

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



BÁO CÁO THỰC HÀNH LAB 4
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Giảng viên hướng dẫn: Trịnh Tuấn Đạt

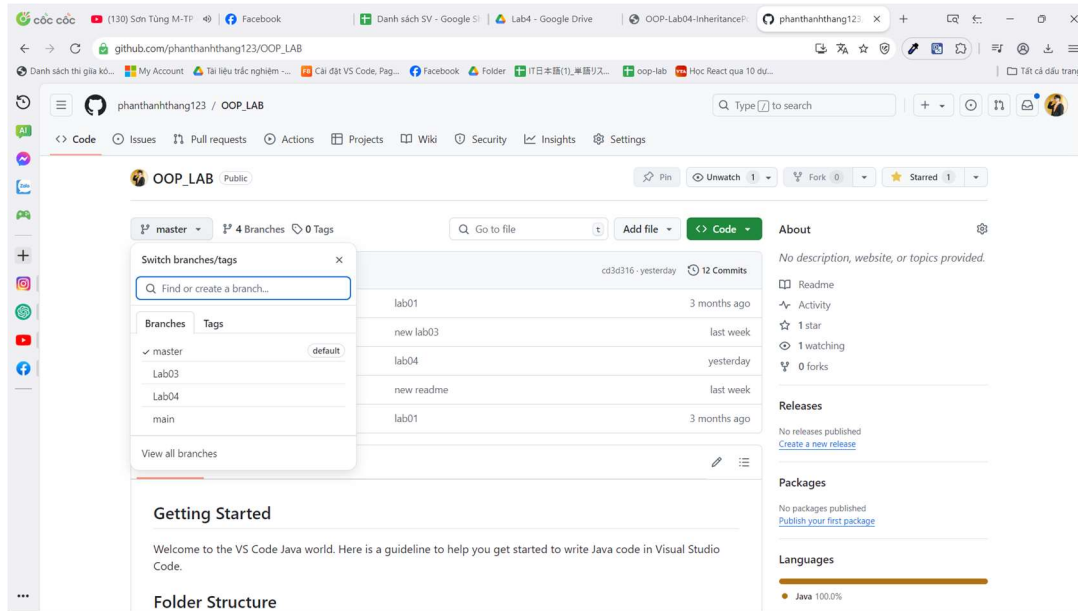
Sinh viên thực hiện: Phan Thanh Thắng

MSSV: 20225927

Mã học phần: IT3103

Mã lớp học: 744521

- Mã nguồn được quản lý bằng Git và nộp vào nhánh release/lab04



- Các câu trả lời được lưu trong tập “answers”

1. Git

- Tạo nhánh mới cho mỗi tính năng: “topic/aims-project/add-media-class”
- Đẩy lên kho lưu trữ và hợp nhất mới nhánh chính “main”

2. Yêu cầu bổ sung cho AIMS

- Mở rộng hệ thống AIMS để hỗ trợ thêm 2 loại phương tiện: sách và CD, bên cạnh DVD đã có từ trước.
- Attribute của từng class:
 - Sách: id, tiêu đề, thể loại, giá và danh sách tác giả
 - CD: id, tiêu đề, thể loại, nghệ sĩ, đạo diễn, danh sách bài hát và giá
- Yêu cầu của phương thức toString() với mỗi class:
 - Sách: tiêu đề, thể loại, danh sách tác giả, độ dài nội dung (số lượng từ)
 - CD: tiêu đề, thể loại, nghệ sĩ, đạo diễn, độ dài CD, giá, thông tin từng bài hát.
 - Tiêu đề, thể loại, đạo diễn, độ dài DVD và giá.

- Một số chức năng khác:
 - CD: phát tất cả các bài hát, hiển thị thông tin CD(Tiêu đề, độ dài)
 - DVD: hiển thị tiêu đề và độ dài DVD.
- Tạo lớp Book:

```

src > Lab02 > hust > soict > dsai > aims > media > Bookjava > Book > authors
1 //Phan Thanh Thang
2 //20225927
3 //class Book
4
5 package Lab02.hust.soict.dsai.aims.media;
6
7 import java.util.ArrayList;
8 import java.util.List;
9
10 public class Book {
11
12     private int id;
13     private String title;
14     private String category;
15     private float cost;
16     private List<String> authors = new ArrayList<String>();
17
18
19
20 //constructor
21 public Book() {
22
23 }
24 public Book(int id, String title, String category, float cost, List<String> authors) {
25     this.id = id;
26     this.title = title;
27     this.category = category;
28     this.cost = cost;
29     this.authors = authors;
30 }
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56

```

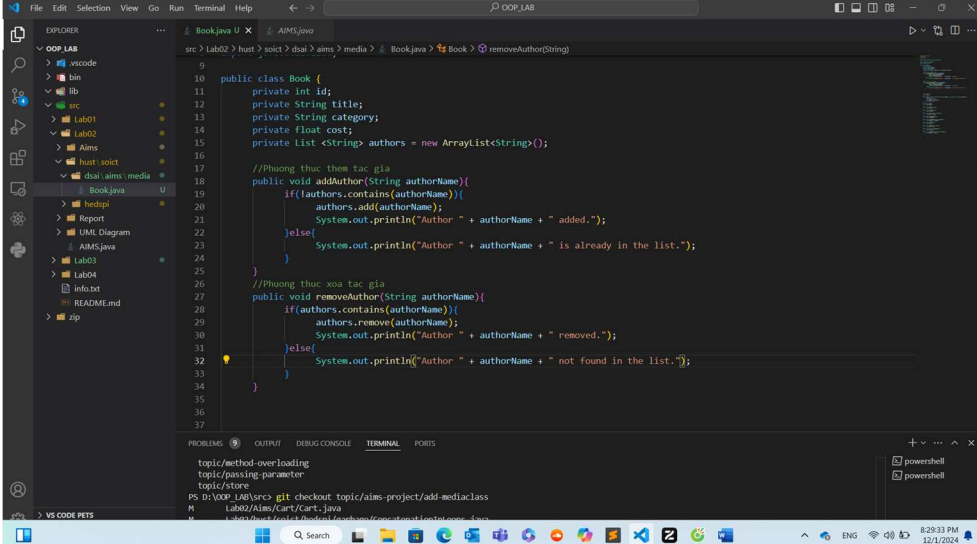
Thêm các phương thức set() và get() cho các atribute trừ authors:

```

src > Lab02 > hust > soict > dsai > aims > media > Bookjava > Book > @getAuthors()
10 public class Book {
11
12     private int id;
13     private String title;
14     private String category;
15     private float cost;
16     private List<String> authors = new ArrayList<String>();
17
18 //setter or getter
19 public int getId() {
20     return id;
21 }
22 public void setId(int id) {
23     this.id = id;
24 }
25 public String getTitle() {
26     return title;
27 }
28 public void setTitle(String title) {
29     this.title = title;
30 }
31 public String getCategory() {
32     return category;
33 }
34 public void setCategory(String category) {
35     this.category = category;
36 }
37 public float getCost() {
38     return cost;
39 }
40 public void setCost(float cost) {
41     this.cost = cost;
42 }
43 public List<String> getAuthors() {
44     return authors;
45 }
46
47
48
49
50
51
52
53
54
55
56

```

Thêm phương thức addAuthor() và phương thức removeAuthor()

