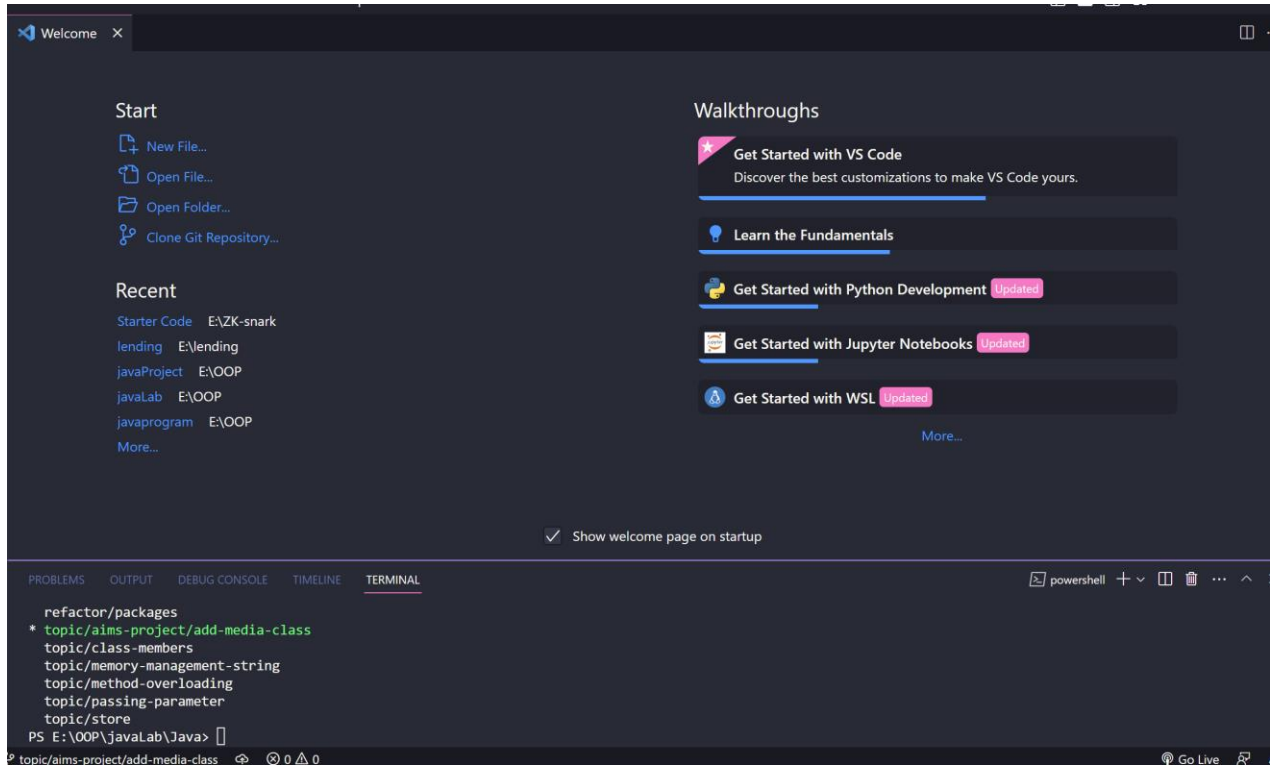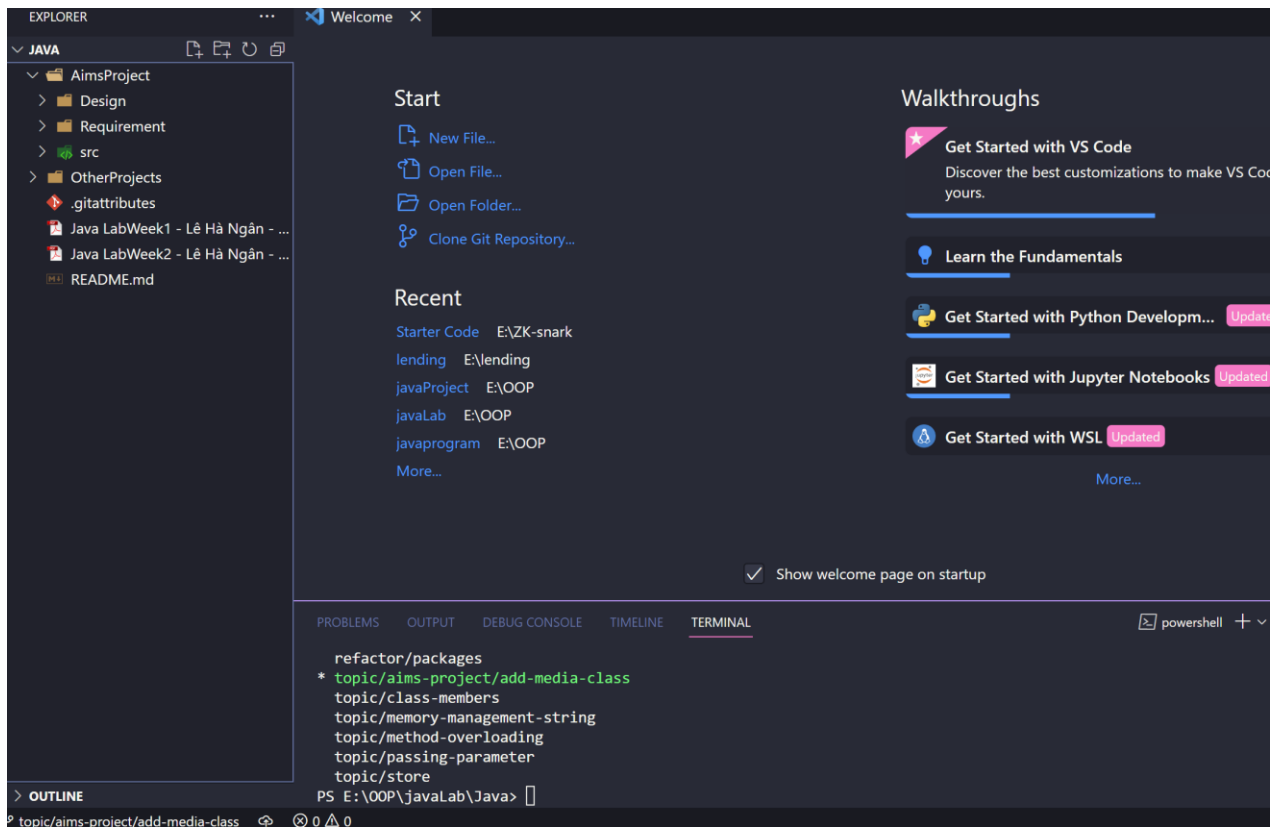# Lab 04: Inheritance and Polymorphism

