BÁO CÁO THỰC HÀNH LAB 4
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG
LỚP: 143577 (LT) – 732870 (TN)

Table of Contents

1. Import the existing project into the workspace of Eclipse

2. Additional requirements of AIMS

3. Creating the Book class

Book.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;
import java.util.*;

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

  }

  public Book(int id, String title, String category, float cost, List<String> authors) {

  }

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

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

  public void addAuthor(String authorName) {
      if (!authors.contains(authorName)) {
          authors.add(authorName);
          System.out.println(authorName+" has been added to the list");
      }
      System.out.println(authorName+" is already in the list");
  }

  public void removeAuthor(String authorName) {
    if (!authors.contains(authorName)) {
      System.out.println(authorName+" is not in the list");
    }
    else {
      authors.remove(authorName);
      System.out.println(authorName+" has been removed from the list");
    }
  }
}
```

4. Creating the abstract Media class

Media.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;

import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;

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

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

  public int getId() { return id; }
  public void setId(int id) { this.id = id; }
  public String getTitle() { return title; }
  public void setTitle(String title) { this.title = title; }
  public String getCategory() { return category; }
  public void setCategory(String category) {this.category = category;}
  public float getCost() { return cost; }
  public void setCost(float cost) { this.cost = cost; }

}
```

Modify Book and DigitalVideoDisc classes to extend Media

Book.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;
import java.util.*;
public class Book extends Media {
  private List<String> authors = new ArrayList<String>();
  public Book(int id, String title, String category, float cost) {
    super(id, title, category, cost);
  }
  public Book(int id, String title, String category, float cost, List<String> authors) {
    super(id, title, category, cost);
    this.authors = authors;
  }
  public List<String> getAuthors() { return authors; }
  public void setAuthors(List<String> authors) { this.authors = authors; }
  public void addAuthor(String authorName) {
      if (!authors.contains(authorName)) {
          authors.add(authorName);
          System.out.println(authorName+" has been added to the list.");
      }
      System.out.println(authorName+" is already in the list.");
  }

  public void removeAuthor(String authorName) {
    if (!authors.contains(authorName)) {
      System.out.println(authorName+" is not in the list.");
    }
    else {
      authors.remove(authorName);
      System.out.println(authorName+" has been removed from the list.");
    }
  }
  public String toString() {
    return "ID - "+ getId()+ ". Book - " + getTitle() + " - " + getCategory() + " - " +
getAuthors() + " - $" + getCost();
  }
}
```

5-6. Creating the CompactDisc class and the Playable interface

Create the Disc class

Disc.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;
public class Disc extends Media {
  private int length;
  private String director;
  public Disc(int id, String title, String category, float cost) {
    super(id,title, category, cost);
  }
  public Disc(int id, String title, String category, float cost, int length, String director){
    super(id, title, category, cost);
    this.length = length;
    this.director = director;
  }

  public int getLength() { return length; }
  public void setLength(int length) { this.length = length; }
  public String getDirector() { return director; }
  public void setDirector(String director) { this.director = director; }

}
```

Create the Track class

Track.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;

public class Track extends CompactDisc implements Playable {
  private String titleTrack;
  private int length;

  public Track(int id, String title, String category, float cost, String titleTrack, int
length) {
    super(id, title, category, cost);
    this.titleTrack = titleTrack;
    this.length = length;
  }