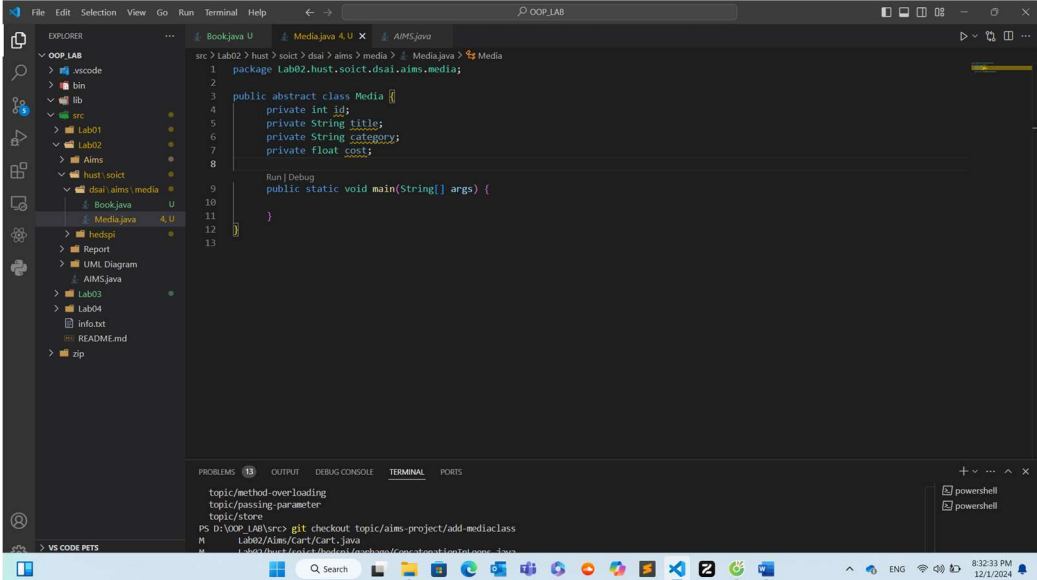


The screenshot shows the VS Code editor with the `Book.java` file open. The file contains the following code:

```
9
10 public class Book {
11     private int id;
12     private String title;
13     private String category;
14     private float cost;
15     private List<String> authors = new ArrayList<String>();
16
17     //Phương thức thêm tác giả
18     public void addAuthor(String authorName){
19         if(!authors.contains(authorName)){
20             authors.add(authorName);
21             System.out.println("Author " + authorName + " added.");
22         }else{
23             System.out.println("Author " + authorName + " is already in the list.");
24         }
25     }
26
27     //Phương thức xóa tác giả
28     public void removeAuthor(String authorName){
29         if(authors.contains(authorName)){
30             authors.remove(authorName);
31             System.out.println("Author " + authorName + " removed.");
32         }else{
33             System.out.println("Author " + authorName + " not found in the list.");
34         }
35     }
36
37 }
```

The Explorer view on the left shows the project structure with folders like `src`, `bin`, `lib`, `Lab01`, `Lab02`, `Aims`, `hust`, `soict`, `dsai`, `aims`, `media`, `Book.java`, `hustpi`, `Report`, `UML Diagram`, `AIMS.java`, `Lab03`, `Lab04`, `Info.txt`, `README.md`, and `zip`.

3. Tạo lớp trừu tượng Media



The screenshot shows the VS Code editor with the `Media.java` file open. The file contains the following code:

```
1 package Lab02.hust.soict.dsai.aims.media;
2
3 public abstract class Media {
4     private int id;
5     private String title;
6     private String category;
7     private float cost;
8
9     Run|Debug
10     public static void main(String[] args) {
11     }
12
13 }
```

The Explorer view on the left shows the project structure with folders like `src`, `bin`, `lib`, `Lab01`, `Lab02`, `Aims`, `hust`, `soict`, `dsai`, `aims`, `media`, `Book.java`, `hustpi`, `Report`, `UML Diagram`, `AIMS.java`, `Lab03`, `Lab04`, `Info.txt`, `README.md`, and `zip`.

- Thêm các phương thức setter() và getter() cho các thuộc tính

```

src > Lab02 > hant > soict > chai > aims > media > Media.java > Media
1 public abstract class Media {
13     public int getId() {
14         return id;
15     }
16
17     public void setId(int id) {
18         this.id = id;
19     }
20
21     public String getTitle() {
22         return title;
23     }
24
25     public void setTitle(String title) {
26         this.title = title;
27     }
28
29     public String getCategory() {
30         return category;
31     }
32
33     public void setCategory(String category) {
34         this.category = category;
35     }
36
37     public float getCost() {
38         return cost;
39     }
40 }
  
```

- Loại bỏ các thuộc tính và phương thức của lớp Book và DigitalVideoDisc

- Book:

```

src > Lab02 > hant > soict > chai > aims > media > Book.java > Book
1 public class Book extends Media {
2     private List<String> authors = new ArrayList<>();
3
4     //Phương thức thêm tác giả
5     public void addAuthor(String authorName) {
6         if(!authors.contains(authorName)) {
7             authors.add(authorName);
8             System.out.println("author " + authorName + " added.");
9         } else {
10             System.out.println("author " + authorName + " is already in the list.");
11         }
12     }
13
14     //Phương thức xóa tác giả
15     public void removeAuthor(String authorName) {
16         if(authors.contains(authorName)) {
17             authors.remove(authorName);
18             System.out.println("author " + authorName + " removed.");
19         } else {
20             System.out.println("author " + authorName + " not found in the list.");
21         }
22     }
23
24     public Book() {
25     }
26
27     public Book(int id, String title, String category, float cost, String author) {
28         super(id, title, category, cost);
29         this.authors = new ArrayList<>(); // thoi tao lai array list
30         this.authors.add(author); // them author vào list
31     }
32
33     @Override
34     public String toString() {
35         return authors.toString() + String.join(", ", this.authors); // ket hop các danh sách author
36         return "Book - " + this.getTitle() + " - " + authorString + " - " + this.getCost() + " $";
37     }
38 }
  
```

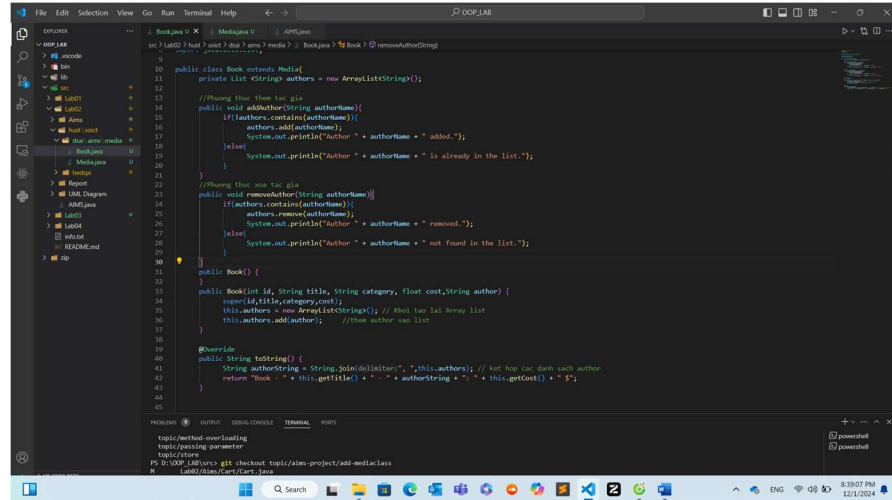
- DigitalVideoDisc:

```

src > Lab02 > aims > disc > DigitalVideoDisc.java > DigitalVideoDisc > equals
1 //Phạm Thành Thuận
2 //2022/2/22
3 //Class "DigitalVideoDisc" source code
4 package Lab02_aims_disc;
5
6 import Lab02_hant_soict_chai_aims_media.Media;
7
8 public class DigitalVideoDisc extends Media {
9     // chỉ hỗ trợ static
10     private static int nDigitalVideoDiscs = 0; // Class attribute cần khai báo
11     private String director;
12     private int length;
13
14     public DigitalVideoDisc() {
15     }
16
17     public DigitalVideoDisc(String title) {
18         setTitle(title);
19         setId(nDigitalVideoDiscs);
20         nDigitalVideoDiscs++;
21     }
22
23     //ghi đè phương thức equals
24     @Override
25     public boolean equals(Object obj) {
26         if(this == obj) return true;
27
28         if(obj == null || getClass() != obj.getClass()) return false;
29         DigitalVideoDisc disc = (DigitalVideoDisc) obj;
30     }
31 }
  
