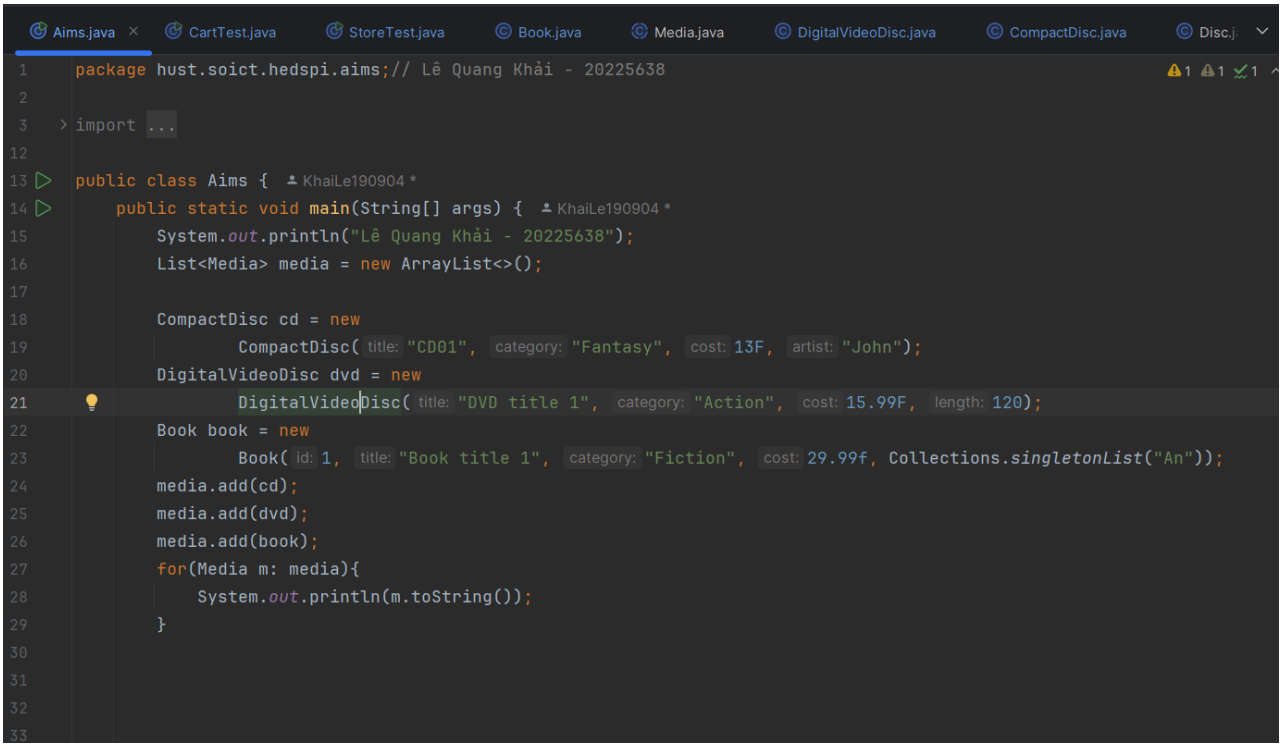


Figure 9.1: Code mô phỏng Polymorphism

Figure 9.2: Override toString() in Media Class

Kết quả

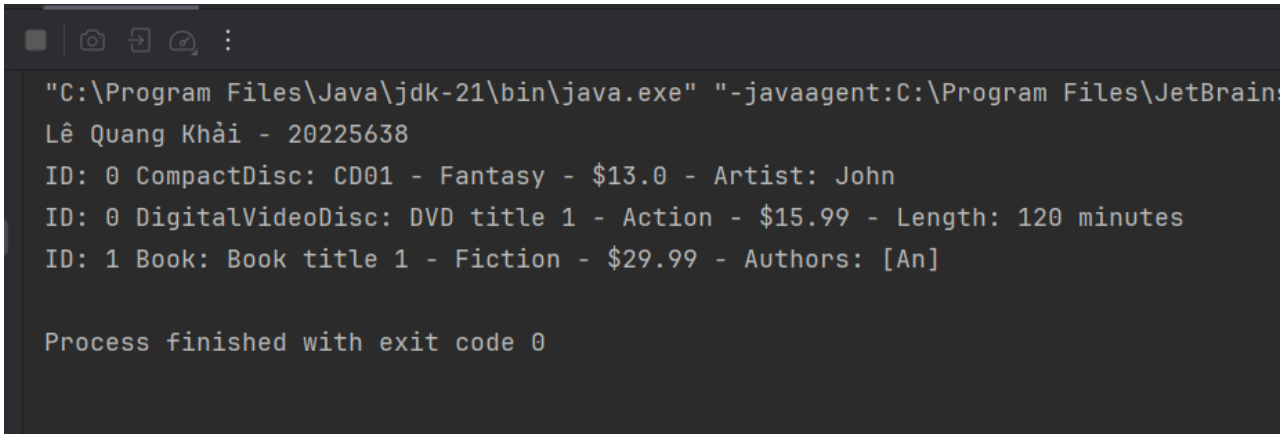


Figure 9.3: Result demo Polymorphism

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

```
// Le Quang Khai 20225638
public abstract class Media {  36 usages  4 inheritors  KhaiLe190904 *
    private int id;  4 usages
    private String title;  6 usages
    private String category;  5 usages
    private float cost;  5 usages

    public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();  no usages
    public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();  no usages
```

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

```
package hust.soict.hedspi.aims.media;
import java.util.Comparator;
// 20225638 - Le Quang Khai

public class MediaComparatorByCostTitle implements Comparator<Media> {  1 usage
    @Override
    public int compare(Media media1, Media media2) {
        // Compare by cost in descending order
        int costComparison = Float.compare(media2.getCost(), media1.getCost());

        // If costs are equal, compare by title in ascending order
        if (costComparison == 0) {
            return media1.getTitle().compareTo(media2.getTitle());
        }

        return costComparison;
    }

}
```

Figure 10.2: MediaComparatorByCostTitle Class

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

import java.util.Comparator;
// 20225638 - Le Quang Khai
public class MediaComparatorByTitleCost implements Comparator<Media>{ 1 usage

    @Override
    public int compare(Media media1, Media media2) {
        // Compare by title in ascending order
        int titleComparison = media1.getTitle().compareTo(media2.getTitle());

        // If titles are equal, compare by cost in descending order
        if (titleComparison == 0) {
            return Float.compare(media2.getCost(), media1.getCost());
        }

        return titleComparison;
    }

}
```

Figure 10.3: MediaComparatorByTitleCost Class

11 Create a complete console application in the Aims class

```
"C:\Program Files\Java\jdk-21\bin\java.
Lê Quang Khải 20225638
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

```
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
1
---------------Items available in the store---------------
Items in the store:
1. DigitalVideoDisc: DVD title 1
2. Book: Book title 1
------------------------------------------------------------
Options:
--------------------------------
1. See a medias 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

```
--------------------------------
1. See a medias 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 media's title: Book title 1
Item found.
ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
```

Figure 11.3: See a media's details

```
ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
Options:
--------------------------------
1. Add to cart
2. Play
0. Back
--------------------------------
Please choose a number: 0-1-2
1
The Book Book title 1 has been added
Number of Medias in current cart: 1
---------------Items available in the store---------------
Items in the store:
1. DigitalVideoDisc: DVD title 1
2. Book: Book title 1
------------------------------------------------------------
Options:
--------------------------------
1. See a medias details
2. Add a media to cart
```

Figure 11.4: Thêm vào Cart

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

```
Options:
--------------------------------
1. See a medias 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 media's title: DVD title 1
Item found.
The DigitalVideoDisc DVD title 1 has been added
Number of Medias in current cart: 2
--------------Items available in the store---------------
Items in the store:
1. DigitalVideoDisc: DVD title 1
2. Book: Book title 1
------------------------------------------------------------
Options:
```

Figure 11.5: Thêm media vào Cart

11.1.3 Người dùng chọn 3: Play a media

```
Options:
--------------------------------
1. See a medias 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 media's title: DVD title 1
Item found.
Playing DVD: DVD title 1
DVD length: 120
```

Figure 11.6: Play a media

11.1.4 Người dùng chọn 4: See current cart

```
Please choose a number: 0-1-2-3-4
4

**********CART**********
Ordered Items:
1. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
2. ID: 1 DigitalVideoDisc: DVD title 1 - Action - $15.99 - Length: 120 minutes
Total cost: 45.98 $
**************************************************
Options:
--------------------------------
1. Filter media in cart
2. Sort media 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 11.7: See current cart

