

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 04

Họ và tên sv: Hồ Tuấn Anh

Lớp: K67-ITEP01

GVHD: Lê Thị Hoa

TA: Đặng Mạnh Cường

Hà Nội 12/2024

BÁO CÁO THỰC HÀNH LAB 4 LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Contents

1	Create the Book class	4
2	Creating the abstract Media class.....	6
3	Creating the CompactDisc class	8
3.1	Create the Disc class extending the Media clas.....	8
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.....	10
3.3	Open the CompactDisc class	11
4	Create the Playable interface	13
5	Update the Cart class to work with Media	14
6	Update the Store class to work with Media	18
7	Constructors of whole classes and parent classes	20
8	Unique item in a list	21
9	Polymorphism with <code>toString()</code> method.....	22
10	Sort media in the car	23
11	Create a complete console application in the Aims class	24
11.1	Người dùng chọn 1: View store	25
11.1.1	Người dùng tiếp tục chọn 1. See a media's details	26
11.1.2	Người dùng chọn 2: Add a media to the cart	27
11.1.3	Người dùng chọn 3: Play a media	27
11.1.4	Người dùng chọn 4: See current cart	28
11.2	Người dùng chọn 2: Update store	29
11.2.1	Người dùng chọn 1: Add a media to the store	29
11.2.2	Người dùng chọn 2: Remove a media from the store	30
11.3	Người dùng chọn 3: See current cart	31
11.3.1	Người dùng chọn 1: Filter medias in cart	32
11.3.2	Người dùng chọn 2: Sort medias in cart	34
11.3.3	Người dùng chọn 3: Remove media from cart	35
11.3.4	Người dùng chọn 4: Play a media	36
11.3.5	Người dùng chọn 5: Place order	36

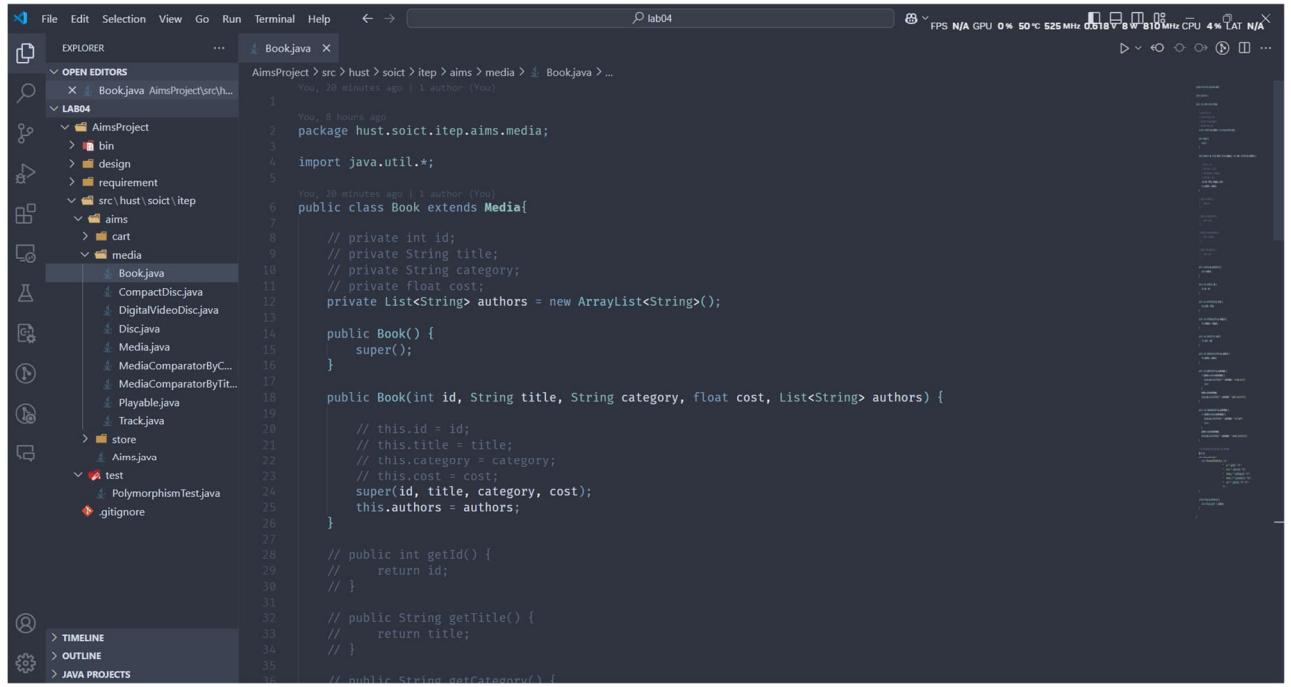
12	Class Diagram.....	38
13	UseCase Diagram	39
14	Answer Questions	39

Table of Figures

Figure 1.1: Book Class 1	4
Figure 1.2: Book Class 2.....	5
Figure 2.1: Media Class 1	6
Figure 2.2: Media Class 2	7
Figure 3.1: Disc Class.....	8
Figure 3.2: DigitalVideoDisc Class.....	9
Figure 3.3: CompactDisc Class.....	9
Figure 3.4: Track Class.....	10
Figure 3.5: CompactDisc Class 1	11
Figure 3.6: CompactDisc Class 2	12
Figure 4.1: Playable interface.....	13
Figure 4.2: Method play() của DigitalVideoDisc.....	13
Figure 4.3: Method play() của Track	13
Figure 4.4: Method play() của CompactDisc.....	13
Figure 5.1: Cart Class 1.....	14
Figure 5.2: Cart Class 2.....	15
Figure 5.3: Cart Class 3.....	16
Figure 5.4: Cart Class 4.....	17
Figure 6.1: Store Class 1	18
Figure 6.2: Store Class 2	19
Figure 7.1: Constructor Track Class	20
Figure 7.2: Constructor CompactDisc Class.....	20
Figure 7.3: Constructor Media Class.....	20
Figure 7.4: Constructor Disc Class.....	21
Figure 8.1: Override equals in Media Class	21
Figure 8.2: Override equals in Track Class.....	21
Figure 9.1: Code mô phỏng Polymorphism	22
Figure 9.2: Override toString() in Media Class.....	22
Figure 9.3: Result demo Polymorphism	22
Figure 10.1: Add the comparators as attributes of the Media class.....	23
Figure 10.2: MediaComparatorByCostTitle Class	23
Figure 10.3: MediaComparatorByTitleCost Class	24
Figure 11.1: Màn hình chính.....	24
Figure 11.2: Vào Trang View Store.....	25
Figure 11.3: See a media's details.....	26
Figure 11.4: Thêm vào Cart	26
Figure 11.5: Thêm media vào Cart.....	27

Figure 11.6: Play a media	27
Figure 11.7: See current cart after sort.....	28
Figure 11.8: Vào Trang Update Store	29
Figure 11.9: Add a media to store	29
Figure 11.10: Result after add media to store	30
Figure 11.11: Remove a media from the store	30
Figure 11.12: Result after remove a media	31
Figure 11.13: Vào trang See current cart	31
Figure 11.14: Media in Cart	32
Figure 11.15: Filter Cart By id	32
Figure 11.16: Filter Cart By Title	33
Figure 11.17: Sort Cart By Title.....	34
Figure 11.18: Sort Cart By Cost	34
Figure 11.19: Remove media by id	35
Figure 11.20: Result after remove media in cart by id	35
Figure 11.21: Play a media in cart.....	36
Figure 11.22: Order	36
Figure 11.23: Result after order	37
Figure 12.1: Class Diagram	38
Figure 13.1: UseCase Diagram.....	39
Figure 14.1: Triển khai Comparable trong lớp trừu tượng Media	40
Figure 14.2: Mở rộng để so sánh nhiều thuộc tính hơn	40
Figure 14.3: Triển khai tại lớp con	40

1 Create the Book class



The screenshot shows a Java code editor interface with the following details:

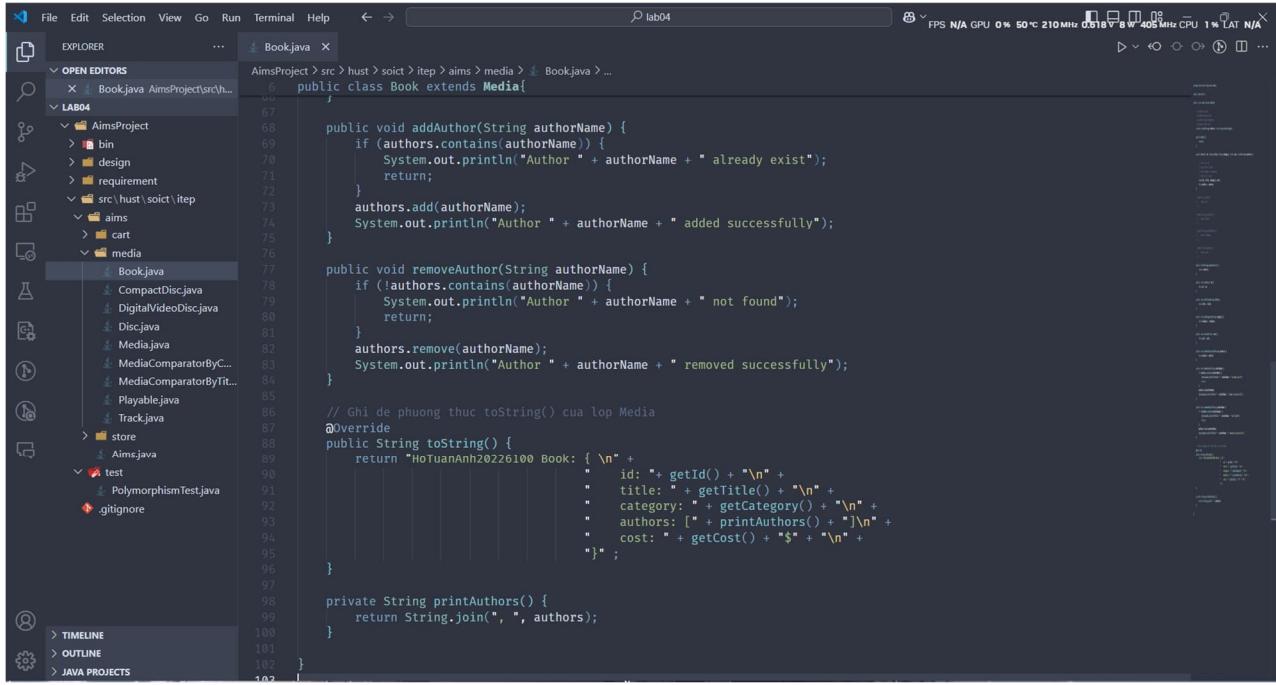
- Project Structure:** The left sidebar shows a project named "AimsProject" with a "src" folder containing "LAB04". "LAB04" contains "AimsProject", "bin", "design", "requirement", "src", and "media" folders. "src" contains "Book.java", "CompactDisc.java", "DigitalVideoDisc.java", "Disc.java", "Media.java", "MediaComparatorByC...", "MediaComparatorByTit...", "Playable.java", "Track.java", "store", "Aims.java", and "test" folder with "PolymorphismTest.java" and ".gitignore".
- Code Editor:** The main area displays the "Book.java" file content.
- Code Content:**

```

1 package hust.soict.itep.media;
2
3 import java.util.*;
4
5 public class Book extends Media{
6
7     // private int id;
8     // private String title;
9     // private String category;
10    // private float cost;
11    private List<String> authors = new ArrayList<String>();
12
13    public Book() {
14        super();
15    }
16
17    public Book(int id, String title, String category, float cost, List<String> authors) {
18
19        // this.id = id;
20        // this.title = title;
21        // this.category = category;
22        // this.cost = cost;
23        super(id, title, category, cost);
24        this.authors = authors;
25    }
26
27    // public int getId() {
28    //     return id;
29    // }
30
31    // public String getTitle() {
32    //     return title;
33    // }
34
35    // public String getCategory() {
36    // }
37}

```
- Terminal:** A terminal window at the top right shows system information: FPS N/A GPU 0% 50°C 525 MHz 0x18 V B W 810 MHz CPU 4% LAT N/A.