```

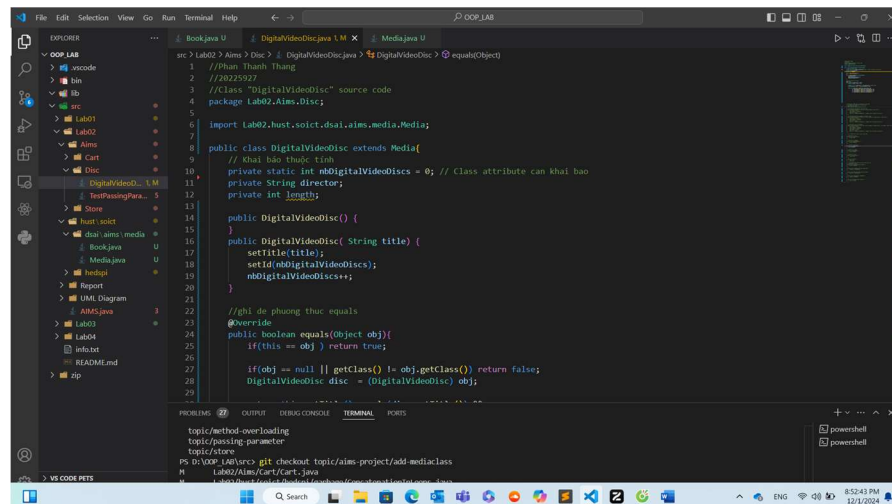
- Cho hai lớp Book và DigitalVideoDisc kế thừa lớp Media

- Book:

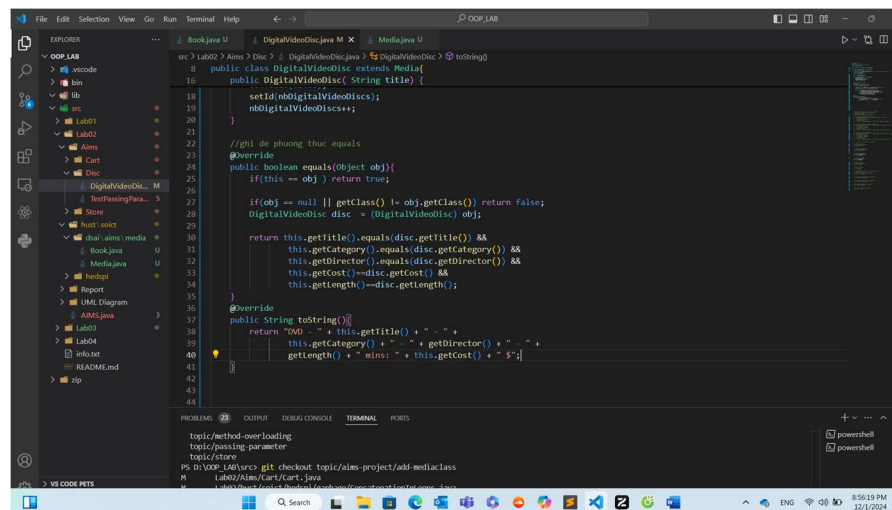


```
10 public class Book extends Media {
11     private List<String> authors = new ArrayList<>();
12
13     //Phương thức thêm tác giả
14     public void addAuthor(String authorName) {
15         if (authors.contains(authorName)) {
16             authors.add(authorName);
17             System.out.println("Author " + authorName + " added.");
18         } else {
19             System.out.println("Author " + authorName + " is already in the list.");
20         }
21     }
22
23     //Phương thức xóa tác giả
24     public void removeAuthor(String authorName) {
25         if (authors.contains(authorName)) {
26             authors.remove(authorName);
27             System.out.println("Author " + authorName + " removed.");
28         } else {
29             System.out.println("Author " + authorName + " not found in the list.");
30         }
31     }
32
33     public Book() {
34     }
35
36     public Book(int id, String title, String category, float cost, String author) {
37         super(id, title, category, cost);
38         this.authors = new ArrayList<>(); // Khởi tạo lại Array List
39         this.authors.add(author); // Thêm tác giả vào list
40     }
41
42     @Override
43     public String toString() {
44         String authorString = String.join(", ", this.authors); // Kết hợp các danh sách tác giả
45         return "Book - " + this.getTitle() + " - " + authorString + " - " + this.getCost() + " $";
46     }
47 }
```

- DigitalVideoDisc:



```
1 //Phan Thuan Thang
2 //2022/2022
3 //Class "DigitalVideoDisc" source code
4 package Lab02.Aims.Disc;
5
6 import Lab02.hust.soict.dial.aims.media.Media;
7
8 public class DigitalVideoDisc extends Media {
9     //Chai hai thuộc tính
10     private static int nbDigitalVideoDiscs = 0; // Class attribute can khai báo
11     private String director;
12     private int length;
13
14     public DigitalVideoDisc() {
15     }
16
17     public DigitalVideoDisc(String title) {
18         setTitle(title);
19         setNbDigitalVideoDiscs();
20     }
21
22     //ghi đè phương thức equals
23     @Override
24     public boolean equals(Object obj) {
25         if (this == obj) return true;
26
27         if (obj == null || getClass() != obj.getClass()) return false;
28         DigitalVideoDisc disc = (DigitalVideoDisc) obj;
29
30         return this.getTitle().equals(disc.getTitle()) &&
31             this.getCategory().equals(disc.getCategory()) &&
32             this.getDirector().equals(disc.getDirector()) &&
33             this.getCost() == disc.getCost() &&
34             this.getLength() == disc.getLength();
35     }
36
37     @Override
38     public String toString() {
39         return "DVD - " + this.getTitle() + " - " +
40             this.getCategory() + " - " + this.getDirector() + " - " +
41             this.getLength() + " mins: " + this.getCost() + " $";
42     }
43 }
```

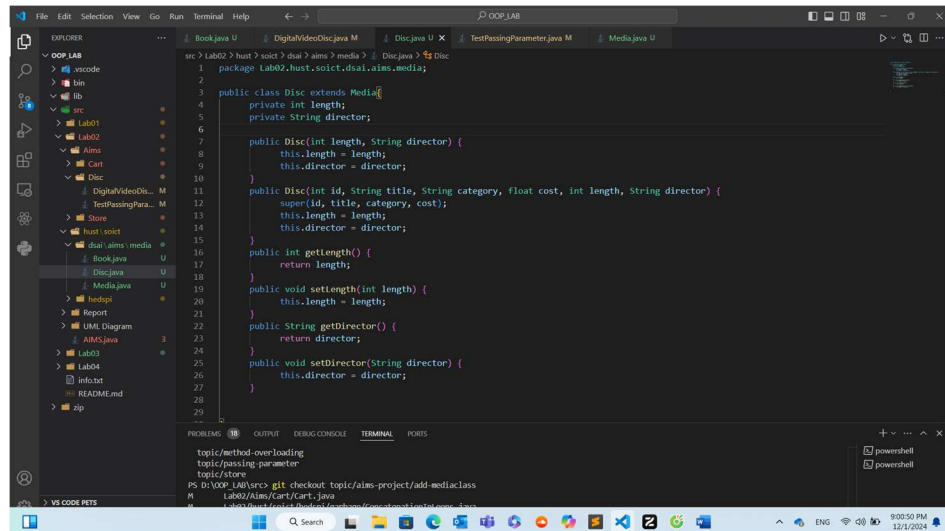


```
8 public class DigitalVideoDisc extends Media {
9     public DigitalVideoDisc(String title) {
10         setTitle(title);
11         setNbDigitalVideoDiscs();
12     }
13
14     //ghi đè phương thức equals
15     @Override
16     public boolean equals(Object obj) {
17         if (this == obj) return true;
18
19         if (obj == null || getClass() != obj.getClass()) return false;
20         DigitalVideoDisc disc = (DigitalVideoDisc) obj;
21
22         return this.getTitle().equals(disc.getTitle()) &&
23             this.getCategory().equals(disc.getCategory()) &&
24             this.getDirector().equals(disc.getDirector()) &&
25             this.getCost() == disc.getCost() &&
26             this.getLength() == disc.getLength();
27     }
28
29     @Override
30     public String toString() {
31         return "DVD - " + this.getTitle() + " - " +
32             this.getCategory() + " - " + this.getDirector() + " - " +
33             this.getLength() + " mins: " + this.getCost() + " $";
34     }
35 }
```