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
Options:
--------------------------------
1. Add to store
2. Remove from store
0. Back
--------------------------------
Please choose a number: 0-1-2
```

Figure 11.8: Vào Trang Update Store

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

```
Please choose a number: 0-1-2
1
Options:
--------------------------------
1. Add CD
2. Add DVD
3. Add Book
0. Back
--------------------------------
Please choose a number: 0-1-2-3
1
Enter CD's title: CD01
Enter CD's category: Pop
Enter CD's director: Hy
Enter CD's cost: 12.3
Enter CD's length: 12
Enter CD's artist: Huy
Enter CD's number of tracks: 2
Enter track 1 name: Track01
Enter track 1 length: 12
Track01 added to the CD.
Enter track 2 name: Track02
Enter track 2 length: 13
Track02 added to the CD.
The CompactDisc CD01 has been added to store
```

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
--------------Items available in the store--------------
Items in the store:
1. DigitalVideoDisc: DVD title 1
2. Book: Book title 1
3. CompactDisc: CD01
-------------------------------------------------------
```

Figure 11.10: Result after add media to store

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

```
Options:
--------------------------------
1. Add to store
2. Remove from store
0. Back
--------------------------------
Please choose a number: 0-1-2
2
Enter media's name:
CD01
The CompactDisc CD01 has been removed from the store
```

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
--------------Items available in the store--------------
Items in the store:
1. DigitalVideoDisc: DVD title 1
2. Book: Book title 1
-------------------------------------------------------
```

Figure 11.12: Result after remove a media

11.3 Người dùng chọn 3: See current cart

```
1. See a medias 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

**********CART**********
Ordered Items:
1. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
2. ID: 1 DigitalVideoDisc: DVD title 1 - Action - $15.99 - Length: 120 minutes
Total cost: 45.98 $
***************************************************
Options:
--------------------------------
1. Filter media in cart
2. Sort media 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

11.3.1 Người dùng chọn 1: Filter medias in cart

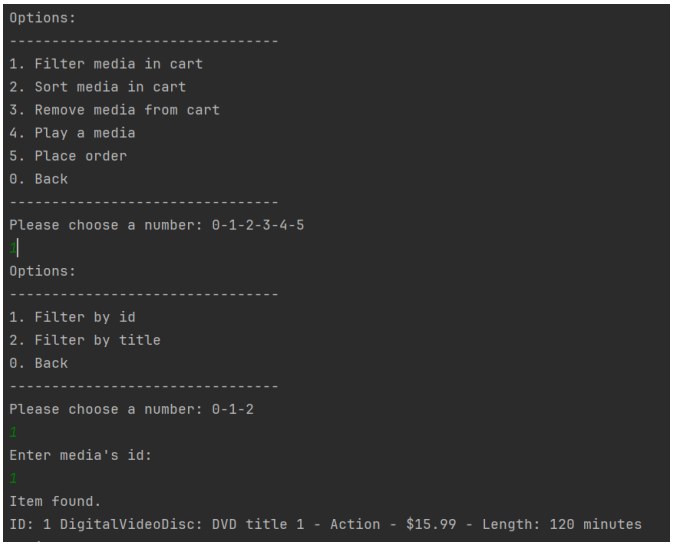


Figure 11.15: Filter Cart By id

```
--------------------------------
1. Filter by id
2. Filter by title
0. Back
--------------------------------
Please choose a number: 0-1-2
2
Enter media's name:
CD01
Item found.
ID: 4 CompactDisc: CD01 - Pop - $12.2 - Artist: Huy
```

Figure 11.16: Filter Cart By Title

11.3.2 Người dùng chọn 2: Sort medias in cart

```
--------------------------------
1. Sort by title
2. Sort by cost
0. Back
--------------------------------
Please choose a number: 0-1-2
1

**********CART**********
Ordered Items:
1. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
2. ID: 6 Book: Book title 2 - action - $16.0 - Authors: [Khai, Huy]
3. ID: 4 CompactDisc: CD01 - Pop - $12.2 - Artist: Huy
4. ID: 1 DigitalVideoDisc: DVD title 1 - Action - $15.99 - Length: 120 minutes
Total cost: 74.18 $
**************************************************
```