Figure 1.1: Book Class 1



```

File Edit Selection View Go Run Terminal Help < > lab04
OPEN EDITORS Book.java ...
EXPLORER AimsProject hust soict itep aims media ...
LAB04
AimsProject
> bin
> design
> requirement
src\hust\soict\itep
> aims
> cart
media
Book.java
public class Book extends Media {
    ...
    public void addAuthor(String authorName) {
        if (authors.contains(authorName)) {
            System.out.println("Author " + authorName + " already exist");
            return;
        }
        authors.add(authorName);
        System.out.println("Author " + authorName + " added successfully");
    }
    ...
    public void removeAuthor(String authorName) {
        if (!authors.contains(authorName)) {
            System.out.println("Author " + authorName + " not found");
            return;
        }
        authors.remove(authorName);
        System.out.println("Author " + authorName + " removed successfully");
    }
    ...
    // Ghi đè phương thức toString() của lớp Media
    @Override
    public String toString() {
        return "HoTuanAnh20226100 Book: { \n" +
               "    id: " + getId() + "\n" +
               "    title: " + getTitle() + "\n" +
               "    category: " + getCategory() + "\n" +
               "    authors: [" + printAuthors() + "]\n" +
               "    cost: " + getCost() + "$" + "\n" +
               "}";
    }
    ...
    private String printAuthors() {
        return String.join(", ", authors);
    }
}

```

Figure 1.2: Book Class 2

2 Creating the abstract Media class

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

```

File Edit Selection View Go Run Terminal Help < > lab04
EXPLORER AimsProject src hust soict itep aims media Media.java
LAB04 AimsProject bin design requirement src\hust\soict\itep aims cart media Book.java CompactDisc.java DigitalVideoDisc.java Disc.java Media.java MediaComparatorBy... MediaComparatorByTit... Playable.java Track.java store Aims.java test PolymorphismTest.java .gitignore
Media.java
1 package hust.soict.itep.aims.media;
2
3 import java.util.Comparator;
4
5 public abstract class Media {
6     // private int id;
7     // private String title;
8     // private String category;
9     // private float cost;
10
11     protected int id; You, 8 hours ago | 1 author hust.soict.itep.aims.media.Media
12     protected String title; You, 8 hours ago * Merge lab04 from release/lab04 into master
13     protected String category;
14     protected float cost;
15
16     public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();
17     public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();
18
19
20     public Media() {}
21
22     public Media(int id, String title, String category, float cost) {
23         this.id = id;
24         this.title = title;
25         this.category = category;
26         this.cost = cost;
27     }
28
29     public int getId() {
30         return id;
31     }
32
33     public void setId(int id) {
34         this.id = id;
35     }
36

```

Figure 2.1: Media Class 1

```

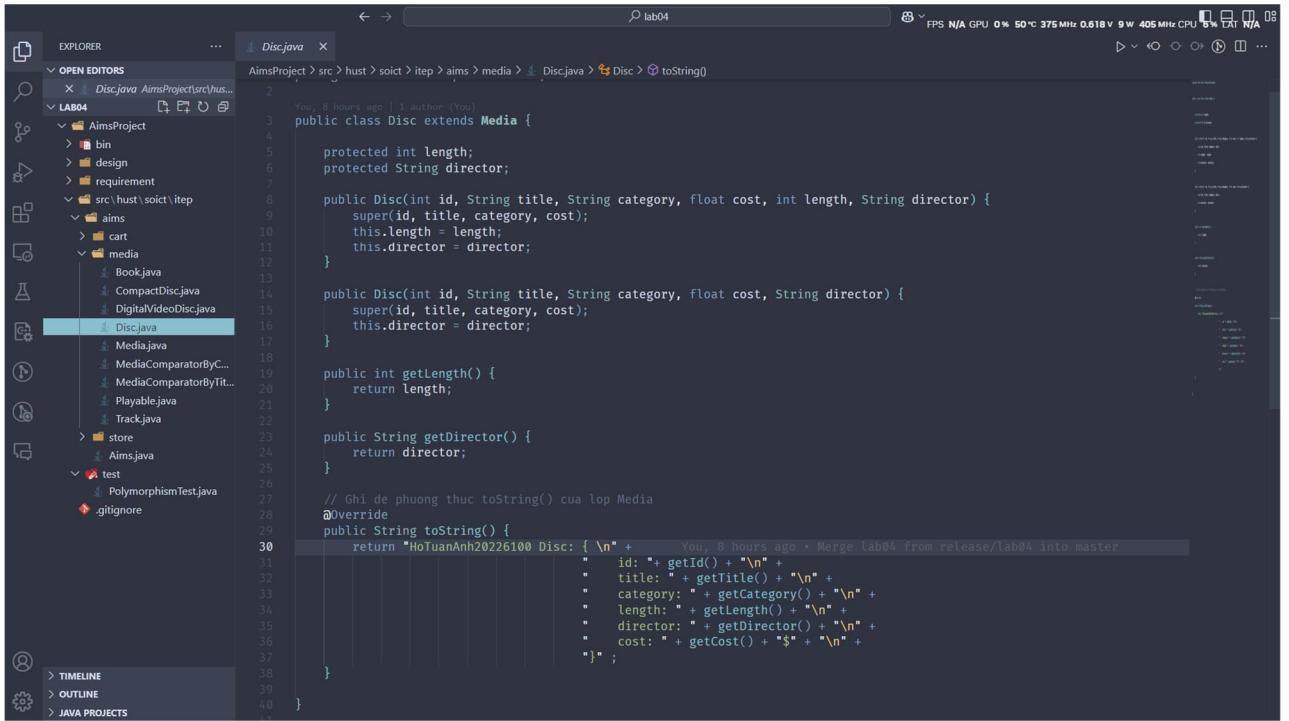
AimsProject > src > hust > soict > itep > aims > media > Media.java > Media.java > Id
6   public abstract class Media {
35     public void setId(int id) {
36       this.id = id;
37     }
38
39     public String getTitle() {
40       return title;
41     }
42
43     public void setTitle(String title) {
44       this.title = title;
45     }
46
47     public String getCategory() {
48       return category;
49     }
50
51     public void setCategory(String category) {
52       this.category = category;
53     }
54
55     public float getCost() {
56       return cost;
57     }
58
59     public void setCost(float cost) {
60       this.cost = cost;
61     }
62
63
64     // Ghi đè phương thức equals() mặc định
65     @Override
66     public boolean equals(Object o) {
67       Media media = (Media) o;
68       return media.getTitle().equals(this.getTitle());
69     }
70
71     @Override
72     public abstract String toString();
73
74

```

Figure 2.2: Media Class 2

3 Creating the CompactDisc class

3.1 Create the Disc class extending the Media clas

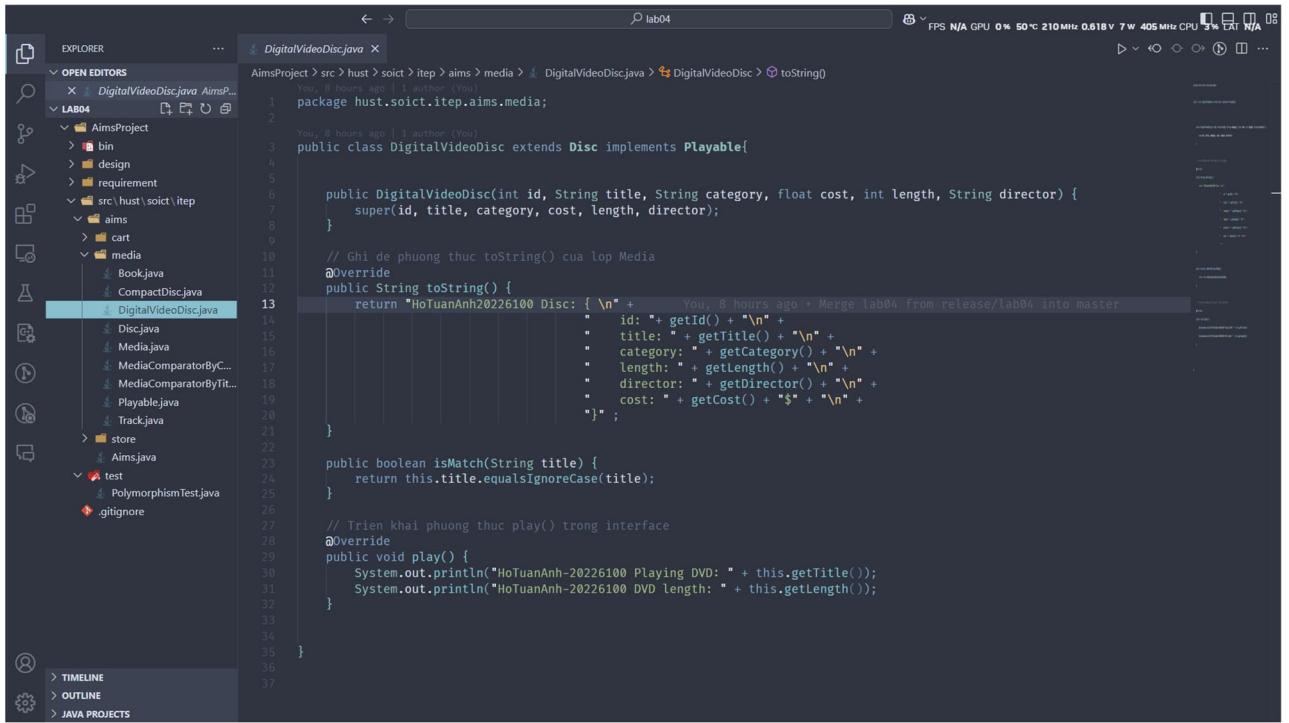


```

AimsProject > src > hust > soict > itep > aims > media > Disc.java > Disc > toString()
You, 8 hours ago | 1 author (You)
public class Disc extends Media {
    protected int length;
    protected String director;
    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;
    }
    public Disc(int id, String title, String category, float cost, String director) {
        super(id, title, category, cost);
        this.director = director;
    }
    public int getLength() {
        return length;
    }
    public String getDirector() {
        return director;
    }
    // Ghi đè phương thức toString() của lớp Media
    @Override
    public String toString() {
        return "HoTuAnh20226100 Disc: [ \n" +
            "    id: " + getId() + "\n" +
            "    title: " + getTitle() + "\n" +
            "    category: " + getCategory() + "\n" +
            "    length: " + getLength() + "\n" +
            "    director: " + getDirector() + "\n" +
            "    cost: " + getCost() + "$" + "\n" +
            "];";
    }
}