Lê Hà Ngân - 20215230
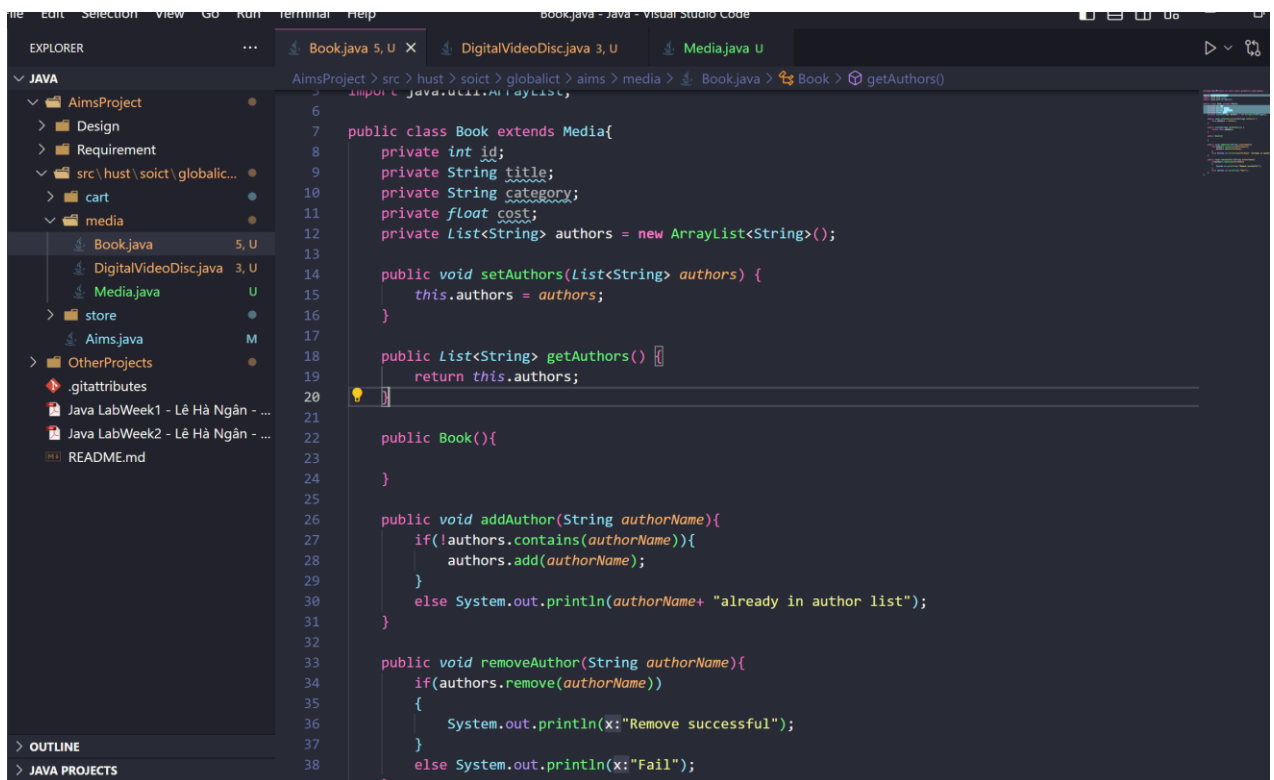
1. Import the existing project into the workspace of Eclipse