4. Tạo lớp CompactDisc

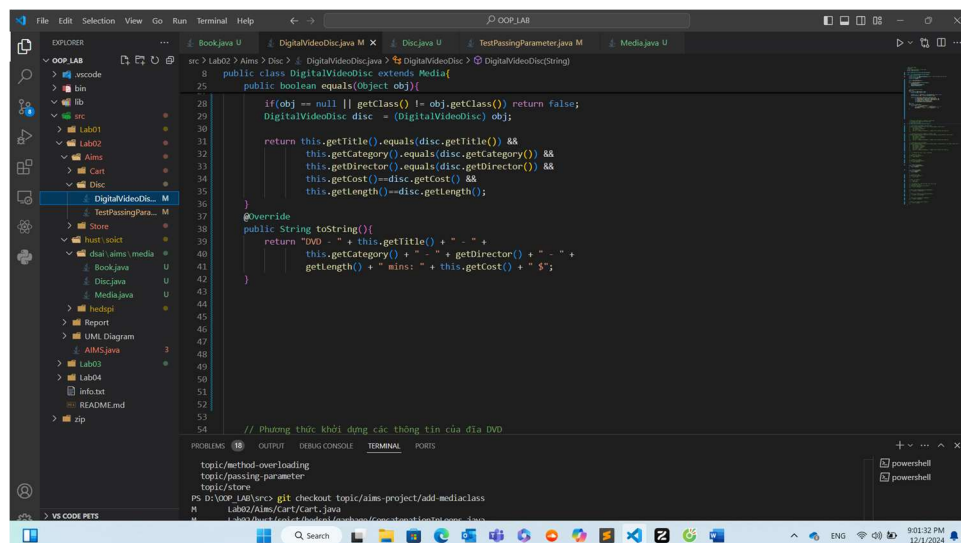
a. Tạo lớp Disc có hai thuộc tính length và director

- Lớp Disc mở rộng lớp Media, có hai thuộc tính mới length, director.
- Tạo Construtor() và phương thức getter() cho hai thuộc tính length, director.



```
1 package Lab02.hust.soiict.dsai.aims.media;
2
3 public class Disc extends Media {
4     private int length;
5     private String director;
6
7     public Disc(int length, String director) {
8         this.length = length;
9         this.director = director;
10    }
11
12    public Disc(int id, String title, String category, float cost, int length, String director) {
13        super(id, title, category, cost);
14        this.length = length;
15        this.director = director;
16    }
17
18    public int getLength() {
19        return length;
20    }
21
22    public void setLength(int length) {
23        this.length = length;
24    }
25
26    public String getDirector() {
27        return director;
28    }
29
30    public void setDirector(String director) {
31        this.director = director;
32    }
33 }
```

- Sửa đổi lớp DigitalVideoDisc để nó mở rộng từ lớp Disc



```
1 public class DigitalVideoDisc extends Disc {
2
3     public boolean equals(Object obj) {
4         if (obj == null || getClass() != obj.getClass()) return false;
5         DigitalVideoDisc disc = (DigitalVideoDisc) obj;
6
7         return this.getTitle().equals(disc.getTitle()) &&
8             this.getCategory().equals(disc.getCategory()) &&
9             this.getDirector().equals(disc.getDirector()) &&
10             this.getCost() == disc.getCost() &&
11             this.getLength() == disc.getLength();
12     }
13
14     @Override
15     public String toString() {
16         return "DVD " + this.getTitle() + " - " +
17             this.getCategory() + " - " + this.getDirector() + " - " +
18             getLength() + " mins: " + this.getCost() + " $";
19     }
20
21     // Phương thức khởi dựng các thông tin của đĩa DVD
22 }
```


- Tạo lớp CompactDisc mở rộng từ lớp Disc.

```

11
12 public class DigitalVideoDisc extends Disc implements Playable {
13     private static int nbDigitalVideoDiscs = 0;
14
15     public DigitalVideoDisc(String title, String category, float cost, int length, String director) {
16         super(nbDigitalVideoDiscs++, title, category, cost, length, director);
17     }
18
19     // Ghi đè phương thức equals
20     @Override
21     public boolean equals(Object obj) {
22         // Kiểm tra nếu đối tượng so sánh chính là đối tượng hiện tại
23         if (this == obj) return true;
24         // Kiểm tra nếu obj là null hoặc không cùng kiểu
25         if (obj == null || getClass() != obj.getClass()) return false;
26         // Ép kiểu obj thành DigitalVideoDisc
27         DigitalVideoDisc disc = (DigitalVideoDisc) obj;
28         // So sánh các thuộc tính
29         return this.getTitle().equals(disc.getTitle()) &&
30             this.getCategory().equals(disc.getCategory()) &&
31             this.getDirector().equals(disc.getDirector()) &&
32             this.getCost() == disc.getCost() &&
33             this.getLength() == disc.getLength();
34     }
35
36     @Override
37     public String toString() {
38         return "DVD - " + this.getTitle() + " - " +
39             this.getCategory() + " - " + this.getDirector() + " - " +
40             this.getCost() + " - " + this.getLength();
41     }
42 }

```

b. Tạo lớp Track

- Các attribute của class Track: title, length
- Tạo phương thức constructor(), getter() cho hai attribute title, length

```

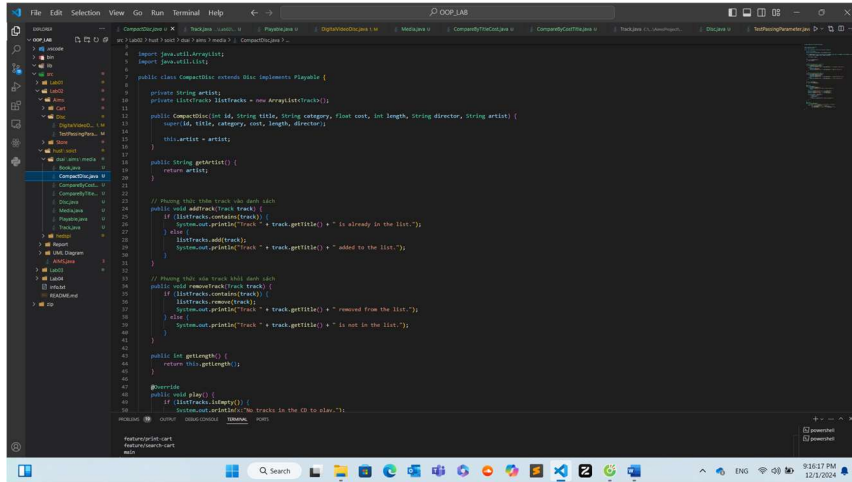
1 package Lab02.hust.soict.dsai.aims.media;
2
3 public class Track implements Playable {
4
5     private String title;
6     private int length;
7
8     public Track(String title, int length) {
9         this.title = title;
10        this.length = length;
11    }
12
13    public String getTitle() {
14        return title;
15    }
16
17    public int getLength() {
18        return length;
19    }
20
21    public void play() {
22        System.out.println("Playing track: " + this.getTitle());
23        System.out.println("Track length: " + this.getLength());
24    }
25
26    @Override
27    public boolean equals(Object obj) {
28        if (this == obj) return true; // Kiểm tra tham chiếu
29        // Kiểm tra nếu obj là null hoặc không cùng kiểu
30        if (obj == null || getClass() != obj.getClass()) return false;
31        // Ép kiểu obj thành Track
32        Track track = (Track) obj;
33        // So sánh các thuộc tính
34        return this.getTitle().equals(track.getTitle()) &&
35            this.getLength() == track.getLength();
36    }
37
38    @Override
39    public String toString() {
40        return "Track - " + this.getTitle() + " - " + this.getLength();
41    }
42 }