```

Figure 3.1: Disc Class

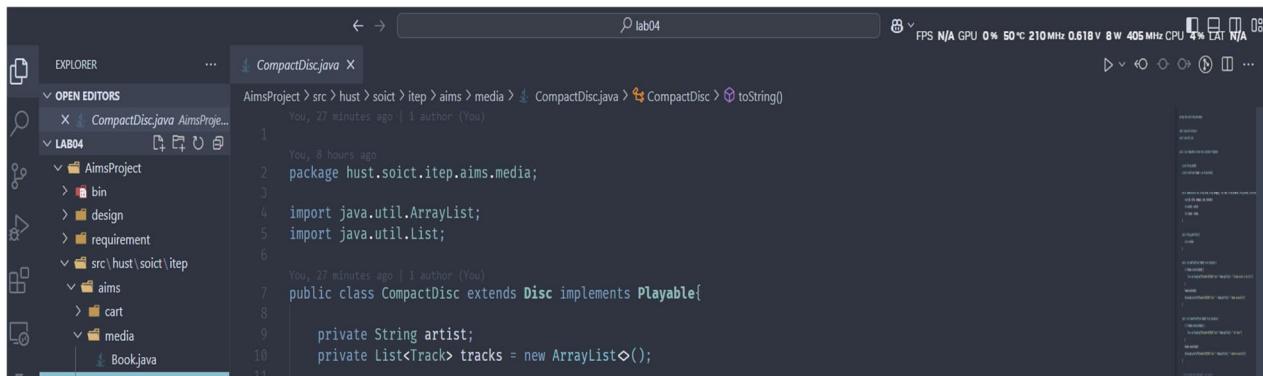


```

DigitalVideoDisc.java
1 package hust.soict.itep.aims.media;
2
3 public class DigitalVideoDisc extends Disc implements Playable{
4
5     public DigitalVideoDisc(int id, String title, String category, float cost, int length, String director) {
6         super(id, title, category, cost, length, director);
7     }
8
9     // Ghi đè phương thức toString() của lớp Media
10    @Override
11    public String toString() {
12        return "HoTuanAnh20226100 Disc: { \n" +
13            "    id: " + getId() + "\n" +
14            "    title: " + getTitle() + "\n" +
15            "    category: " + getCategory() + "\n" +
16            "    length: " + getLength() + "\n" +
17            "    director: " + getDirector() + "\n" +
18            "    cost: " + getCost() + "$" + "\n" +
19            "}";
20    }
21
22    public boolean isMatch(String title) {
23        return this.title.equalsIgnoreCase(title);
24    }
25
26    // Triển khai phương thức play() trong interface
27    @Override
28    public void play() {
29        System.out.println("HoTuanAnh-20226100 Playing DVD: " + this.getTitle());
30        System.out.println("HoTuanAnh-20226100 DVD length: " + this.getLength());
31    }
32
33}
34
35}
36
37}

```

Figure 3.2: DigitalVideoDisc Class



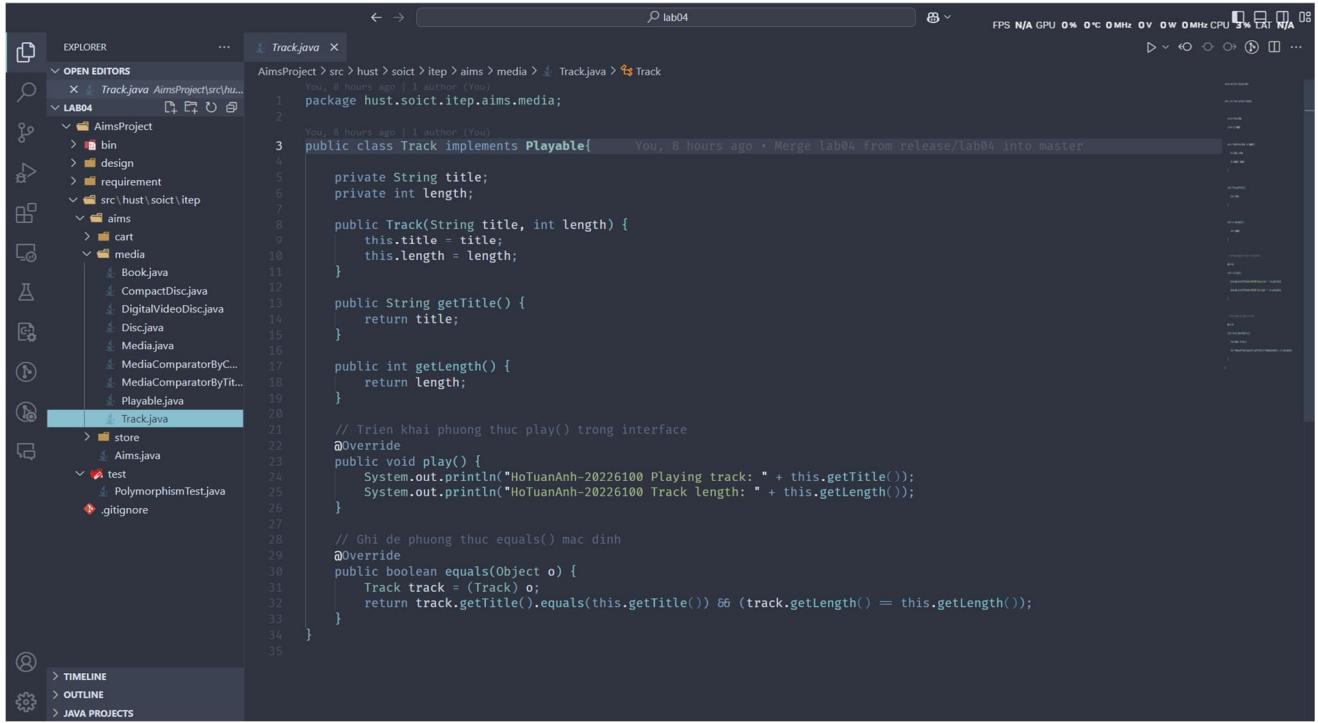
```

CompactDisc.java
1 package hust.soict.itep.aims.media;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class CompactDisc extends Disc implements Playable{
7
8     private String artist;
9     private List<Track> tracks = new ArrayList();
10
11    public CompactDisc(String artist) {
12        this.artist = artist;
13    }
14
15    @Override
16    public void play() {
17        System.out.println("HoTuanAnh-20226100 Playing CD: " + this.getArtist());
18        System.out.println("HoTuanAnh-20226100 CD tracks: " + this.getTracks());
19    }
20
21    @Override
22    public String toString() {
23        return "HoTuanAnh20226100 CD: { \n" +
24            "    artist: " + getArtist() + "\n" +
25            "    tracks: " + getTracks() +
26            "}";
27    }
28
29    public String getArtist() {
30        return artist;
31    }
32
33    public void setArtist(String artist) {
34        this.artist = artist;
35    }
36
37    public List<Track> getTracks() {
38        return tracks;
39    }
40
41    public void setTracks(List<Track> tracks) {
42        this.tracks = tracks;
43    }
44}

```

Figure 3.3: CompactDisc Class

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



```

package hust.soict.itep.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 int getLength() {
        return length;
    }

    // Triển khai phương thức play() trong interface
    @Override
    public void play() {
        System.out.println("HoTuanAnh-20226100 Playing track: " + this.getTitle());
        System.out.println("HoTuanAnh-20226100 Track length: " + this.getLength());
    }

    // Ghi đè phương thức equals() mặc định
    @Override
    public boolean equals(Object o) {
        Track track = (Track) o;
        return track.getTitle().equals(this.getTitle()) && (track.getLength() == this.getLength());
    }
}

```

Figure 3.4: Track Class

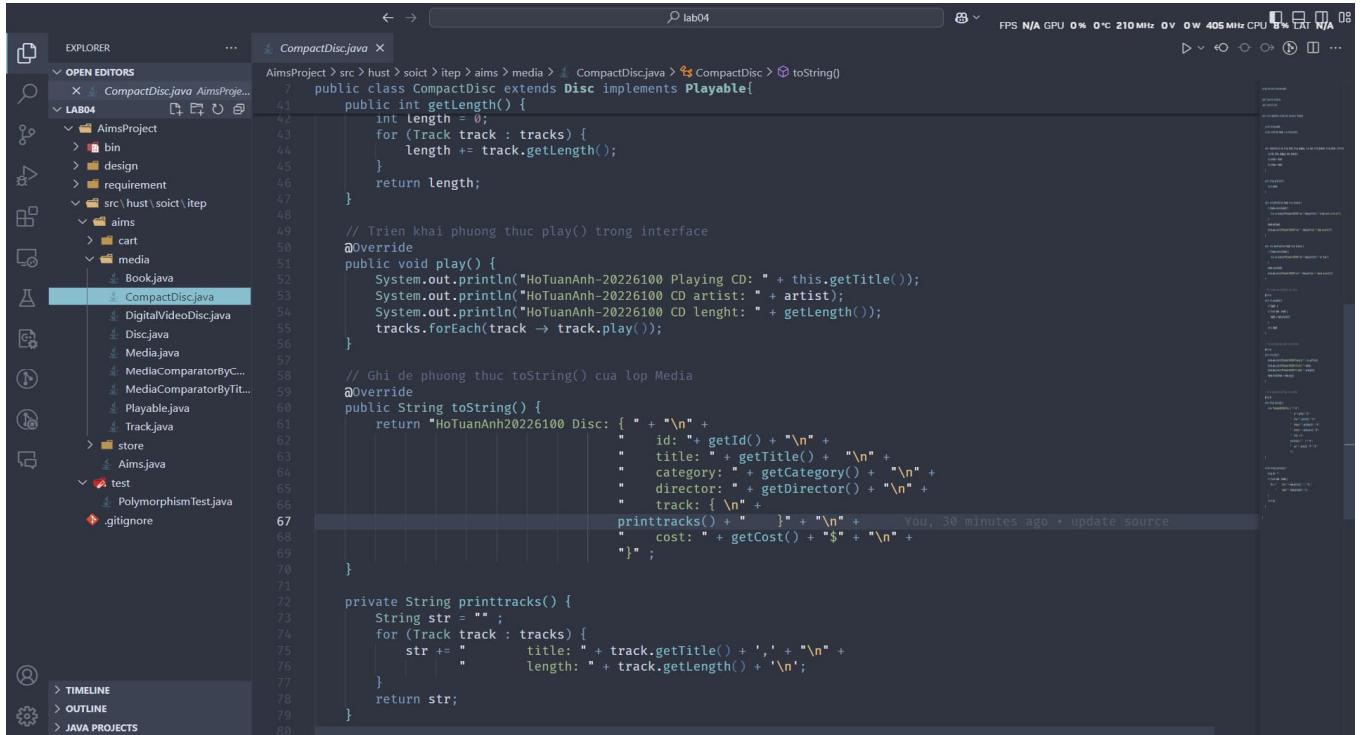
3.3 Open the CompactDisc class

```

CompactDisc.java ×
AimsProject > src > hust > soict > itep > aims > media > CompactDisc.java > CompactDisc > toString()
You 30 minutes ago | 1 author (You)
2 package hust.soict.itep.aims.media;
3
4 import java.util.ArrayList;
5 import java.util.List;
6
7 public class CompactDisc extends Disc implements Comparable<CompactDisc> {
8
9     private String artist;
10    private List<Track> tracks = new ArrayList<Track>();
11
12
13    public CompactDisc(int id, String title, String category, float cost, String director, String artist, List<Track> tracks) {
14        super(id, title, category, cost, director);
15        this.artist = artist;
16        this.tracks = tracks;
17    }
18
19    public String getArtist() {
20        return artist;
21    }
22
23    public void addTrack(Track track) throws Exception {
24        if (tracks.contains(track)) {
25            throw new Exception("HoTuanAnh-20226100 Track " + track.getTitle() + " already exists in the list.");
26        }
27        tracks.add(track);
28        System.out.println("HoTuanAnh-20226100 Track " + track.getTitle() + " added successfully");
29    }
30
31    public void removeTrack(Track track) throws Exception {
32        if (!tracks.contains(track)) {
33            throw new Exception("HoTuanAnh-20226100 Track " + track.getTitle() + " not found.");
34        }
35        tracks.remove(track);
36        System.out.println("HoTuanAnh-20226100 Track " + track.getTitle() + " removed successfully");
37    }
38
39    // Ghi đè phương thức getLength() của lớp Disc
40    @Override
41    public int compareTo(CompactDisc o) {
42        return this.title.compareTo(o.title);
43    }
44}