2. Additional requirements of AIMS
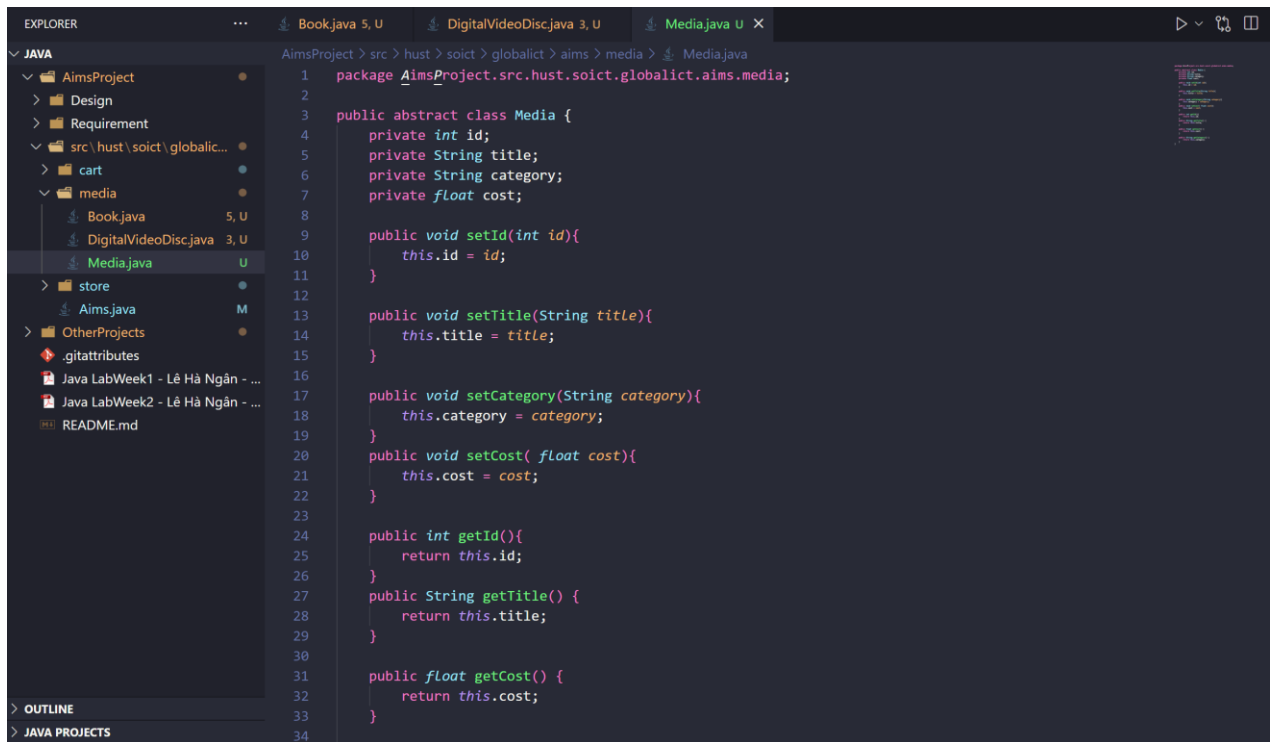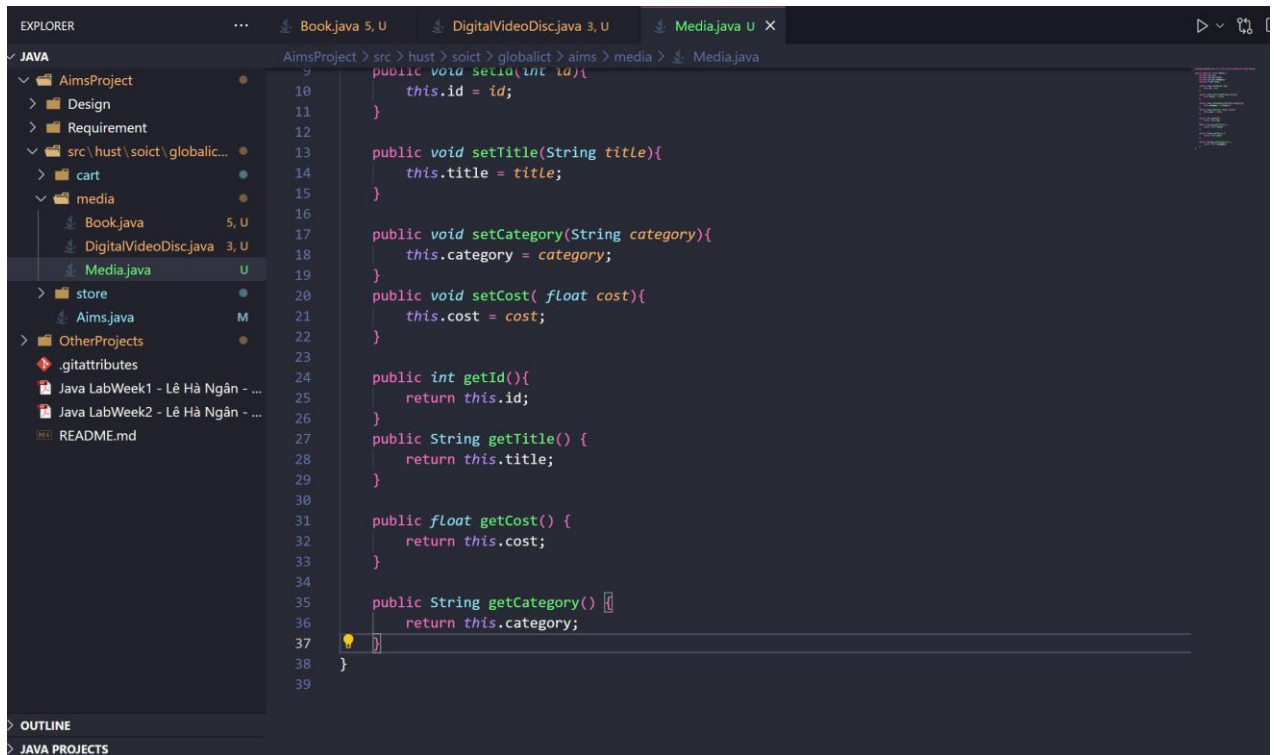## 3. Creating the Book class


```java
import java.util.ArrayList;

public class Book extends Media{
    private int id;
    private String title;
    private String category;
    private float cost;
    private List<String> authors = new ArrayList<String>();

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

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

    public Book(){

    }

    public void addAuthor(String authorName){
        if(!authors.contains(authorName)){
            authors.add(authorName);
        }
        else System.out.println(authorName+ "already in author list");
    }

    public void removeAuthor(String authorName){
        if(authors.remove(authorName))
        {
            System.out.println(x:"Remove successful");
        }
        else System.out.println(x:"Fail");
```