```

c. Sửa đổi lớp CompactDisc

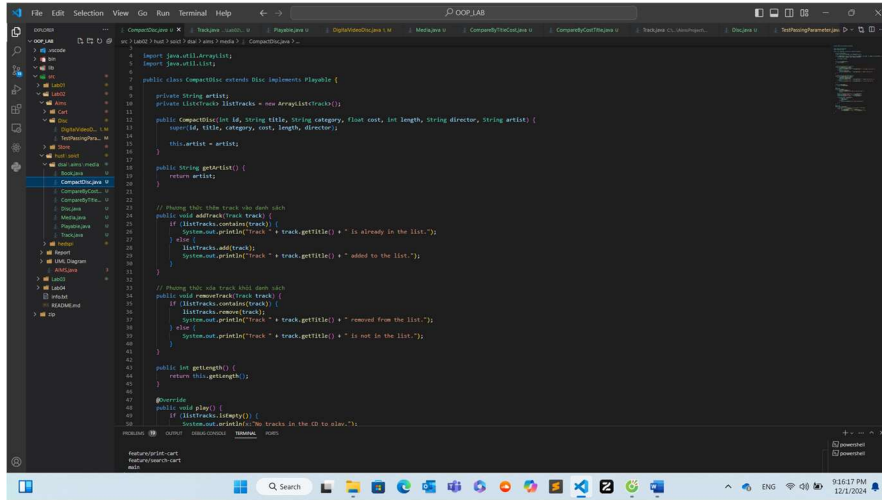
- Thêm hai thuộc tính cho lớp: artist, và một ArrayList kiểu Track để lưu danh sách các bài nhạc.
- Đặt các thuộc tính này là private và tạo phương thức getter cho thuộc tính artist.
- Tạo constructor, sử dụng super()
- Tạo hai phương thức addTrack(), removeTrack()

- Tạo phương thức `getLength()`



5. Tạo interface Playable

- Tạo interface Playable: thêm phương thức play() vào interface



- Triển khai interface Playable cho các lớp CompactDisc, DigitalVideoDisc và Track.

- CompactDisc

```

1  import java.util.ArrayList;
2  import java.util.List;
3
4  public class CompactDisc implements Playable {
5      private String artist;
6      private List<Track> listTracks = new ArrayList<>();
7
8      public CompactDisc(int id, String title, String category, float cost, int length, String director, String artist) {
9          super(id, title, category, cost, length, director);
10         this.artist = artist;
11     }
12
13     public String getArtist() {
14         return artist;
15     }
16
17     // Thêm một đĩa nhạc vào danh sách
18     public void addTrack(Track track) {
19         if (listTracks.contains(track)) {
20             System.out.println("Track " + track.getTitle() + " is already in the list.");
21         } else {
22             listTracks.add(track);
23             System.out.println("Track " + track.getTitle() + " added to the list.");
24         }
25     }
26
27     // Xóa một đĩa nhạc khỏi danh sách
28     public void removeTrack(Track track) {
29         if (listTracks.contains(track)) {
30             listTracks.remove(track);
31             System.out.println("Track " + track.getTitle() + " removed from the list.");
32         } else {
33             System.out.println("Track " + track.getTitle() + " is not in the list.");
34         }
35     }
36
37     public int getLength() {
38         return listTracks.length();
39     }
40
41     @Override
42     public void play() {
43         if (listTracks.isEmpty()) {
44             System.out.println("No tracks in the CD to play.");
45         }
46     }
47
48     @Override
49     public boolean equals(Object obj) {
50         if (obj == null) return false;
51         if (obj.getClass() != this.getClass()) return false;
52         CompactDisc disc = (CompactDisc) obj;
53         return this.getTitle().equals(disc.getTitle()) &&
54                this.getCategory().equals(disc.getCategory()) &&
55                this.getDirector().equals(disc.getDirector()) &&
56                this.getCost() == disc.getCost() &&
57                this.getLength() == disc.getLength();
58     }
59
60     @Override
61     public String toString() {
62         return "CD " + this.getTitle() + " - " +
63                this.getCategory() + " - " +
64                this.getDirector() + " - " +
65                this.getCost() + " $";
66     }
67 }

```

- DigitalVideoDisc

```

1  public class DigitalVideoDisc implements Playable {
2      public boolean equals(Object obj) {
3          if (obj == null) return false;
4          // Kiểm tra nếu obj là null hoặc không cùng kiểu
5          if (obj == null || getClass() != obj.getClass()) return false;
6          // Kiểm tra nếu obj là null hoặc không cùng kiểu
7          DigitalVideoDisc disc = (DigitalVideoDisc) obj;
8          // So sánh các thuộc tính
9          return this.getTitle().equals(disc.getTitle()) &&
10                 this.getCategory().equals(disc.getCategory()) &&
11                 this.getDirector().equals(disc.getDirector()) &&
12                 this.getCost() == disc.getCost() &&
13                 this.getLength() == disc.getLength();
14     }
15
16     @Override
17     public String toString() {
18         return "DVD " + this.getTitle() + " - " +
19                this.getCategory() + " - " +
20                this.getDirector() + " - " +
21                this.getCost() + " $";
22     }
23
24     @Override
25     public void play() {
26         System.out.println("Playing DVD: " + this.getTitle());
27         System.out.println("DVD length: " + this.getLength());
28     }
29 }

```

- Track

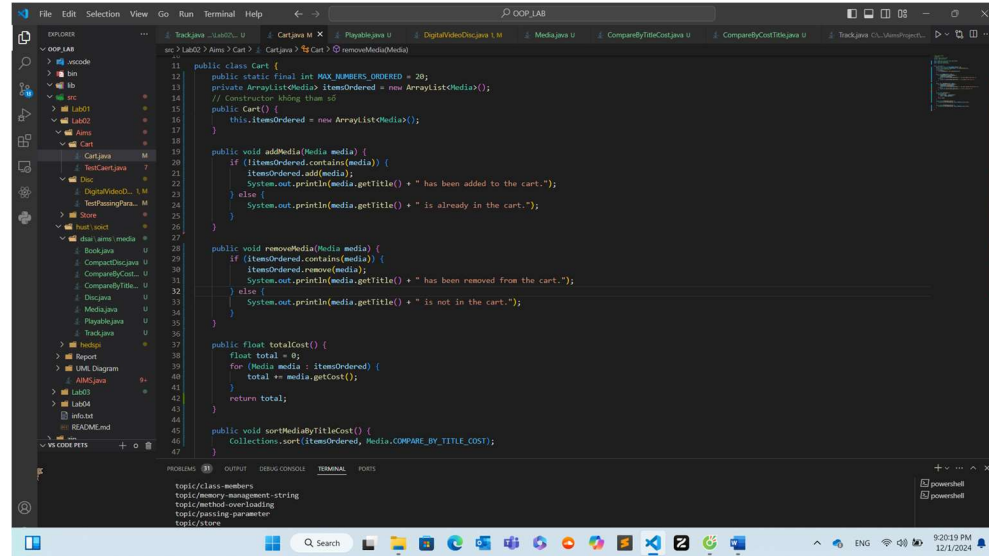
```

1  public class Track implements Playable {
2      private String title;
3      private int length;
4
5      public Track(String title, int length) {
6          this.title = title;
7          this.length = length;
8      }
9
10     public String getTitle() {
11         return title;
12     }
13
14     public int getLength() {
15         return length;
16     }
17
18     public void play() {
19         System.out.println("Playing track: " + this.getTitle());
20         System.out.println("Track length: " + this.getLength());
21     }
22
23     @Override
24     public boolean equals(Object obj) {
25         if (obj == null) return false;
26         // Kiểm tra nếu obj là null hoặc không cùng kiểu
27         if (obj == null || getClass() != obj.getClass()) return false;
28         // Kiểm tra nếu obj là null hoặc không cùng kiểu
29         Track track = (Track) obj;
30         return this.getTitle().equals(track.getTitle()) && this.getLength() == track.getLength();
31     }
32
33     @Override
34     public String toString() {
35         return "Track " + this.getTitle() + " - " +
36                this.getLength() + "s";
37     }
38 }