```

Figure 3.5: CompactDisc Class 1



```

AimsProject > src > hust > soict > itep > aims > media > CompactDisc.java > CompactDisc.java > toString()
  7  public class CompactDisc extends Disc implements Playable{
  8      public int getLength() {
  9          int length = 0;
 10         for (Track track : tracks) {
 11             length += track.getLength();
 12         }
 13         return length;
 14     }
 15
 16     // Triển khai phương thức play() trong interface
 17     @Override
 18     public void play() {
 19         System.out.println("HoTuanAnh-20226100 Playing CD: " + this.getTitle());
 20         System.out.println("HoTuanAnh-20226100 CD artist: " + artist);
 21         System.out.println("HoTuanAnh-20226100 CD lenght: " + getLength());
 22         tracks.forEach(track -> track.play());
 23     }
 24
 25     // Ghi đè phương thức toString() của lớp Media
 26     @Override
 27     public String toString() {
 28         return "HoTuanAnh20226100 Disc: { " + "\n" +
 29                 " id: " + getId() + "\n" +
 30                 " title: " + getTitle() + "\n" +
 31                 " category: " + getCategory() + "\n" +
 32                 " director: " + getDirector() + "\n" +
 33                 " track: { \n" +
 34                     printtracks() + " }" + "\n" +
 35                     " You, 30 minutes ago * update source
 36                     " cost: " + getCost() + "$" + "\n" +
 37                     " }" ;
 38     }
 39
 40     private String printtracks() {
 41         String str = "";
 42         for (Track track : tracks) {
 43             str += " title: " + track.getTitle() + ',' + "\n" +
 44                   " length: " + track.getLength() + '\n';
 45         }
 46         return str;
 47     }
 48 }

```

Figure 3.6: CompactDisc Class 2

4 Create the Playable interface

```
AimsProject > src > hust > soict > itep > aims > media > Playable.java > ...
You, 8 hours ago | 1 author (You)
1 package hust.soict.itep.aims.media;
2
3 You, 8 hours ago | 1 author (You)
3 ✓ public interface Playable {
4
5     void play();
6
7 }
```

Figure 4.1: Playable interface

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

```
// Trien khai phuong thuc play() trong interface
@Override
public void play() {
    System.out.println("HoTuanAnh-20226100 Playing DVD: " + this.getTitle());
    System.out.println("HoTuanAnh-20226100 DVD length: " + this.getLength());
}
```

Figure 4.2: Method play() của DigitalVideoDisc

```
// Trien khai phuong thuc play() trong interface
@Override
public void play() {
    System.out.println("HoTuanAnh-20226100 Playing track: " + this.getTitle());
    System.out.println("HoTuanAnh-20226100 Track length: " + this.getLength());
}
```

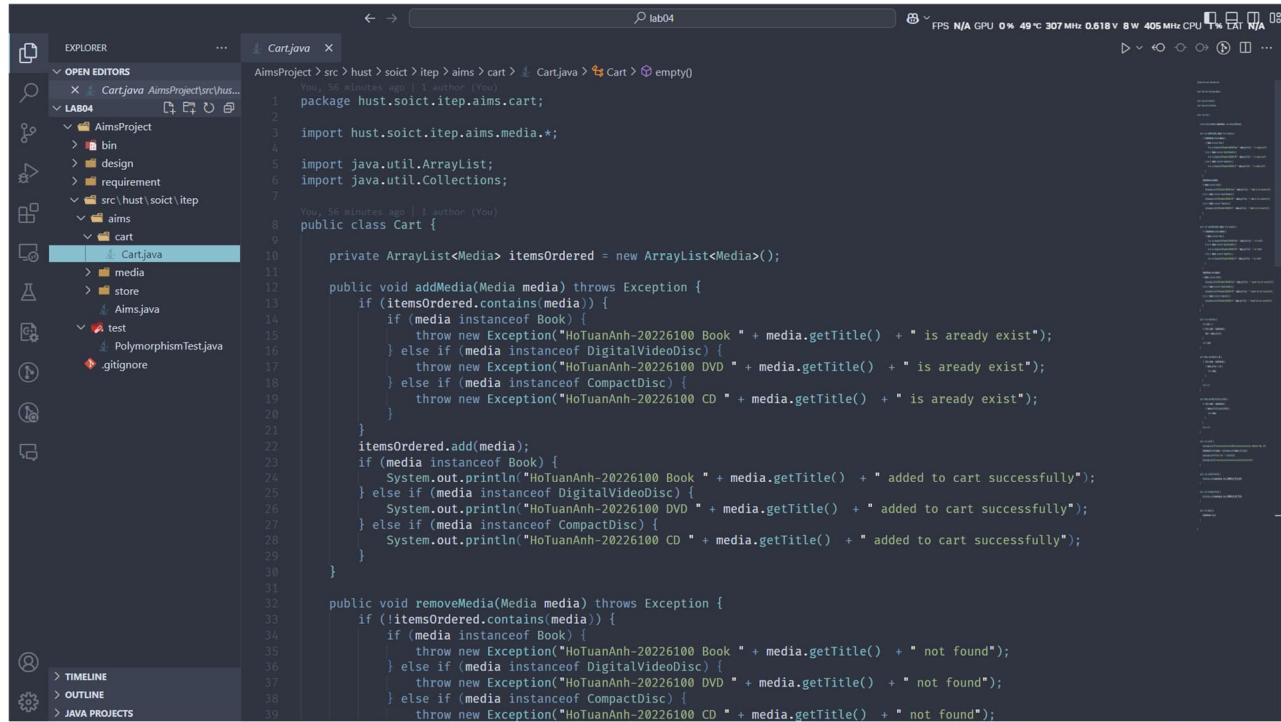
Figure 4.3: Method play() của Track

```
// Trien khai phuong thuc play() trong interface
@Override
public void play() {
    System.out.println("HoTuanAnh-20226100 Playing CD: " + this.getTitle());
    System.out.println("HoTuanAnh-20226100 CD artist: " + artist);
    System.out.println("HoTuanAnh-20226100 CD lenght: " + getLength());
    tracks.forEach(track → 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.



```

package hust.soict.itep.aims.cart;

import hust.soict.itep.aims.media.*;
import java.util.ArrayList;
import java.util.Collections;

public class Cart {
    private ArrayList<Media> itemsOrdered = new ArrayList<Media>();

    public void addMedia(Media media) throws Exception {
        if (!itemsOrdered.contains(media)) {
            if (media instanceof Book) {
                throw new Exception("HoTuanAnh-20226100 Book " + media.getTitle() + " is already exist");
            } else if (media instanceof DigitalVideoDisc) {
                throw new Exception("HoTuanAnh-20226100 DVD " + media.getTitle() + " is already exist");
            } else if (media instanceof CompactDisc) {
                throw new Exception("HoTuanAnh-20226100 CD " + media.getTitle() + " is already exist");
            }
            itemsOrdered.add(media);
            if (media instanceof Book) {
                System.out.println("HoTuanAnh-20226100 Book " + media.getTitle() + " added to cart successfully");
            } else if (media instanceof DigitalVideoDisc) {
                System.out.println("HoTuanAnh-20226100 DVD " + media.getTitle() + " added to cart successfully");
            } else if (media instanceof CompactDisc) {
                System.out.println("HoTuanAnh-20226100 CD " + media.getTitle() + " added to cart successfully");
            }
        }
    }

    public void removeMedia(Media media) throws Exception {
        if (!itemsOrdered.contains(media)) {
            if (media instanceof Book) {
                throw new Exception("HoTuanAnh-20226100 Book " + media.getTitle() + " not found");
            } else if (media instanceof DigitalVideoDisc) {
                throw new Exception("HoTuanAnh-20226100 DVD " + media.getTitle() + " not found");
            } else if (media instanceof CompactDisc) {
                throw new Exception("HoTuanAnh-20226100 CD " + media.getTitle() + " not found");
            }
        }
    }
}

```

Figure 5.1: Cart Class 1