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

  public String getTitle() { return titleTrack; }
  public void setTitle(String title) { this.titleTrack = title; }
  public int getLength() { return length; }
  public void setLength(int length) { this.length = length; }
  public boolean equals(Object obj) {
    if (obj instanceof Track) {
      Track track = (Track) obj;
      if (this.titleTrack.equals(track.titleTrack) && this.length == track.length) {
        return true;
      }
    }
    return false;
  }
}
```

Create the CompactDisc class

CompactDisc.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;
import java.util.ArrayList;

public class CompactDisc extends Disc implements Playable {
  private String artist;
  private ArrayList<Track> tracks = new ArrayList<Track>();
  public CompactDisc(int id, String title, String category, float cost) {
    super(id, title, category, cost);
  }
  public CompactDisc(int id, String title, String category, float cost, String artist,
ArrayList<Track> tracks) {
    super(id, title, category, cost);
    this.artist = artist;
    this.tracks = tracks;
  }
  public void play() {
    System.out.println("Playing CD: " + this.getTitle());
    System.out.println("CD length: " + this.getLength());
    for (Track track : tracks) track.play();
  }

  public void addTrack(Track track) {
    if(!tracks.contains(track)) {
      tracks.add(track);
      System.out.println(track+" has been added");
    } System.out.println(track+" is already in the track list");
  }
  public void removeTrack(Track track) {
    if (!tracks.contains(track)) System.out.println(track+" is not in the track list");
    else {
      tracks.remove(track);
      System.out.println(track+" has been removed from the track list");
    }
  }
  public int getLength() {
        int totalLength = 0;
        for (Track track : tracks) totalLength += track.getLength();
        return totalLength;
    }
  public String getArtist() { return artist; }
  public void setArtist(String artist) { this.artist = artist; }
  public String toString() {
    return "ID - "+getId()+ ". CD - " + getTitle() + " - " + getCategory() + " - " +
getDirector() + " - " + getArtist() + " - " + getLength() + "s - $" + getCost();
  }
}
```

Create the Playable interface

PLayable.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;

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

Modify the DigitalVideoDisc class

DigitalVideoDisc.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;

public class DigitalVideoDisc extends Disc implements Playable {
  public DigitalVideoDisc(int id, String title, String category, float cost) {
    super(id, title, category, cost);
  }

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

  //toString method
  public String toString() {
    return "ID - " +getId()+ ". DVD - " + getTitle() + " - " + getCategory() + " - " +
getDirector() + " - " + getLength() + "s - $" + getCost();
  }
}
```

7. Update the Cart class to work with Media

Cart.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.cart;
import java.util.ArrayList;
import hust.soict.hedspi.aims.media.Media;

public class Cart {
  private final int MAX_NUMBER_ORDERED = 20;
  private ArrayList<Media> itemsOrdered = new ArrayList<Media>(MAX_NUMBER_ORDERED);
  public void addMedia(Media media) {
    if (itemsOrdered.size()<MAX_NUMBER_ORDERED) {
      itemsOrdered.add(media);
      System.out.println(media.getTitle()+" has been added to the cart.");
    } else System.out.println("The cart is full.");
  }
  public void removeMedia(Media media) {
      boolean found = false;
      for (Media item : itemsOrdered) {
          if (item.equals(media)) {
              itemsOrdered.remove(item);
              System.out.println(media.getTitle() + " has been removed from the cart.");
              found = true;
              break;
          }
      }
      if (!found) System.out.println(media.getTitle() + " is not in the cart.");
  }
  public ArrayList<Media> getItemsInCart(){ return itemsOrdered; }
  //Get total cost
  public float totalCost() {
        float total = 0.0f;
        for (int i = 0; i < itemsOrdered.size(); i++) {
            total += itemsOrdered.get(i).getCost();
        }
        return total;
    }
  //Print cart method
  public void printCart() {
    System.out.println("**********************CART**********************");
    System.out.println("Ordered Items:");
    for (int i = 0 ; i < itemsOrdered.size(); i++) {
      System.out.println((i+1) + itemsOrdered.get(i).toString());
    }
    System.out.println("Total cost: $" + totalCost());
    System.out.println("************************************************");
  }

  //Search for DVDs in the cart by ID and display them
  //Notify to user if no match is found
  public void searchID(int id){
    boolean found = false;
    for (int i = 0; i < itemsOrdered.size(); i++) {
      if (itemsOrdered.get(i).getId() == id) {
        found = true;
        System.out.println("DVD found: ");
        System.out.println(itemsOrdered.get(i).toString());
        break;
      }
    }
    if (!found) System.out.println("No match is found!");
  }
  //Search for DVDs in the cart by title and display them
  //Notify to user if no match is found
  public void searchTitle(String title){
    boolean found = false;
    for (int i = 0; i < itemsOrdered.size(); i++) {
      if (itemsOrdered.get(i).getTitle().equals(title)) {
        found = true;
        System.out.println("DVD found: ");
        System.out.println(itemsOrdered.get(i).toString());
        break;
      }
    }
    if (!found) System.out.println("No match is found!");
  }
}
```