4. Creating the abstract Media class


```java
package AimsProject.src.hust.soict.globalict.aims.media;

public abstract class Media {
    private int id;
    private String title;
    private String category;
    private float cost;

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

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

    public void setCategory(String category){
        this.category = category;
    }
    public void setCost( float cost){
        this.cost = cost;
    }

    public int getId(){
        return this.id;
    }
    public String getTitle() {
        return this.title;
    }

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



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

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

    public void setCategory(String category){
        this.category = category;
    }
    public void setCost( float cost){
        this.cost = cost;
    }

    public int getId(){
        return this.id;
    }
    public String getTitle() {
        return this.title;
    }

    public float getCost() {
        return this.cost;
    }

    public String getCategory() {
        return this.category;
    }
}
```

```java
public class Book extends Media{
    private int id;
    private String title;
    private String category;
    private float cost;
    private List<String> authors = new ArrayList<String>();

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

```java
package AimsProject.src.hust.soict.globalict.aims.media;

public class DigitalVideoDisc extends Media{
    private String title;
    private String category;
    private String director;
    private int length;
    private float cost;
    private int id;

    private static int nbDigitalVideoDiscs = 0;
```

5. **Creating the CompactDisc class**

```java
package AimsProject.src.hust.soict.globalict.aims.media;

import java.util.ArrayList;

public class CompactDisc extends Disc{
    private String artist;
    private ArrayList<Track> tracks = new ArrayList<Track>();

    public void setArtist(String artist){
        this.artist = artist;
    }

    public String getArtist(){
        return this.artist;
    }
    public CompactDisc(String title){
        super(title);
    }

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

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

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

    public void addTrack(Track trackName){
```

```java
27        }
28
29        public CompactDisc(String title, String category, String director, int Length, float cost)
30        {
31            super(title, category, director, length, cost);
32        }
33
34        public void addTrack(Track trackName){
35            if(!tracks.contains(trackName)){
36                tracks.add(trackName);
37            }
38            else System.out.println(trackName+ "already in track list");
39        }
40
41        public void removeTrack(Track trackName){
42            if(tracks.remove(trackName))
43            {
44                System.out.println(x:"Remove successful");
45            }
46            else System.out.println(x:"Fail remove");
47        }
48
49        public int getLength() {
50            int trackLength = 0;
51            for (Track t : tracks) {
52                trackLength += t.getLength();
53            }
54            return trackLength;
55        }
56    }
57
```

```java
package AimsProject.src.hust.soict.globalict.aims.media;

public class Disc extends Media{
    private int length;
    private String director;