```

AimsProject > src > hust > soict > itep > aims > cart > Cart.java > Cart > empty()
8   public class Cart {
9     public void removeMedia(Media media) throws Exception {
10       if (media instanceof Book) {
11         throw new Exception("HoTuanAnh-20226100 CD " + media.getTitle() + " not found");
12       }
13       itemsOrdered.remove(media);
14       if (media instanceof Book) {
15         System.out.println("HoTuanAnh-20226100 Book " + media.getTitle() + " removed from cart successfully");
16       } else if (media instanceof DigitalVideoDisc) {
17         System.out.println("HoTuanAnh-20226100 DVD " + media.getTitle() + " removed from cart successfully");
18       } else if (media instanceof CompactDisc) {
19         System.out.println("HoTuanAnh-20226100 CD " + media.getTitle() + " removed from cart successfully");
20       }
21     }
22
23     public float totalCost() {
24       float total = 0;
25       for (Media media : itemsOrdered) {
26         total += media.getCost();
27       }
28       return total;
29     }
30
31     public Media searchById(int id) {
32       for (Media media : itemsOrdered) {
33         if (media.getId() == id) {
34           return media;
35         }
36       }
37       return null;
38     }
39
40     public Media searchByTitle(String title) {
41       for (Media media : itemsOrdered) {
42         if (media.getTitle().equals(title)) {
43           return media;
44         }
45       }
46       return null;
47     }
48
49   }
50
51 }
52
53 }
54
55 }
56
57 }
58
59 }
60
61 }
62
63 }
64
65 }
66
67 }
68
69 }
70
71 }
72
73 }
74
75 }
76

```

Figure 5.2: Cart Class 2

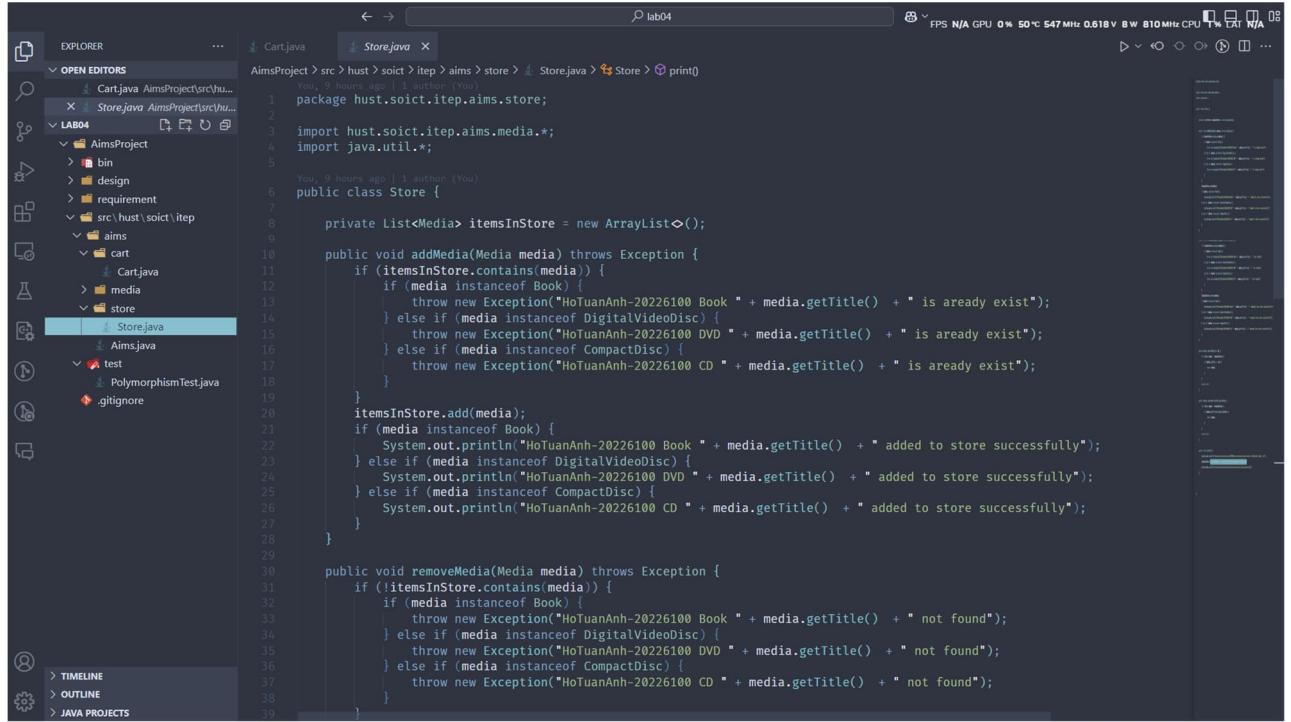
```

AimsProject > src > hust > soict > itep > aims > cart > Cart.java > Cart > getItemsOrdered()
8   public class Cart {
9     public Media searchById(int id) {
10       ...
11       return null;
12     }
13
14     public Media searchByTitle(String title) {
15       for (Media media : itemsOrdered) {
16         if (media.getTitle().equals(title)) {
17           ...
18           return media;
19         }
20       }
21       return null;
22     }
23
24     public void print() {
25       System.out.println("\n*****CART*****\nOrdered items: \n");
26       itemsOrdered.forEach(media -> System.out.println(media.toString()));
27       System.out.println("Total cost: " + totalCost());
28       System.out.println("*****\n");
29     }
30
31     public void sortByTitleCost() {
32       Collections.sort(itemsOrdered, Media.COMPARE_BY_TITLE_COST);
33     }
34
35     public void sortByCostTitle() {
36       Collections.sort(itemsOrdered, Media.COMPARE_BY_COST_TITLE);
37     }
38
39     public ArrayList<Media> getItemsOrdered() { You, 1 second ago + Uncommitted changes
40       ...
41       return itemsOrdered;
42     }
43
44     public void empty() {
45       itemsOrdered.clear();
46     }
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102

```

Figure 5.3: Cart Class 3

6 Update the Store class to work with Media



```

AimsProject > src > hust > soict > itep > aims > store > Store.java > Store > print()
You, 9 hours ago | 1 author (You)
package hust.soict.itep.aims.store;
import hust.soict.itep.aims.media.*;
import java.util.*;

private List<Media> itemsInStore = new ArrayList<>();

public void addMedia(Media media) throws Exception {
    if (itemsInStore.contains(media)) {
        if (media instanceof Book) {
            throw new Exception("HoTuanAnh-20226100 Book " + media.getTitle() + " is already exist");
        } else if (media instanceof DigitalVideoDisc) {
            throw new Exception("HoTuanAnh-20226100 DVD " + media.getTitle() + " is already exist");
        } else if (media instanceof CompactDisc) {
            throw new Exception("HoTuanAnh-20226100 CD " + media.getTitle() + " is already exist");
        }
    }
    itemsInStore.add(media);
    if (media instanceof Book) {
        System.out.println("HoTuanAnh-20226100 Book " + media.getTitle() + " added to store successfully");
    } else if (media instanceof DigitalVideoDisc) {
        System.out.println("HoTuanAnh-20226100 DVD " + media.getTitle() + " added to store successfully");
    } else if (media instanceof CompactDisc) {
        System.out.println("HoTuanAnh-20226100 CD " + media.getTitle() + " added to store successfully");
    }
}

public void removeMedia(Media media) throws Exception {
    if (!itemsInStore.contains(media)) {
        if (media instanceof Book) {
            throw new Exception("HoTuanAnh-20226100 Book " + media.getTitle() + " not found");
        } else if (media instanceof DigitalVideoDisc) {
            throw new Exception("HoTuanAnh-20226100 DVD " + media.getTitle() + " not found");
        } else if (media instanceof CompactDisc) {
            throw new Exception("HoTuanAnh-20226100 CD " + media.getTitle() + " not found");
        }
    }
}

```

Figure 6.1: Store Class 1

```

AimsProject > src > hust > soict > itep > aims > store > Store.java > Store > print()
6  public class Store {
7      public void removeMedia(Media media) throws Exception {
8          if (media instanceof Book) {
9              throw new Exception("HoTuanAnh-20226100 DVD " + media.getTitle() + " not found");
10         } else if (media instanceof CompactDisc) {
11             throw new Exception("HoTuanAnh-20226100 CD " + media.getTitle() + " not found");
12         }
13         itemsInStore.remove(media);
14         if (media instanceof Book) {
15             System.out.println("HoTuanAnh-20226100 Book " + media.getTitle() + " removed from store successfully");
16         } else if (media instanceof DigitalVideoDisc) {
17             System.out.println("HoTuanAnh-20226100 DVD " + media.getTitle() + " removed from store successfully");
18         } else if (media instanceof CompactDisc) {
19             System.out.println("HoTuanAnh-20226100 CD " + media.getTitle() + " removed from store successfully");
20         }
21     }
22
23     public Media searchById(int id) {
24         for (Media media : itemsInStore) {
25             if (media.getId() == id) {
26                 return media;
27             }
28         }
29         return null;
30     }
31
32     public Media searchByTitle(String title) {
33         for (Media media : itemsInStore) {
34             if (media.getTitle().equals(title)) {
35                 return media;
36             }
37         }
38         return null;
39     }
40
41     public void print() {
42         System.out.println("\n*****STORE*****\nOrdered items: \n");
43         itemsInStore.forEach(media -> System.out.println(media.toString()));
44         System.out.println("*****\n");
45     }
46 }

```

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

```
public CompactDisc(int id, String title, String category, float cost, String director, String artist, List<Track> tracks)
    super(id, title, category, cost, director);
    this.artist = artist;
    this.tracks = tracks;
}
```

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() {}

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

Figure 7.3: Constructor Media Clas

```
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;
}

public Disc(int id, String title, String category, float cost, String director) {
    super(id, title, category, 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 lặp vào danh sách.

```
// Ghi đè phương thức equals() mặc định
@Override
public boolean equals(Object o) {
    Media media = (Media) o;
    return media.getTitle().equals(this.getTitle());
}
```

Figure 8.1: Override equals in Media Class

```
// Ghi đè phương thức equals() mặc định
@Override
public boolean equals(Object o) {
    Track track = (Track) o;
    return track.getTitle().equals(this.getTitle()) && (track.getLength() == this.getLength());
}
```

Figure 8.2: Override equals in Track Class

9 Polymorphism with `toString()` method

```

AimsProject > src > hust > soict > itep > test > PolymorphismTest.java > PolymorphismTest > main(String[])
You, 1 hour ago | 1 author (You)
1 package hust.soict.itep.test;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 import hust.soict.itep.aims.media.*;
7
8
9 public class PolymorphismTest {
10     public static void main(String[] args) {
11         List<Media> mediae = new ArrayList<Media>();
12         DigitalVideoDisc dvd = new DigitalVideoDisc(id:1, title:"dvd 1", category:"dvd", cost:5, length:10, director:"xxx");
13
14         List<Track> tracks = new ArrayList<Track>();
15         tracks.add(new Track(title:"track 1",length:3));
16         tracks.add(new Track(title:"track 2",length:4));
17         CompactDisc cd = new CompactDisc(id:2, title:"cd 1", category:"cd", cost:2.5f, director:"yyy", artist:"abc", tracks);
18
19         List<String> authors = new ArrayList<String>();
20         authors.add("name 1");
21         authors.add("name 2");
22         Book book = new Book(id:3,title:"book 1", category:"book", cost:10, authors);
23
24         mediae.add(dvd);
25         mediae.add(cd);
26         mediae.add(book);
27
28         for(Media m : mediae) { You, 1 hour ago * update source
29             System.out.println(m.toString());
30         }
31     }
32 }
33

```

Figure 9.1: Code mô phỏng Polymorphism

Kết quả

```

AimsProject > src > hust > soict > itep > test > PolymorphismTest.java > PolymorphismTest > main(String[])
You, 1 hour ago | 1 author (You)
PROBLEMS OUTPUT TERMINAL
... ^ x
+ ... > >
power... Run A... Run P...
PS D:\WORK\Programs\Java\IT3103.744528.2024.1.20226100.HoTuanAnh\lab04> ^
PS D:\WORK\Programs\Java\IT3103.744528.2024.1.20226100.HoTuanAnh\lab04> d; cd 'd:\WORK\Programs\Java\IT3103.744528.2024.1.20226100.6971cffbb1bd2ee94a09f6c99f31e\redhat.java\jdt_ws\lab04_8128e8c4\bin'\hust.soict.itep.test.PolymorphismTest'
HoTuanAnh20226100 DVD: {
    id: 1
    title: dvd 1
    category: dvd
    length: 10
    director: xxx
    cost: 5.0$
}
HoTuanAnh20226100 CD: {
    id: 2
    title: cd 1
    category: cd
    director: yyy
    track: [
        title: track 1,
        length: 3
        title: track 2,
        length: 4
    ]
    cost: 2.5$
}
HoTuanAnh20226100 Book: {
    id: 3
    title: book 1
    category: book
    authors: [name 1, name 2]
    cost: 10.0$
}
PS D:\WORK\Programs\Java\IT3103.744528.2024.1.20226100.HoTuanAnh\lab04> []

```

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.