```

6. Cập nhật lớp Cart để hoạt động với Media

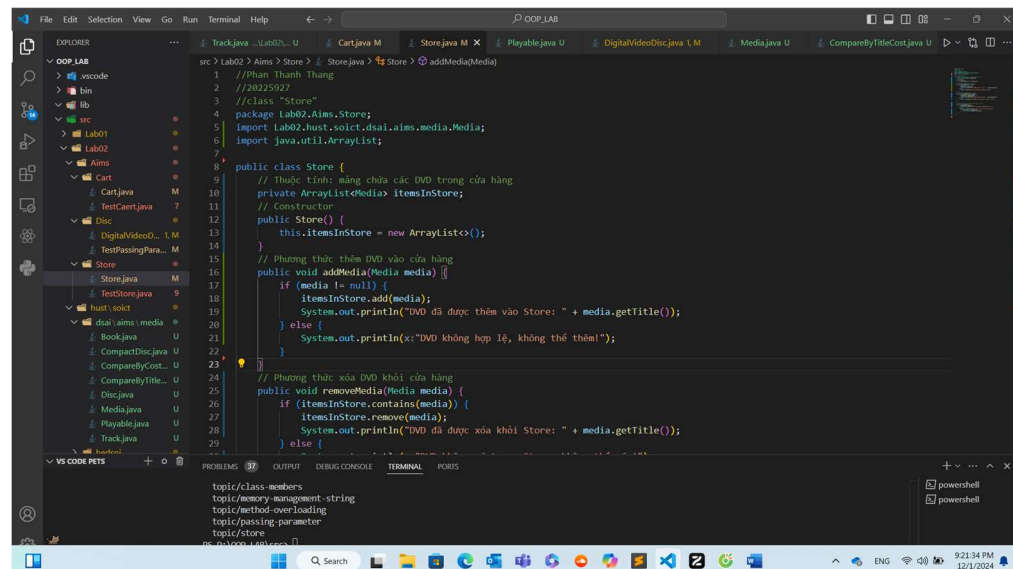
- Thêm các phương thức addMedia(), removeMedia() để quản lý DVD, Book và CompactDisc.



```
11 public class Cart {
12     public static final int MAX_ITEMS_ORDERED = 20;
13     private ArrayList<Media> itemsOrdered = new ArrayList<Media>();
14     // Constructor không tham số
15     public Cart() {
16         this.itemsOrdered = new ArrayList<Media>();
17     }
18
19     public void addMedia(Media media) {
20         if (!itemsOrdered.contains(media)) {
21             itemsOrdered.add(media);
22             System.out.println(media.getTitle() + " has been added to the cart.");
23         } else {
24             System.out.println(media.getTitle() + " is already in the cart.");
25         }
26     }
27
28     public void removeMedia(Media media) {
29         if (itemsOrdered.contains(media)) {
30             itemsOrdered.remove(media);
31             System.out.println(media.getTitle() + " has been removed from the cart.");
32         } else {
33             System.out.println(media.getTitle() + " is not in the cart.");
34         }
35     }
36
37     public float getTotalCost() {
38         float total = 0;
39         for (Media media : itemsOrdered) {
40             total += media.getCost();
41         }
42         return total;
43     }
44
45     public void sortMediaByTitleCost() {
46         Collections.sort(itemsOrdered, Media.COMPARE_BY_TITLE_COST);
47     }
48 }
```

7. Cập nhật lớp Store để hoạt động với Media

- Tương tự như lớp Cart, thay đổi thuộc tính itemsInStore[] của lớp Store thành kiểu ArrayList<Media>.
- Thay thế các phương thức addDigitalVideoDisc() và removeDigitalVideoDisc() bằng các phương thức addMedia() và removeMedia().

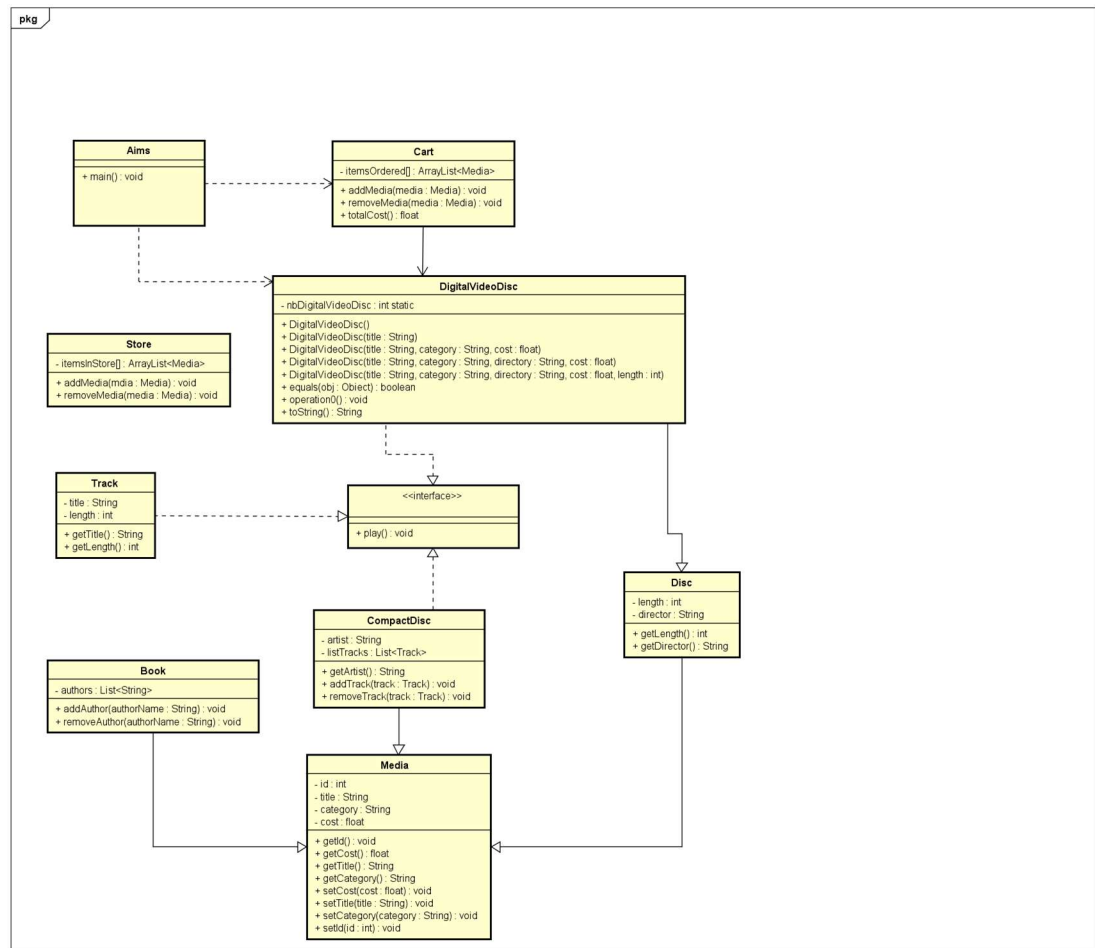


```
1 //Phan Thanh Thang
2 //20225927
3 //class "Store"
4 package Lab02.Aims.Store;
5 import Lab02.hust.soiict.dsai.aims.media.Media;
6 import java.util.ArrayList;
7
8 public class Store {
9     // Thuộc tính: mảng chứa các DVD trong cửa hàng
10     private ArrayList<Media> itemsInStore;
11     // Constructor
12     public Store() {
13         this.itemsInStore = new ArrayList<>();
14     }
15     // Phương thức thêm DVD vào cửa hàng
16     public void addMedia(Media media) {
17         if (media != null) {
18             itemsInStore.add(media);
19             System.out.println("DVD đã được thêm vào Store: " + media.getTitle());
20         } else {
21             System.out.println("DVD không hợp lệ, không thể thêm!");
22         }
23     }
24     // Phương thức xóa DVD khỏi cửa hàng
25     public void removeMedia(Media media) {
26         if (itemsInStore.contains(media)) {
27             itemsInStore.remove(media);
28             System.out.println("DVD đã được xóa khỏi Store: " + media.getTitle());
29         } else {
30
31         }
32     }
33 }
```

8. Constructor của tất cả các lớp và lớp cha

- Cập nhật sơ đồ lớp UML cho dự án AimsProject.