Figure 11.17: Sort Cart By Title

```
--------------------------------
1. Sort by title
2. Sort by cost
0. Back
--------------------------------
Please choose a number: 0-1-2
2

**********CART**********
Ordered Items:
1. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
2. ID: 6 Book: Book title 2 - action - $16.0 - Authors: [Khai, Huy]
3. ID: 1 DigitalVideoDisc: DVD title 1 - Action - $15.99 - Length: 120 minutes
4. ID: 4 CompactDisc: CD01 - Pop - $12.2 - Artist: Huy
Total cost: 74.17999 $
**************************************************
```

Figure 11.18: Sort Cart By Cost

11.3.3 Người dùng chọn 3: Remove media from cart

```
Options:
------------------------------
1. Filter media in cart
2. Sort media 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 media's name:
DVD title 1
Item found.
The DigitalVideoDisc DVD title 1 has been removed from the cart
```

Figure 11.19: Remove media by title

=> Kết quả

```
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


**********************CART**********************
Ordered Items:
1. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
2. ID: 6 Book: Book title 2 - action - $16.0 - Authors: [Khai, Huy]
3. ID: 4 CompactDisc: CD01 - Pop - $12.2 - Artist: Huy
Total cost: 58.19 $
************************************************
```

Figure 11.20: Result after remove media in cart by id

11.3.4 Người dùng chọn 4: Play a media

```
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 media's name:
CD01
Item found.
Information of Compact Disc :

CD artist: Huy
Playing DVD: Track01
DVD length: 12
Playing DVD: Track02
DVD length: 13
```

Figure 11.21: Play a media in cart

11.3.5 Người dùng chọn 5: Place order

```
**********CART**********
Ordered Items:
1. ID: 1 DigitalVideoDisc: DVD title 1 - Action - $15.99 - Length: 120 minutes
2. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
3. ID: 2 Book: Book title 1 - Fiction - $29.99 - Authors: [An]
Total cost: 75.97 $
***************************************************
Options:
-------------------------------
1. Filter media in cart
2. Sort media 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
An order is created.
Current cart is: empty

**********CART**********
Ordered Items:
Total cost: 0.0 $
```