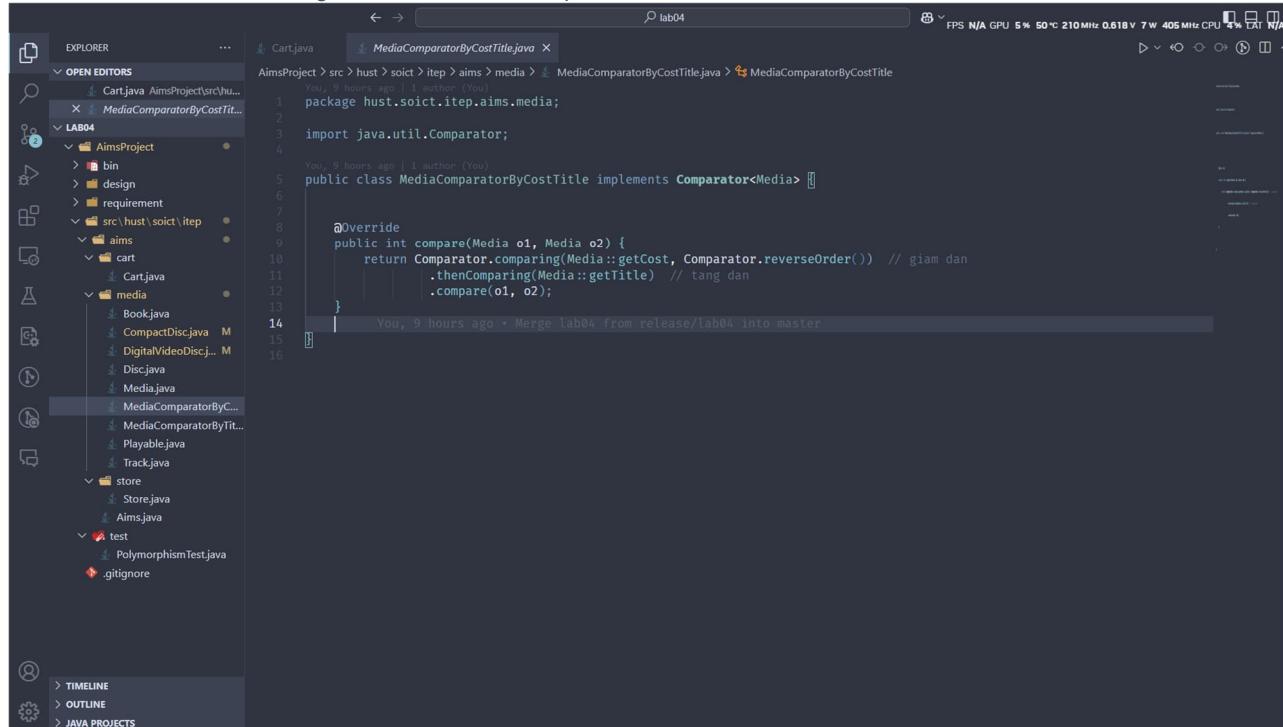
10 Sort media in the car

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

```
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



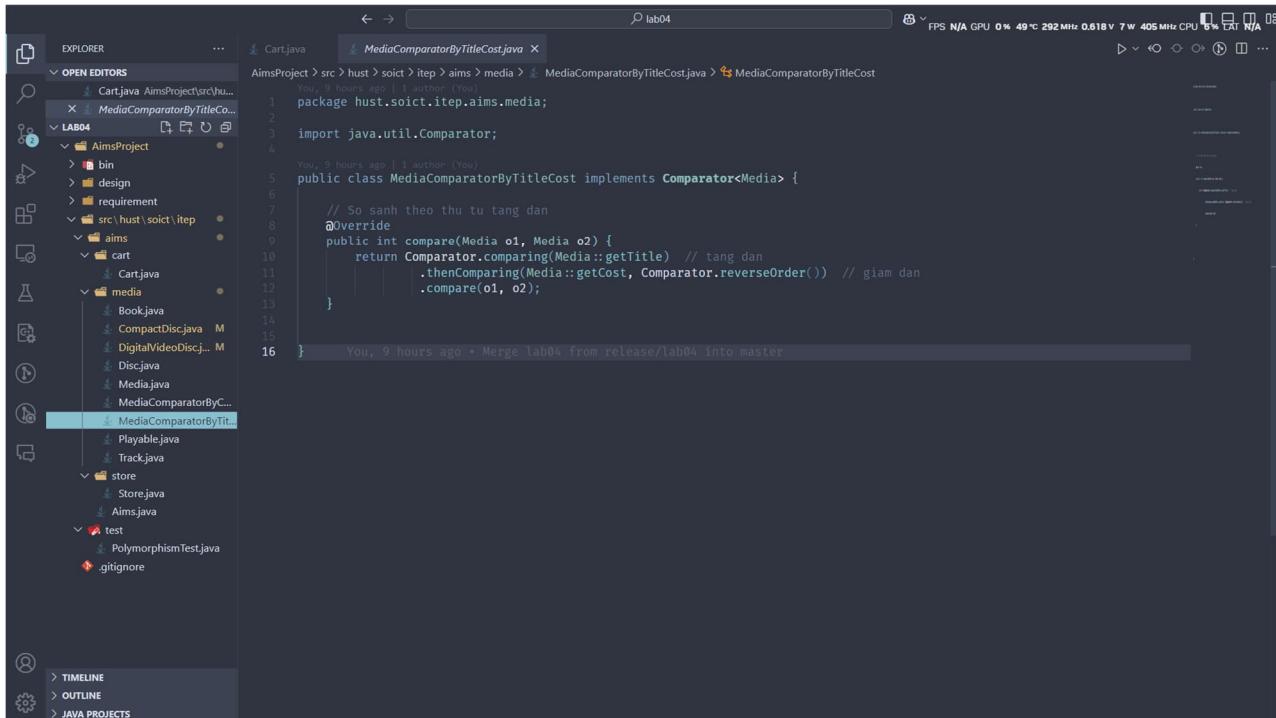
```
package hust.soict.itep.aims.media;

import java.util.Comparator;

public class MediaComparatorByCostTitle implements Comparator<Media> {

    @Override
    public int compare(Media o1, Media o2) {
        return Comparator.comparing(Media::getCost, Comparator.reverseOrder()) // giam dan
                .thenComparing(Media::getTitle) // tang dan
                .compare(o1, o2);
    }
}
```

Figure 10.2: MediaComparatorByCostTitle Class

Figure 10.3: `MediaComparatorByTitleCost` Class

11 Create a complete console application in the Aims class

```

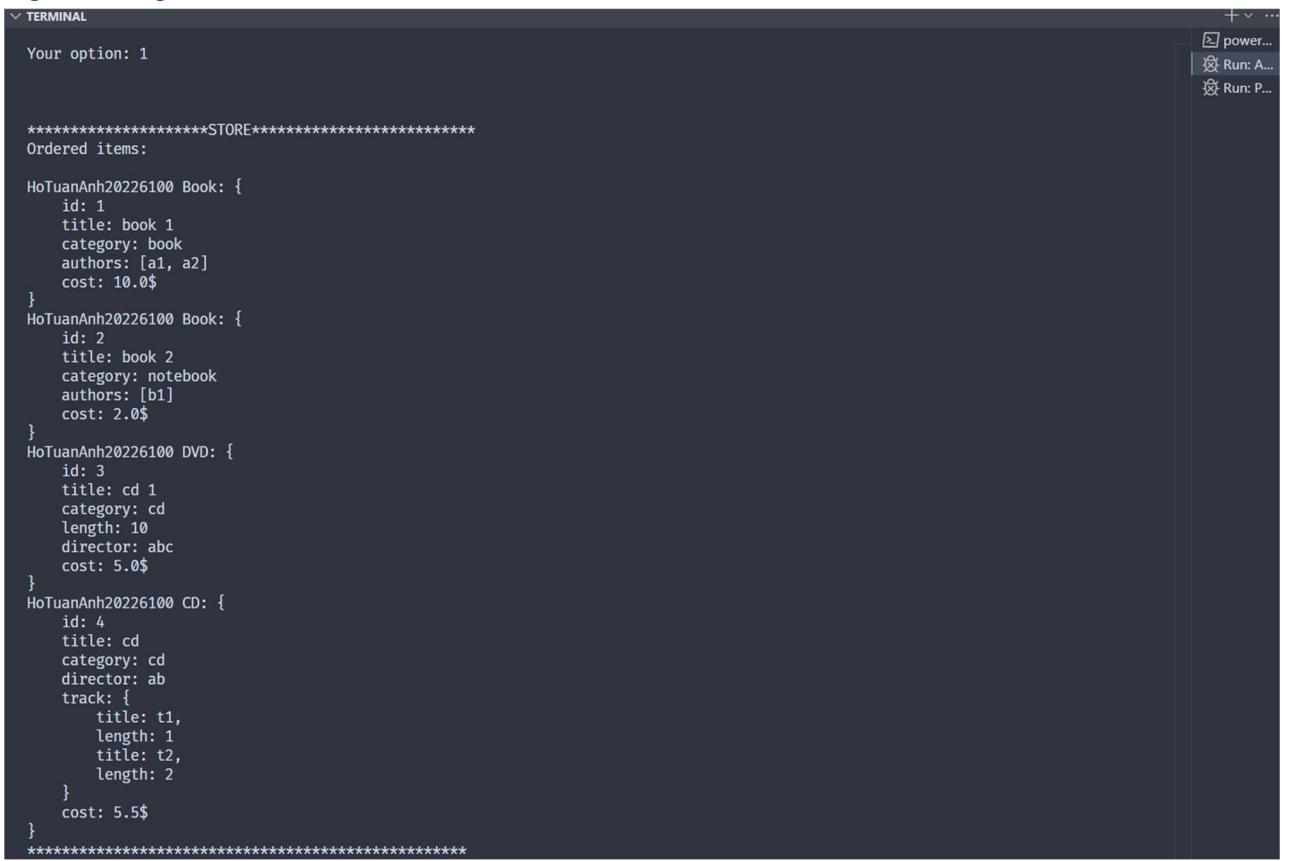
PS D:\WORK\Programs\Java\IT3103.744528.2024.1.20226100.HoTuanAnh\Lab04> d:; cd 'd:\WORK\Programs\Java\IT3103.744528.2024.1.20226100. HoTuanAnh\Lab04'; & 'C:\Program Files\Java\jre-1.8\bin\java.exe' '-cp' 'C:\Users\xavii\AppData\Roaming\Code\User\workspaceStorage\7516971cffbb1bd2ee94a09f6c99f31e\redhat.java\jdt_ws\lab04_8128e8c4\bin' 'hust.soict.itep.aims.Aims'

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

```

Figure 11.1: Màn hình chính

11.1 Người dùng chọn 1: View store



```
+ v ...
power...
Run: A...
Run: P...
TERMINAL
Your option: 1

*****
Ordered items:
HoTuanAnh20226100 Book: {
    id: 1
    title: book 1
    category: book
    authors: [a1, a2]
    cost: 10.0$
}
HoTuanAnh20226100 Book: {
    id: 2
    title: book 2
    category: notebook
    authors: [b1]
    cost: 2.0$
}
HoTuanAnh20226100 DVD: {
    id: 3
    title: cd 1
    category: cd
    length: 10
    director: abc
    cost: 5.0$
}
HoTuanAnh20226100 CD: {
    id: 4
    title: cd
    category: cd
    director: ab
    track: {
        title: t1,
        length: 1
        title: t2,
        length: 2
    }
    cost: 5.5$
}
*****
```

Figure 11.2: Vào Trang View Store

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