- Tạo nhánh mới đẩy ảnh sơ đồ lớp và gộp vào nhánh master.



9. Unique item in a list

- Thêm phương thức equals cho lớp Media

```

src > Lab02 > hunt > soict > dual > aims > media > Media.java
5 public abstract class Media {
39 public void setTitle(String title) {
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }

public boolean equals(Object obj) {
    if (this == obj) return true; // Kiểm tra tham chiếu
    if (obj == null || getClass() != obj.getClass()) return false; // Kiểm tra kiểu đối tượng
    Media media = (Media) obj; // Ép kiểu
    return this.title.equals(media.title); // So sánh thuộc tính title
}

```

- Thêm phương thức equals cho lớp Track

```

1 public class Track implements Playable {
2
3     public void play() {
4         System.out.println("Playing track: " + this.getTitle());
5         System.out.println("Track length: " + this.getLength());
6     }
7
8     @Override
9     public boolean equals(Object obj) {
10        if (this == obj) return true; // Kiểm tra tham chiếu
11        if (obj == null || getClass() != obj.getClass()) return false; // Kiểm tra kiểu đối tượng
12
13        Track track = (Track) obj; // Ép kiểu
14        return this.title.equals(track.title) && this.length == track.length; // So sánh title và length
15    }
16 }

```

10. Polymorphism with toString() method

- Tạo arrayList của Media, thêm một số sản phẩm (DVD, Book, CD) vào danh sách sau đó duyệt qua từng sản phẩm, sử dụng toString() in ra thông tin của mỗi sản phẩm.

```

1 public class MediaTest {
2
3     public static void main(String[] args) {
4         // Tạo một ArrayList chứa các phương tiện Media
5         ArrayList<Media> mediaList = new ArrayList<Media>();
6
7         // Thêm các phương tiện vào danh sách
8         mediaList.add(new CompactDisc(id1, title:"CD Title", category:"Music", cost:15.99f, length:60, director:"Director", arti:"");
9         mediaList.add(new DigitalVideoDisc(title:"DVD Title", category:"Movies", cost:20.99f, length:120, director:"Director");
10        mediaList.add(new Book(id3, title:"Book Title", category:"Literature", cost:10.99f, author:"Author");
11
12        // Duyệt qua danh sách và in thông tin của các phương tiện
13        for (Media media : mediaList) {
14            System.out.println(media.toString());
15        }
16    }
17 }

```

- Quan sát kết quả và giải thích chi tiết.

```

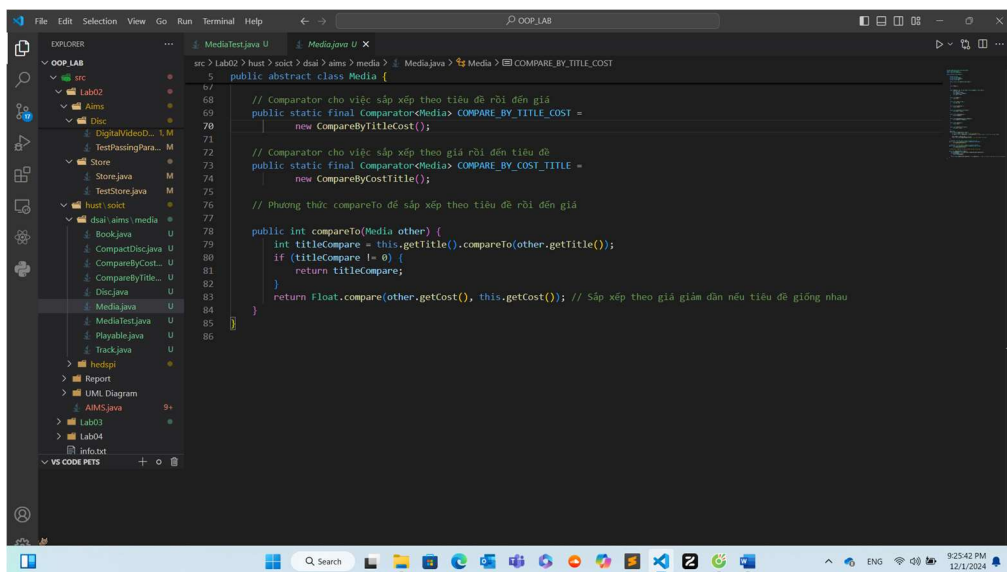
PS D:\VOP-LAB > cd "C:\Program Files\Java\jdk-22\bin\java.exe" "-Xmx4096m" -cp "D:\VOP-LAB\bin" Lab02.Hust.Voicet.Dual.Aims.MediaTest
CD - CD Title - Music - Artist - 60 mins: 15.99 $
DVD - DVD Title - Movies - Director - 120 mins: 20.99 $
Book - Book Title - Author: 10.99 $

```

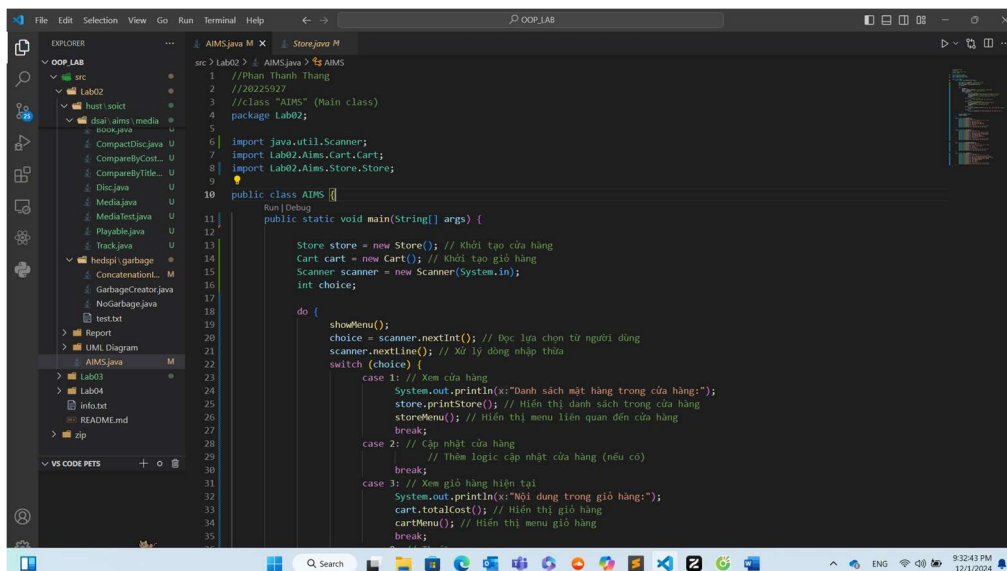

Mặc dù tất cả các lớp CD, DVD, Book đều là lớp con của lớp Media và đều kế thừa phương thức toString() từ lớp cha, mỗi lớp con có thể có một cách triển khai riêng cho phương thức này. Khi duyệt các sản phẩm và sử dụng phương thức toString() cho từng đối tượng, mỗi đối tượng sẽ gọi phương thức toString() tương ứng của nó. Do đó kết quả hiển thị của mỗi sản phẩm là khác nhau.