Figure 11.22: Order

12 Class Diagram

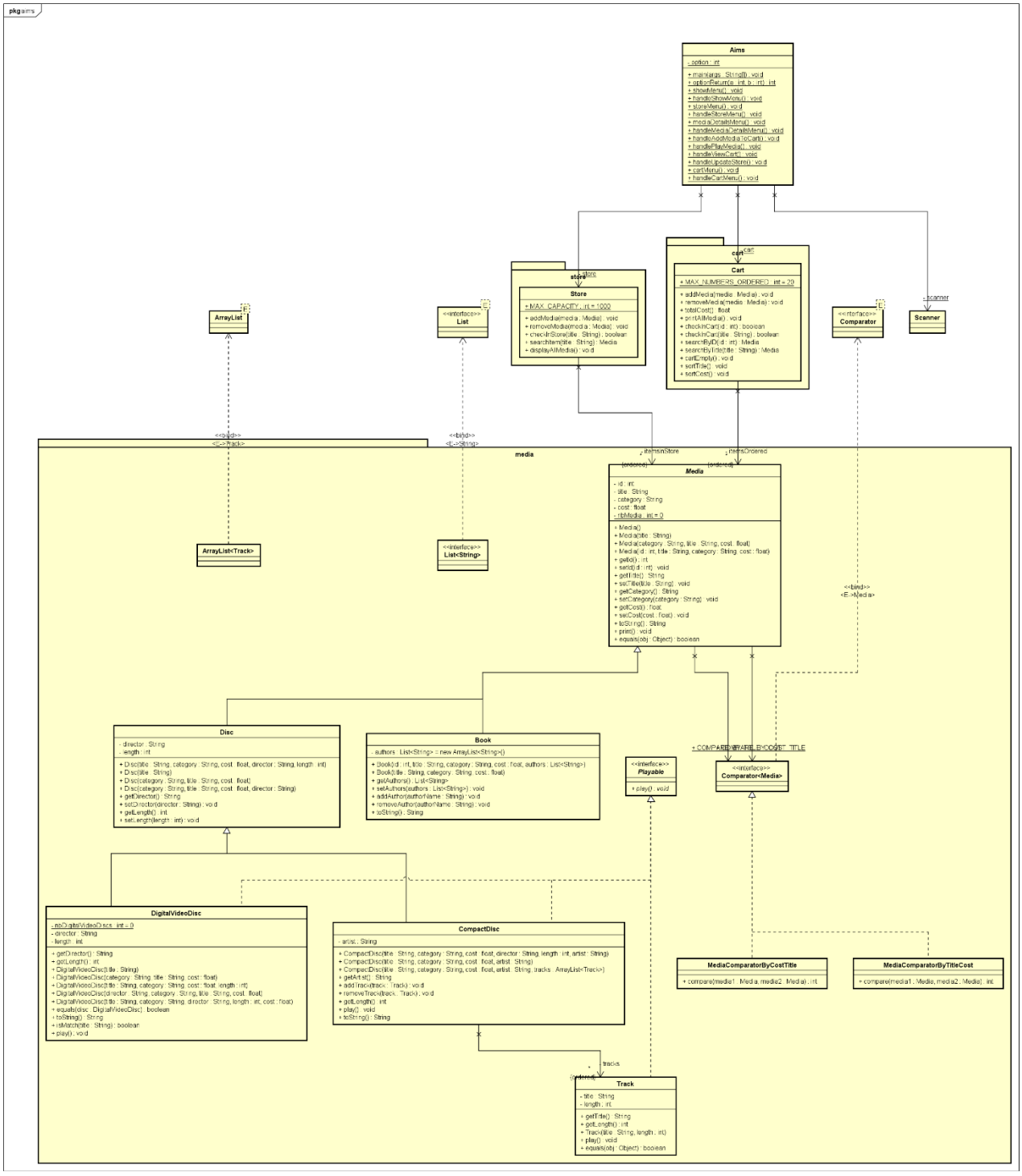


Figure 12.1: Class Diagram

13 UseCase Diagram

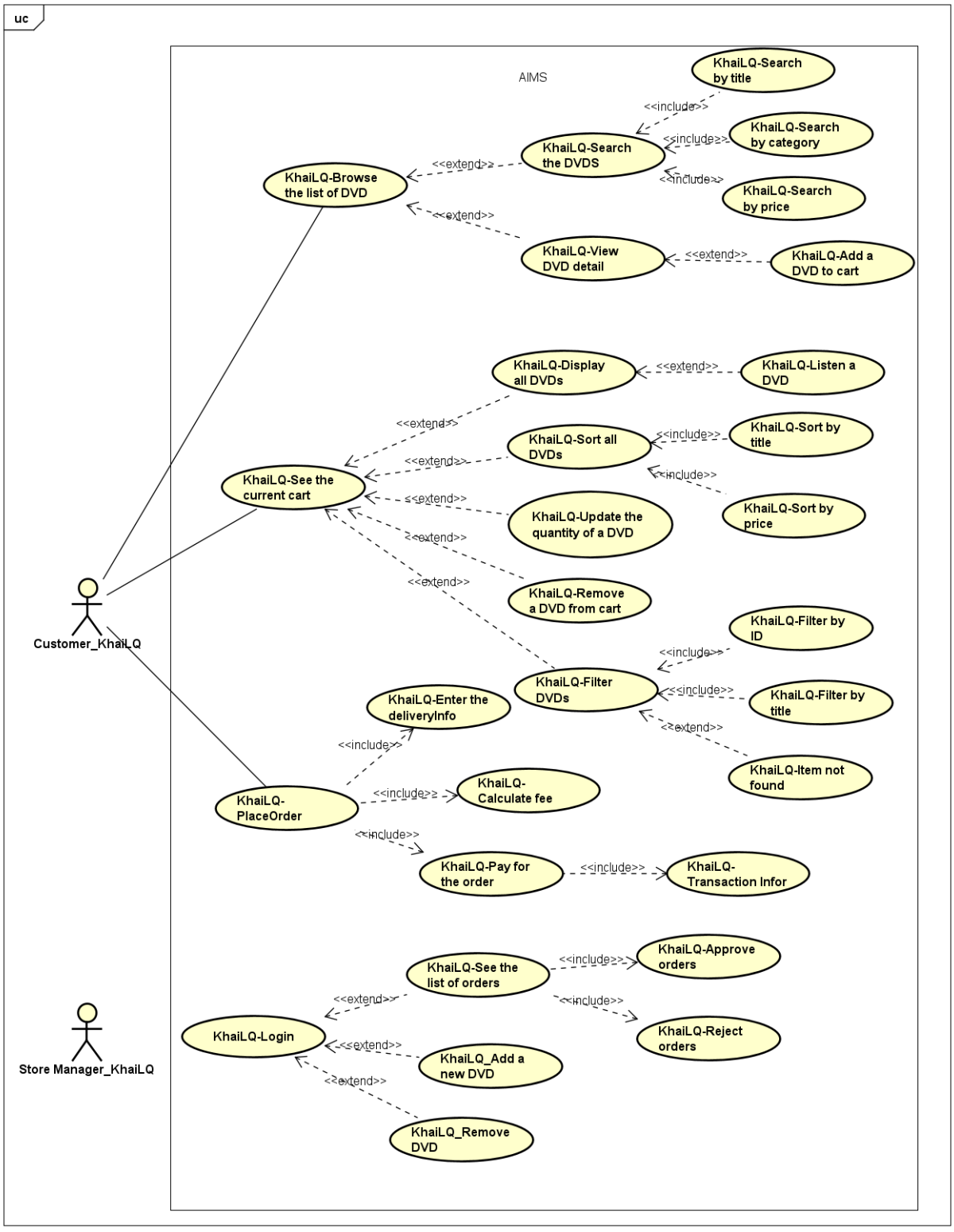


Figure 13.1: UseCase Diagram

14 Answer Questions

10. Unique Item in a List

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

Ví dụ minh họa trong lớp Aims:

```
Media book1 = new Book(5, "Book1", "Horror", 4.5f);
Media book2 = new Book(6, "Book1", "Scientific", 5.5f);
if (book1.equals(book2)) {
   System.out.println("Two objects are equal");
} else {
   System.out.println("Two objects are not equal");
}
```

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

1. Lớp nào nên triển khai giao diện Comparable?

Lớp đại diện cho các mặt hàng (ví dụ: Item, Media, hoặc lớp con cụ thể như Book, DVD) nên triển khai giao diện Comparable. Điều này phụ thuộc vào cách cấu trúc lớp của chương trình, nhưng thường lớp cơ bản (ví dụ: Media) sẽ là lớp thích hợp để triển khai.

2. Làm thế nào để triển khai phương thức compareTo() để phản ánh thứ tự mong muốn?

Trong lớp triển khai, bạn cần override phương thức compareTo() để xác định logic so sánh.
Ví dụ, nếu muốn sắp xếp theo **title** trước rồi đến **cost**, có thể viết như sau:

```
@Override
public int compareTo(Media other) {
   int titleComparison = this.title.compareTo(other.title);
   if (titleComparison != 0) {
      return titleComparison; // Sắp xếp theo title