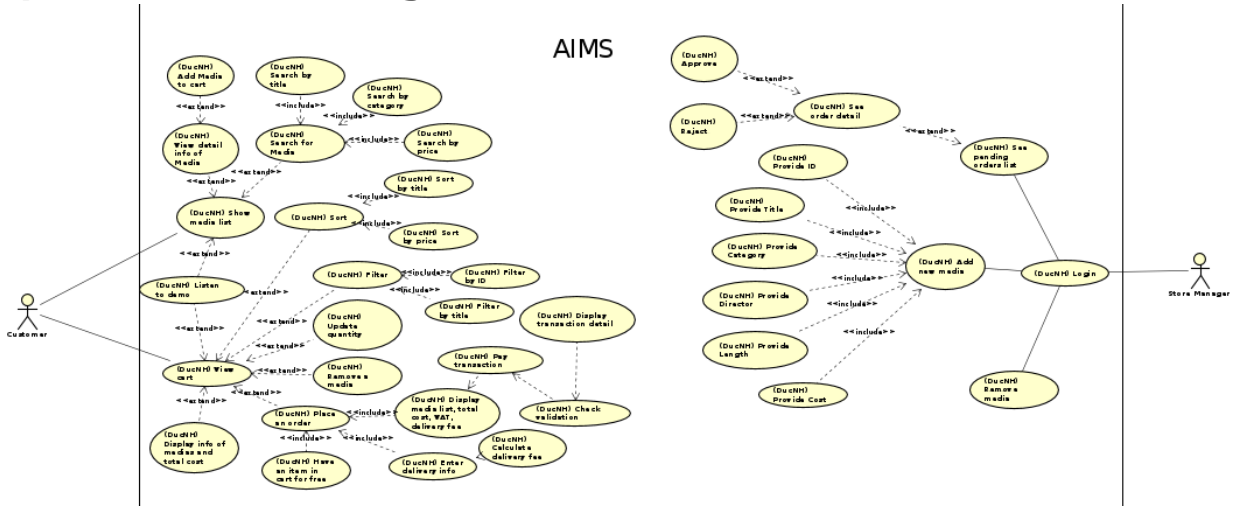
8. Update the Store class to work with Media