    public int getLength(){
        return this.length;
    }

    public String getDirector(){
        return this.director;
    }

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

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

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

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

```java
package AimsProject.src.hust.soict.globalict.aims.media;

public class Track {
    private String title;
    private int length;

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

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

    public String getTitle(){
        return this.title;
    }

    public int getLength(){
        return this.length;
    }

    public Track(String title){
        this.setTitle(title);
    }

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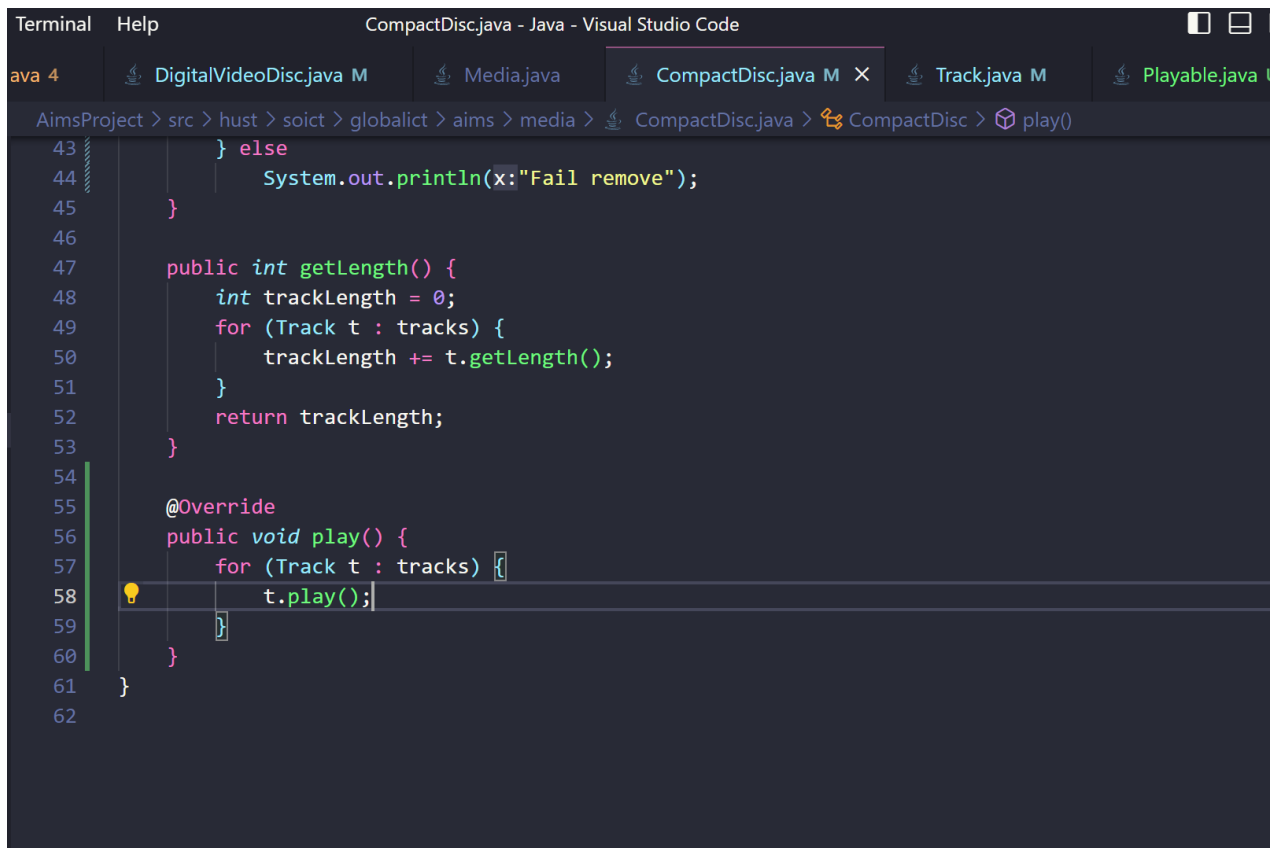```

## 6. Create the Playable interface

AimsProject > src > hust > soict > globalict > aims > media > Playable.java > Playable

```java
package AimsProject.src.hust.soict.globalict.aims.media;

public interface Playable {
    public void play();
}
```

```java
19          }
20
21          public DigitalVideoDisc(String title, String category, String diretor, int length, float c
22              super(title, category, diretor, length, cost);
23              nbDigitalVideoDisc++;
24          }
25
26          public boolean isMatch(String title) {
27              return (this.getTitle() == title);
28          }
29
30          public static int getnbDigitalVideoDisc() {
31              return nbDigitalVideoDisc;
32          }
33
34          public String toString() {
35              return getTitle() + " - " + getCategory() + " - " + getDirector() + " - " + getLength(
36                      + "$";
37          }
38
39          @Override
40          public void play() {
41              System.out.println("Playing DVD: " + this.getTitle());
42              System.out.println("DVD length: " + this.getLength());
43          }
44      }
45
```

AimsProject > src > hust > soict > globalict > aims > media > CompactDisc.java > CompactDisc > play()

```java
43              } else
44                  System.out.println(x:"Fail remove");
45          }
46
47      public int getLength() {
48          int trackLength = 0;
49          for (Track t : tracks) {
50              trackLength += t.getLength();
51          }
52          return trackLength;
53      }
54
55      @Override
56      public void play() {
57          for (Track t : tracks) {
58              t.play();
59          }
60      }
61  }
62
```

7. **Update the Cart class to work with Media**

AimsProject > src > hust > soict > globalict > aims > cart > Cart.java > Cart