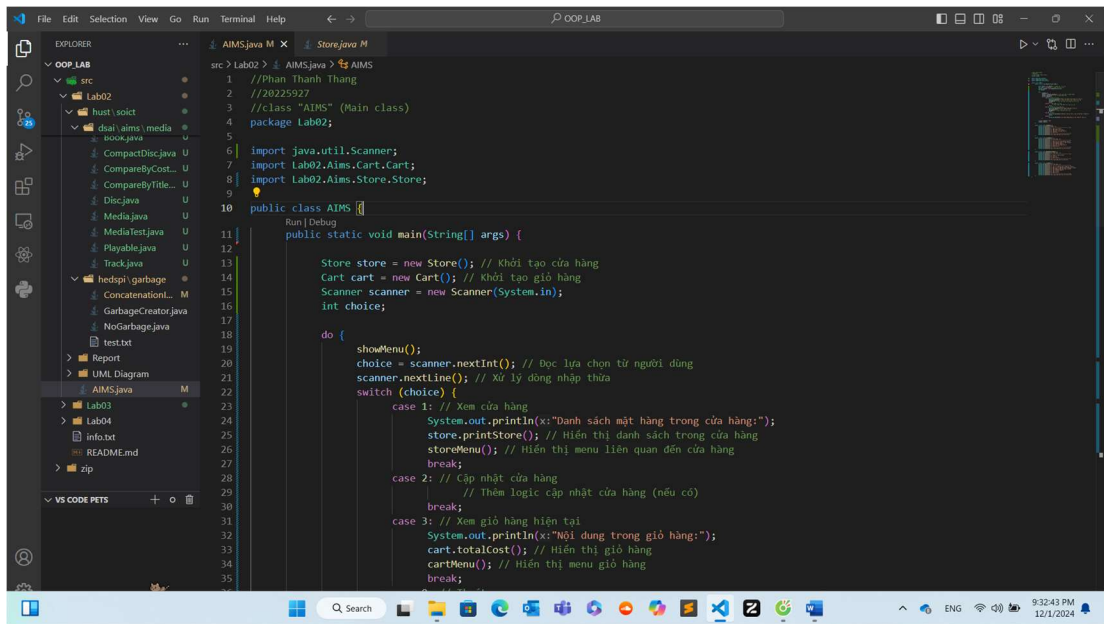
11. Sắp xếp phương tiện trong giỏ hàng

- Sắp xếp theo tiêu đề
- Sắp xếp theo giá
- Định nghĩa trong lớp Media

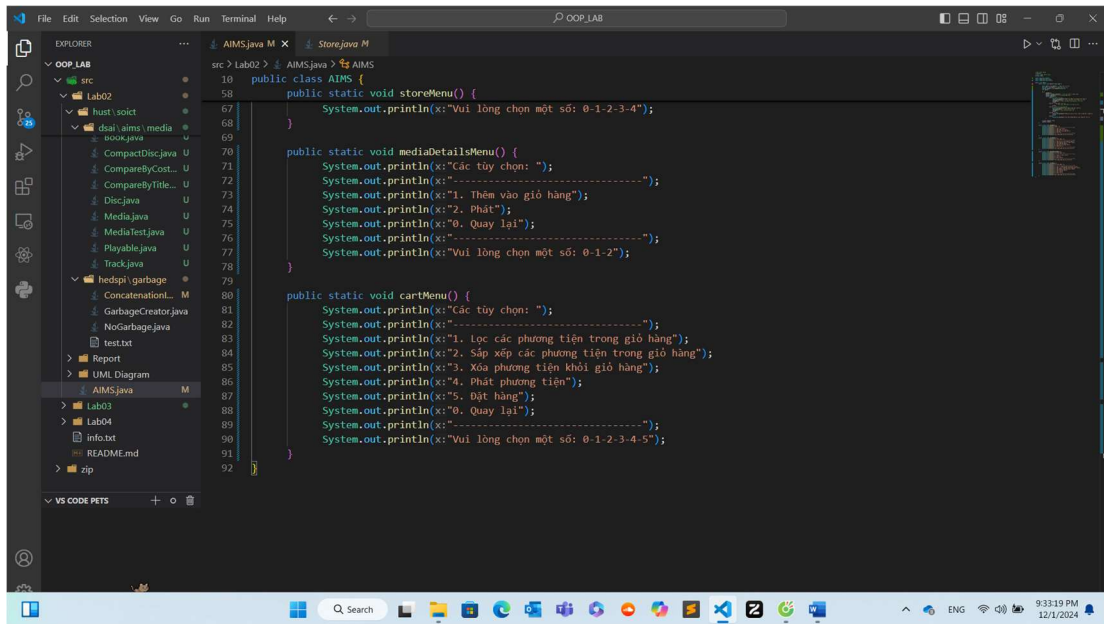


12. Create a complete console application in the Aims class





```
File Edit Selection View Go Run Terminal Help
src > Lab02 > AIMS.java > AIMS
1 //Phan Thanh Thang
2 //20225927
3 //class "AIMS" (Main class)
4 package Lab02;
5
6 import java.util.Scanner;
7 import Lab02.Aims.Cart.Cart;
8 import Lab02.Aims.Store.Store;
9
10 public class AIMS {
11
12     Run | Debug
13     public static void main(String[] args) {
14
15         Store store = new Store(); // Khởi tạo cửa hàng
16         Cart cart = new Cart(); // Khởi tạo giỏ hàng
17         Scanner scanner = new Scanner(System.in);
18         int choice;
19
20         do {
21             showMenu();
22             choice = scanner.nextInt(); // Đọc lựa chọn từ người dùng
23             scanner.nextLine(); // Xử lý dòng nhập thừa
24             switch (choice) {
25                 case 1: // Xem cửa hàng
26                     System.out.println(x:"Danh sách mặt hàng trong cửa hàng:");
27                     store.printStore(); // Hiển thị danh sách trong cửa hàng
28                     storeMenu(); // Hiển thị menu liên quan đến cửa hàng
29                     break;
30                 case 2: // Cập nhật cửa hàng
31                     // Thêm logic cập nhật cửa hàng (nếu có)
32                     break;
33                 case 3: // Xem giỏ hàng hiện tại
34                     System.out.println(x:"Nội dung trong giỏ hàng:");
35                     cart.totalCost(); // Hiển thị giỏ hàng
36                     cartMenu(); // Hiển thị menu giỏ hàng
37                     break;
38             }
39         } while (choice != 0);
40     }
41 }
```



```
File Edit Selection View Go Run Terminal Help
src > Lab02 > AIMS.java > AIMS
10 public class AIMS {
11
12     public static void showMenu() {
13         System.out.println(x:"Vui lòng chọn một số: 0-1-2-3-4");
14     }
15
16     public static void mediaDetailsMenu() {
17         System.out.println(x:"Các tùy chọn: ");
18         System.out.println(x:"-----");
19         System.out.println(x:"1. Thêm vào giỏ hàng");
20         System.out.println(x:"2. Phát");
21         System.out.println(x:"0. Quay lại");
22         System.out.println(x:"-----");
23         System.out.println(x:"Vui lòng chọn một số: 0-1-2");
24     }
25
26     public static void cartMenu() {
27         System.out.println(x:"Các tùy chọn: ");
28         System.out.println(x:"-----");
29         System.out.println(x:"1. Lọc các phương tiện trong giỏ hàng");
30         System.out.println(x:"2. Sắp xếp các phương tiện trong giỏ hàng");
31         System.out.println(x:"3. Xóa phương tiện khỏi giỏ hàng");
32         System.out.println(x:"4. Phát phương tiện");
33         System.out.println(x:"5. Đặt hàng");
34         System.out.println(x:"0. Quay lại");
35         System.out.println(x:"-----");
36         System.out.println(x:"Vui lòng chọn một số: 0-1-2-3-4-5");
37     }
38 }
```