```

TERMINAL
title: t1,
length: 1
title: t2,
length: 2
}
cost: 5.5$
}
*****
Options:
-----
1. See a media 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
Your option: 1
Enter media's title: cd
HoTuanAnh20226100 CD: {
id: 4
title: cd
category: cd
director: ab
track: {
title: t1,
length: 1
title: t2,
length: 2
}
cost: 5.5$
}
Options:
-----
1. Add to cart
2. Play
0. Back
-----
Please choose a number: 0-1-
Your option: 

```

Figure 11.3: See a media's details

```

Options:
-----
1. See a media 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
Your option: 1
Enter media's title: cd
HoTuanAnh20226100 CD: {
id: 4
title: cd
category: cd
director: ab
track: {
title: t1,
length: 1
title: t2,
length: 2
}
cost: 5.5$
}
Options:
-----
1. Add to cart
2. Play
0. Back
-----
Please choose a number: 0-1-
Your option: 1
HoTuanAnh-20226100 CD cd added to cart successfully
Options:
-----
1. Add to cart
2. Play
0. Back
-----
Please choose a number: 0-1-
Your option: 

```

Figure 11.4: Thêm vào Cart

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

```
Options:  
-----  
1. See a media 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  
Your option: 2  
Enter media's title: book 1  
HoTuanAnh-20226100 Book book 1 added to cart successfully
```

Figure 11.5: Thêm media vào Cart

11.1.3 Người dùng chọn 3: Play a media

```
Options:  
-----  
1. See a media 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  
Your option: 3  
Enter media's title: cd  
HoTuanAnh-20226100 Playing CD: cd  
HoTuanAnh-20226100 CD artist: xyz  
HoTuanAnh-20226100 CD lenght: 3  
HoTuanAnh-20226100 Playing track: t1  
HoTuanAnh-20226100 Track length: 1  
HoTuanAnh-20226100 Playing track: t2  
HoTuanAnh-20226100 Track length: 2
```

Figure 11.6: Play a media

11.1.4 Người dùng chọn 4: See current cart

```

4. See current cart
0. Back
-----
Please choose a number: 0-1-2-3-4
Your option: 4

*****CART*****
Ordered items:

HoTuanAnh20226100 CD: {
    id: 4
    title: cd
    category: cd
    director: ab
    track: {
        title: t1,
        length: 1
        title: t2,
        length: 2
    }
    cost: 5.5$
}
HoTuanAnh20226100 Book: {
    id: 1
    title: book 1
    category: book
    authors: [a1, a2]
    cost: 10.0$
}
Total cost: 15.5
*****


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 11.7: 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  
Your option: 2  
  
Options:  
-----  
1. Add media  
2. Remove media  
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

```
Options:  
-----  
1. Add media  
2. Remove media  
0. Back  
-----  
Please choose a number: 0-1-2  
Your option: 1  
Choose media type:  
-----  
1. Book  
2. DVD  
3. CD  
0. Back  
-----  
Please choose a number: 0-1-2-3  
Your option: 3  
Enter id: 4  
Enter title: cd 1  
Enter category: cd  
Enter cost: 5.5  
Enter director's name: ab  
Enter artist's name: xxx  
Enter amount of tracks: 2  
Enter title of track 1 : t1  
Enter length of track 1: 2  
Enter title of track 2 : t2  
Enter length of track 2: 3  
HoTuanAnh-20226100 CD cd 1 added to store successfully
```

Figure 11.9: Add a media to store

=> Kết quả sau khi thêm

```
*****STORE*****
```

Ordered items:

```
HoTuanAnh20226100 Book: {  
    id: 1  
    title: book 1  
    category: book  
    authors: [a1, a2]  
    cost: 10.0$  
}  
HoTuanAnh20226100 Book: {  
    id: 2  
    title: book 2  
    category: book  
    authors: [a3]  
    cost: 2.0$  
}  
HoTuanAnh20226100 DVD: {  
    id: 3  
    title: dvd  
    category: dvd  
    length: 20  
    director: abc  
    cost: 5.0$  
}  
HoTuanAnh20226100 CD: {  
    id: 4  
    title: cd 1  
    category: cd  
    director: ab  
    track: {  
        title: t1,  
        length: 2  
        title: t2,  
        length: 3  
    }  
    cost: 5.5$  
}
```

```
*****
```

Figure 11.10: Result after add media to store

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

```
AIMS:  
-----  
-----  
1. View store  
2. Update store  
3. See current cart  
0. Exit  
-----  
Please choose a number: 0-1-2-3  
Your option: 2  
  
Options:  
-----  
1. Add media  
2. Remove media  
0. Back  
-----  
Please choose a number: 0-1-2  
Your option: 2  
Enter media's title:  
book 2  
HoTuanAnh-20226100 Book book 2 removed from store successfully  
  
AIMS:  
-----  
1. View store  
2. Update store  
3. See current cart  
0. Exit  
-----  
Please choose a number: 0-1-2-3  
Your option: 1
```

Figure 11.11: Remove a media from the store

=> Kết quả sau khi remove

```
Your option: 1

*****
*****STORE*****
Ordered items:

HoTuanAnh20226100 Book: {
    id: 1
    title: book 1
    category: book
    authors: [a1, a2]
    cost: 10.0$
}
HoTuanAnh20226100 DVD: {
    id: 3
    title: dvd
    category: dvd
    length: 20
    director: abc
    cost: 5.0$
}
HoTuanAnh20226100 CD: {
    id: 4
    title: cd 1
    category: cd
    director: ab
    track: {
        title: t1,
        length: 2
        title: t2,
        length: 3
    }
    cost: 5.5$
}
*****
```

Figure 11.12: Result after remove a media

11.3 Người dùng chọn 4: See current cart

```
Options:  
-----  
1. See a media 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  
Your option: 4  
*****CART*****  
Ordered items:  
  
HoTuanAnh20226100 Book: {  
    id: 1  
    title: book 1  
    category: book  
    authors: [a1, a2]  
    cost: 10.0$  
}  
HoTuanAnh20226100 CD: {  
    id: 4  
    title: cd 1  
    category: cd  
    director: ab  
    track: {  
        title: t1,  
        length: 2  
        title: t2,  
        length: 3  
    }  
    cost: 5.5$  
}  
Total cost: 15.5  
*****
```

Figure 11.13: Vào trang See current cart

Giả sử lúc này trong Cart sẽ có các Media sau

11.3.1 Người dùng chọn 1: Filter medias in cart

```
0. Back
-----
Please choose a number: 0-1-2-3-4-5
Your option: 1
Option:

-----
1. Filter by id
2. Filter by title

-----
Please choose a number: 1-2
Your option: 1
Enter media id: 1
HoTuanAnh20226100 Book: {
    id: 1
    title: book 1
    category: book
    authors: [a1, a2]
    cost: 10.0$
}

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
Your option: █
```

Figure 11.15: Filter Cart By id

```
Please choose a number: 0-1-2-3-4-5
Your option: 1
Option:

1. Filter by id
2. Filter by title

Please choose a number: 1-2
Your option: 2
Enter media title: cd 1
HoTuanAnh20226100 CD: {
    id: 4
    title: cd 1
    category: cd
    director: abc
    track: {
        title: t1,
        length: 2
        title: t2,
        length: 3
    }
    cost: 5.5$
}

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
Your option: █
```

Figure 11.16: Filter Cart By Title

11.3.2 Người dùng chọn 2: Sort medias in cart

```
*****CART*****
Ordered items:

HoTuanAnh20226100 Book: {
    id: 1
    title: book 1
    category: book
    authors: [a1, a2]
    cost: 10.0$
}
HoTuanAnh20226100 CD: {
    id: 4
    title: cd 1
    category: cd
    director: abc
    track: {
        title: t1,
        length: 2
        title: t2,
        length: 3
    }
    cost: 5.5$
}
HoTuanAnh20226100 CD: {
    id: 5
    title: cd 2
    category: cd
    director: abcd
    track: {
        title: t3,
        length: 10
    }
    cost: 8.0$
}
HoTuanAnh20226100 DVD: {
    id: 3
    title: dvd 1
    category: dvd
    length: 20
    director: aaa
    cost: 5.0$
}
Total cost: 28.5
```

Figure 11.17: Sort Cart By Title

```
Ordered items:  
HoTuanAnh20226100 Book: {  
    id: 1  
    title: book 1  
    category: book  
    authors: [a1, a2]  
    cost: 10.0$  
}  
HoTuanAnh20226100 CD: {  
    id: 5  
    title: cd 2  
    category: cd  
    director: abcd  
    track: {  
        title: t3,  
        length: 10  
    }  
    cost: 8.0$  
}  
HoTuanAnh20226100 CD: {  
    id: 4  
    title: cd 1  
    category: cd  
    director: abc  
    track: {  
        title: t1,  
        length: 2  
        title: t2,  
        length: 3  
    }  
    cost: 5.5$  
}  
HoTuanAnh20226100 DVD: {  
    id: 3  
    title: dvd 1  
    category: dvd  
    length: 20  
    director: aaa  
    cost: 5.0$  
}  
Total cost: 28.5  
*****
```

Figure 11.18: Sort Cart By Cost

11.3.3 Người dùng chọn 3: Remove media from 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
Your option: 3
Enter media's title: dvd 1
HoTuanAnh-20226100 DVD dvd 1 removed from cart successfully
```

Figure 11.19: Remove media by id

=> Kết quả

```
Please choose a number: 0-1-2-3-4-5
Your option: 2
Option:

1. Sort by title
2. Sort by cost

Please choose a number: 1-2
Your option: 2

*****CART*****
Ordered items:

HoTuanAnh20226100 Book: {
    id: 1
    title: book 1
    category: book
    authors: [a1, a2]
    cost: 10.0$
}
HoTuanAnh20226100 CD: {
    id: 5
    title: cd 2
    category: cd
    director: abcd
    track: {
        title: t3,
        length: 10
    }
    cost: 8.0$
}
HoTuanAnh20226100 CD: {
    id: 4
    title: cd 1
    category: cd
    director: abc
    track: {
        title: t1,
        length: 2
        title: t2,
        length: 3
    }
    cost: 5.5$
}
Total cost: 23.5$
```

Figure 11.20: Result after remove media in cart by id

11.3.4 Người dùng chọn 4: Play a media

```
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
Your option: 4
Enter media's title: book 1
This media is unplayable

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
Your option: 4
Enter media's title: cd 1
HoTuanAnh-20226100 Playing CD: cd 1
HoTuanAnh-20226100 CD artist: xxx
HoTuanAnh-20226100 CD lenght: 5
HoTuanAnh-20226100 Playing track: t1
HoTuanAnh-20226100 Track length: 2
HoTuanAnh-20226100 Playing track: t2
HoTuanAnh-20226100 Track length: 3
```

Figure 11.21: Play a media in cart

11.3.5 Người dùng chọn 5: Place order

```
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  
Your option: 5  
An order is created.
```

Figure 11.22: Order

=> Kết quả sau khi order

```
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  
Your option: 5  
An order is created.  
  