```java
1    package AimsProject.src.hust.soict.globalict.aims.cart;
2
3    import AimsProject.src.hust.soict.globalict.aims.media.Media;
4
5    import java.util.ArrayList;
6
7    public class Cart {
8        public static final int MAX_NUMBERS_ORDERED = 20;
9        private float totalCost = 0f;
10       private ArrayList<Media> itemsOrdered = new ArrayList<Media> ();
11
12       public boolean isFulled() {
13           if (itemsOrdered.size() == MAX_NUMBERS_ORDERED)
14               return true;
15           else
16               return false;
17       }
18
19       public void addMedia(Media[] medias) {
20           if (isFulled())
21               System.out.println(x:"The Cart is almost full");
22           for (Media i:medias){
23               itemsOrdered.add(i);
24               if (isFulled()){
25                   System.out.println(x:"The Cart almost full");
26                   break;
27               }
28           }
29       }
30
31
32       public void addMedia(Media media1, Media media2){
33           if (isFulled())
34               System.out.println(x:"The Cart is almost full");
```

```java
27          }
28        }
29      }
30
31
32      public void addMedia(Media media1, Media media2){
33          if (isFulled())
34              System.out.println(x:"The Cart is almost full");
35          itemsOrdered.add(media1);
36          if (isFulled())
37              System.out.println(x:"The Cart is full, can't input media 2");
38          itemsOrdered.add(media2);
39          System.out.println(x:"2 media have been added");
40      }
41
42      public void addMedia(Media media) {
43          if (isFulled())
44              System.out.println(x:"The Cart is almost full");
45          itemsOrdered.add(media);
46          System.out.println(x:"The disc has been added");
47      }
48
49      public void removeMedia(Media media) {
50          if(itemsOrdered.remove(media)) {
51              System.out.println(x:"The media has been removed");
52          } else System.out.println(x:"Not have this media");
53      }
54
55      public float getTotalCost() {
56          for (Media i: itemsOrdered) {
57              totalCost += i.getCost();
58          }
59          return totalCost;
60      }
```

8. Update the Store class to work with Media

```java
package AimsProject.src.hust.soict.globalict.aims.store;

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

import AimsProject.src.hust.soict.globalict.aims.media.Media;

public class Store {
    private List<Media> itemsInStore = new ArrayList<Media>();

    public void addMedia(Media media) {
        itemsInStore.add(media);
        System.out.println(x:"The media has been added");
    }

    public void removeMedia(Media media) {
        if(itemsInStore.remove(media)) {
            System.out.println(x:"The media has been removed");
        } else System.out.println(x:"Not have this media in store");
    }
}
```

9. **Constructors of whole classes and parent classes**
10. **Unique item in a list**

Book.java 4   DigitalVideoDisc.java   Media.java M ✕   Track.java M

AimsProject > src > hust > soict > globalict > aims > media > Media.java > Media >

```java
59
60      @Override
61      public boolean equals(Object obj) {
62          if (obj == this)
63              return true;
64          if (!(obj instanceof Media))
65              return false;
66          Media media = (Media) obj;
67          return media.getTitle() == this.title;
68      }
69
70  }
```

```java
37          }
38
39          @Override
40          public boolean equals(Object obj){
41              if( obj == this )
42                  return true;
43              if(!(obj instanceof Track))
44                  return false;
45              Track track = (Track) obj;
46              return this.title == track.getTitle() && this.length == track.getLength();
47          }
48
49  }
50
```

11. Polymorphism with toString() method

```java
38          }
39
40          @Override
41          public String toString(){
42              return  getTitle() + " - " + getCategory() + " - " +  getAuthors() + ": " + getCost() + "$" ;
43          }
44  }
45
```

```java
27      public static int getnbDigitalVideoDisc() {
28          return nbDigitalVideoDisc;
29      }
30
31      public String toString() {
32          return getTitle() + " - " + getCategory() + " - " + getDirector() + " - " + getLength() + ": " + getCost()
33              + "$";
34      }
35
```

MediaComparatorByCostTitle.java M ✕   answerLab4.txt U   Track.java M   Store.java M   CompactDisc.java M ✕   Cart.java M   Aims.java 1, M

AimsProject > src > hust > soict > globalict > aims > media > CompactDisc.java > CompactDisc > toString()