   }
   return Double.compare(this.cost, other.cost); // Nếu title giống nhau, sắp xếp theo cost
}
```

3. Có thể có hai quy tắc sắp xếp (theo title rồi cost **và theo** cost rồi title**) nếu dùng Comparable không?**

Không, giao diện Comparable chỉ cho phép định nghĩa **một quy tắc sắp xếp duy nhất** thông qua phương thức compareTo().
Nếu cần nhiều quy tắc sắp xếp, bạn nên sử dụng Comparator thay vì Comparable.

4. Nếu DVD có quy tắc sắp xếp khác (theo title**, rồi đến** độ dài giảm dần**, rồi đến** cost**), làm thế nào để sửa đổi mã?**

Để xử lý quy tắc sắp xếp riêng cho DVD, bạn cần override phương thức compareTo() trong lớp DVD và triển khai logic so sánh riêng:

```
@Override
public int compareTo(Media other) {
    if (other instanceof DVD) {
        DVD otherDVD = (DVD) other;
        int titleComparison = this.title.compareTo(otherDVD.title);
        if (titleComparison != 0) {
            return titleComparison; // Sắp xếp theo title
        }
        int lengthComparison = Integer.compare(otherDVD.length, this.length);
        if (lengthComparison != 0) {
            return lengthComparison; // Sắp xếp theo độ dài giảm dần
        }
        return Double.compare(this.cost, otherDVD.cost); // Nếu cả title và length
giống nhau, sắp xếp theo cost
    }
    // Nếu không phải DVD, xử lý theo logic mặc định
    return super.compareTo(other);
}
```