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  
Your option: 0  
  
Options:  
-----  
1. See a media 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  
Your option: 4  
  
*****CART*****  
Ordered items:  
  
Total cost: 0.0$  
*****
```

Figure 11.23: Result after order

Class Diagram

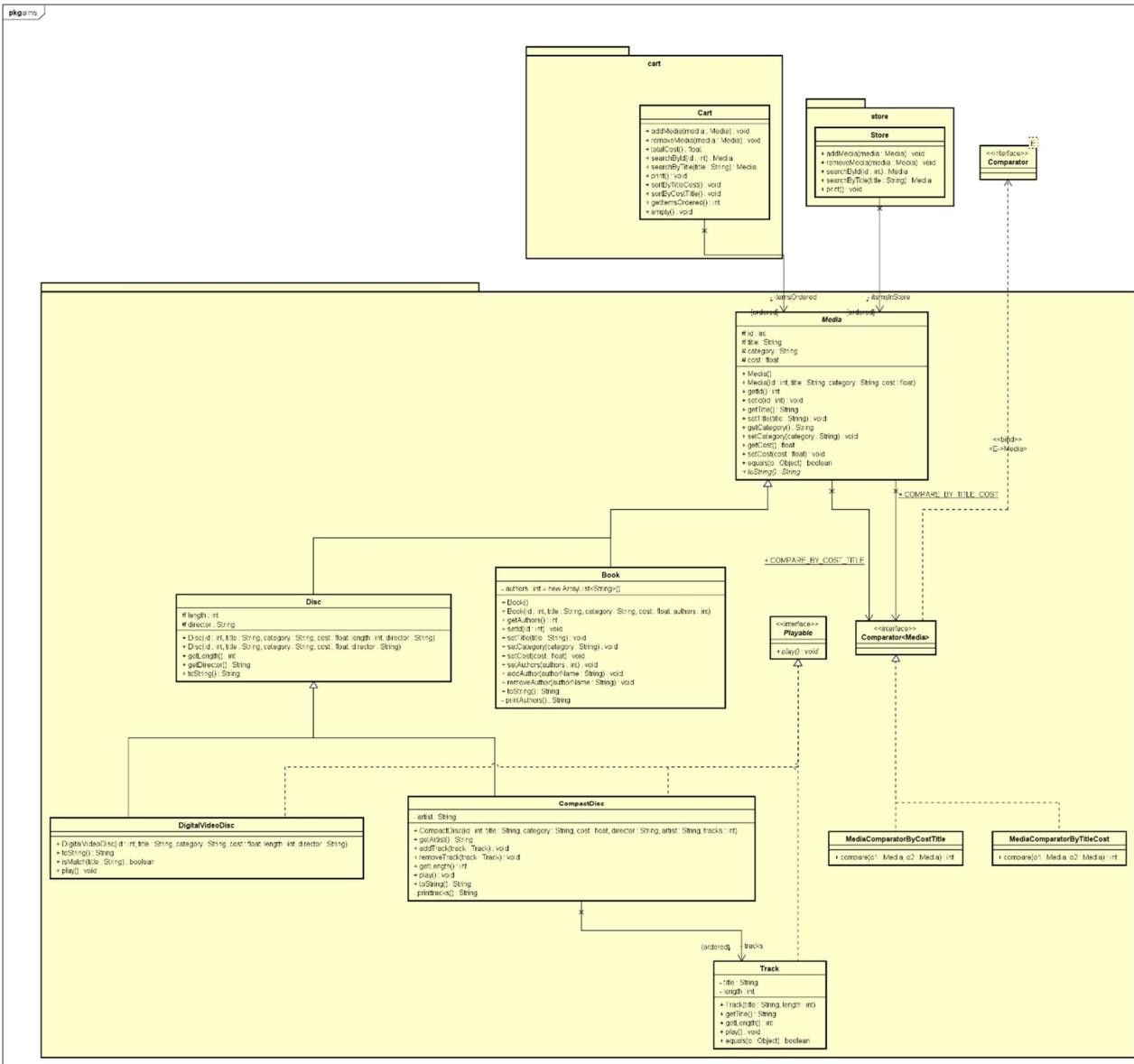


Figure 12.1: Class Diagram

12 UseCase Diagram

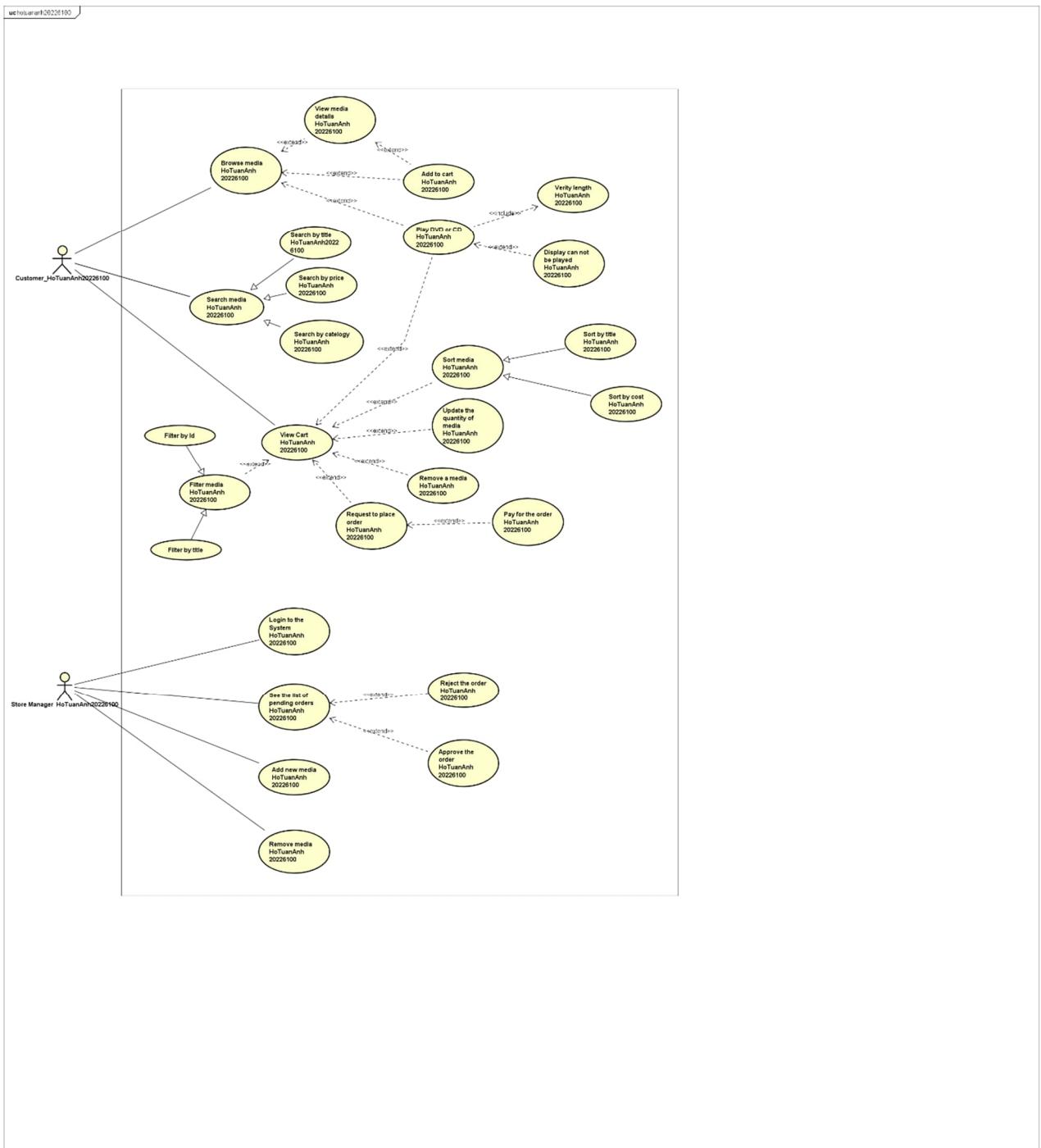


Figure 13.1: UseCase Diagram

13 Answer Questions

Câu hỏi (Tiếng Việt): Thay vì sử dụng Comparator để so sánh các mục trong giỏ hàng, chúng ta có thể sử dụng giao diện Comparable và ghi đè phương thức compareTo(). Giả sử chúng ta đang áp dụng Comparable interface approach.

- Lớp nào nên triển khai giao diện Comparable?

→ Lớp nên triển khai giao diện Comparable là lớp chứa đối tượng mà bạn muốn so sánh, trong trường hợp này, là abstract class “Media”.

- Trong các lớp đó, phải triển khai phương thức compareTo() như thế nào để phản ánh đúng thứ tự mà chúng ta muốn?

→ Để triển khai phương thức compareTo() thì cần so sánh các thuộc tính của đối tượng. Đối với yêu cầu sắp xếp theo title sau đó là cost, có thể triển khai như sau:

@Override

```
public int compareTo(Media other) {
    if (!this.getTitle().equals(other.getTitle())) {
        return this.getTitle().compareTo(other.getTitle());
    } else {
        return Double.compare(this.getCost(), other.getCost());
    }
}
```

- Liệu có thể có hai quy tắc sắp xếp (theo title sau đó là cost và theo cost sau đó là title) nếu sử dụng cách tiếp cận này với giao diện Comparable?

→ Không, với giao diện Comparable, bạn chỉ có thể có một quy tắc sắp xếp cho mỗi lớp. Điều này là do phương thức compareTo() chỉ trả về một giá trị int.

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.

- Giả sử các đĩa DVD có một quy tắc sắp xếp khác biệt so với các loại phương tiện khác, tức là theo tiêu đề, sau đó là độ dài giảm dần, sau đó là giá. Bạn sẽ làm thế nào để điều chỉnh mã của mình để cho phép điều này?

→ Có thể tạo một lớp riêng cho DVD và triển khai phương thức compareTo():

```
public class DVD extends Media implements Comparable<DVD> {
    @Override
    public int compareTo(DVD other) {
        if (!this.getTitle().equals(other.getTitle())) {
            return this.getTitle().compareTo(other.getTitle());
        } else if (this instanceof DVD && other instanceof DVD) {
            return Integer.compare(((DVD) other).getLength(), ((DVD)
this).getLength());
        } else {
            return Double.compare(this.getCost(), other.getCost());
        }
    }
}

6   public abstract class Media implements Comparable<Media>{      You, 1 secon
74  ||| @Override
75  ||| public int compareTo(Media other) {
76  |||     if (!this.getTitle().equals(other.getTitle())) {
77  |||         return this.getTitle().compareTo(other.getTitle());
78  |||     } else {
79  |||         return Double.compare(this.getCost(), other.getCost());
80  |||     }
81  ||}
82  ||
83  ||}
84 }
```

Figure 14.1: Triển khai Comparable trong lớp trừu tượng Media

```
public class DigitalVideoDisc extends Disc implements Playable, Comparable<DigitalVideoDisc> {      You, I second a
}

@Override
public int compareTo(DigitalVideoDisc other) {
    if (!this.getTitle().equals(other.getTitle())) {
        return this.getTitle().compareTo(other.getTitle());
    } else if (this instanceof DigitalVideoDisc && other instanceof DigitalVideoDisc) {
        return Integer.compare(((DigitalVideoDisc) other).getLength(), ((DigitalVideoDisc) this).getLength());
    } else {
        return Double.compare(this.getCost(), other.getCost());
    }
}
```

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.