```java
59          }
60      }
61
62      @Override
63      public String toString(){
64          return getId() + " - " + getTitle() + " - " + getCategory() + " - " + getArtist() + " - " + getDirector() + ": " + getCost()
65      }
66  }
67
```

12. Sort media in the cart

Book.java M   DigitalVideoDisc.java M   Media.java M   MediaComparatorByCostTitle.java M ✕   Disc.java

AimsProject > src > hust > soict > globalict > aims > media > MediaComparatorByCostTitle.java > MediaComparatorByCostTitle > c

```java
package AimsProject.src.hust.soict.globalict.aims.media;

import java.util.Comparator;

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

DigitalVideoDisc.java M   Media.java M   MediaComparatorByCostTitle.java M   MediaComparatorByTitleCost.java ✕

AimsProject > src > hust > soict > globalict > aims > media > MediaComparatorByTitleCost.java > MediaComparatorByTitleCost > compa

```java
package AimsProject.src.hust.soict.globalict.aims.media;

import java.util.Comparator;

public class MediaComparatorByTitleCost implements Comparator<Media> {

    @Override
    public int compare(Media m1, Media m2) {
        int compare = m1.getTitle().compareTo(m2.getTitle());
        if (compare != 0)
            return compare;
        return Double.compare(m1.getCost(), m2.getCost());
    }
}
```

```java
    public void sortByTitleCost(){
        Collections.sort( itemsOrdered, Media.COMPARE_BY_TITLE_COST);
    }

    public void sortByCostTitle(){
        Collections.sort(itemsOrdered, Media.COMPARE_BY_COST_TITLE);
    }
```

13. Create a complete console application in the Aims class

```java
public class Aims {
    public static Cart cart = new Cart();
    public static Store store = new Store();

    public static void showMenu() {
        System.out.println(x:"AIMS: ");
        System.out.println(x:"--------------------------------");
        System.out.println(x:"1. View store");
        System.out.println(x:"2. Update store");
        System.out.println(x:"3. See current cart");
        System.out.println(x:"0. Exit");
        System.out.println(x:"--------------------------------");
        System.out.println(x:"Please choose a number: 0-1-2-3");
        Scanner keyboard = new Scanner(System.in);
        int input = keyboard.nextInt();
        if (input == 1) {
            storeMenu();
        } else if (input == 2) {
            updateStoreOptions();
        } else if (input == 3) {
            cartMenu();
        } else if (input == 0) {
            System.out.println(x:"Exit");
        } else {
            System.out.println(x:"Please choose a number: 0-1-2-3");
            showMenu();
        }
        keyboard.close();
    }

    public static void updateStoreOptions() {
        DigitalVideoDisc dvd1 = new DigitalVideoDisc(title:"The Lion King", category:"Animation", diret
```

```java
43          public static void updateStoreOptions() {
44              DigitalVideoDisc dvd1 = new DigitalVideoDisc(title:"The Lion King", category:"Animation", dire
45              DigitalVideoDisc dvd2 = new DigitalVideoDisc(title:"Star Wars", category:"Science Fiction", di
46              Book book1 = new Book(title:"The Father", category:"Drama", cost:97);
47              CompactDisc cd1 = new CompactDisc(title:"La La Land", category:"Romance, Musical", cost:19.66
48
49              System.out.println(x:"UPDATE STORE MENU OPTIONS:");
50              System.out.println(x:"------------------------------------");
51              System.out.println(x:"1.Add a Media");
52              System.out.println(x:"2.Remove a Media");
53              System.out.println(x:"0.Back");
54              System.out.println(x:"------------------------------------");
55              System.out.println(x:"Please choose number: 0-1-2");
56              Scanner keyboard = new Scanner(System.in);
57              int input = keyboard.nextInt();
58              if (input == 1) {
59                  store.addMedia(dvd1);
60                  store.addMedia(dvd2);
61                  store.addMedia(cd1);
62                  store.addMedia(book1);
63                  updateStoreOptions();
64              } else if (input == 2) {
65                  store.removeMedia(cd1);
66                  updateStoreOptions();
67              } else if (input == 0) {
68                  System.out.println(x:"Back");
69                  showMenu();
70              } else {
71                  System.out.println(x:"Please choose a number: 0-1-2");
72                  updateStoreOptions();
73              }
74              keyboard.close();
75          }
76
```