Store.java

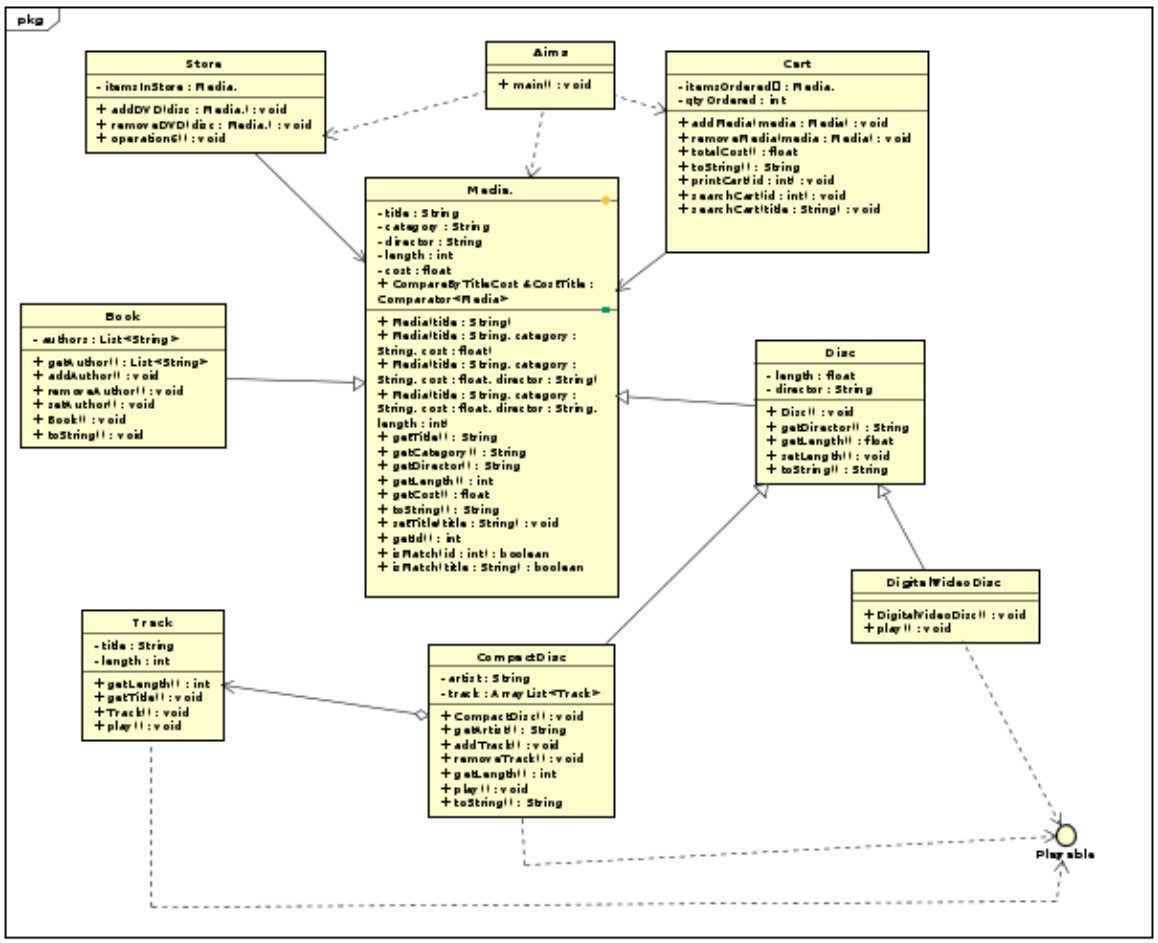
```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.store;
import java.util.ArrayList;

import hust.soict.hedspi.aims.media.*;

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

    public void addMedia(Media media) {
        itemsInStore.add(media);
        System.out.println(media.getTitle()+ " has been added to the store.");
    }

    public void removeMedia(Media media) {
      boolean found = false;
      for (Media item : itemsInStore) {
          if (item.equals(media)) {
            itemsInStore.remove(item);
              System.out.println(media.getTitle() + " has been removed from the store.");
              found = true;
              break;
          }
      }
      if (!found) System.out.println(media.getTitle() + " is not in the store.");
  }
    public ArrayList<Media> getItemsInStore() {return itemsInStore;}
}
```

9. Constructors of whole classes and parent classes

Update Use-Case Diagram



Update Class Diagram



10. Unique item in a list

Override the boolean equals() in Media and Track

Media.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
 // Override equals in Media
public boolean equals(Object obj) {
  if (obj instanceof Media) {
    Media media = (Media) obj;
    if (this.id == media.id) return true;
  }
  return false;
}
```

Track.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
 // Override equals in Track
public boolean equals(Object obj) {
  if (obj instanceof Track) {
    Track track = (Track) obj;
    if (this.titleTrack.equals(track.titleTrack) && this.length == track.length) {
      return true;
    }
  }
  return false;
}
```

11. Polymorphism with toString() method

```
    public static void main(String[] args) {
        ArrayList<Media> mediae = new ArrayList<>();

        CompactDisc cd = new CompactDisc(1,"Soledad", "Ballad", 12.5f);
        DigitalVideoDisc dvd = new DigitalVideoDisc(3,"Final Fantasy X", "Fantasy", 222.22f );
        Book book = new Book(2,"Operating System Concepts", "ICT", 30f);

        // Add some media objects to the list
        mediae.add(cd);
        mediae.add(dvd);
        mediae.add(book);

        for (Media media : mediae) {
            System.out.println(media.toString());
        }
    }
}
```

Problems @ Javadoc Declaration Console

```
rminated> Media [Java Application] C:\Users\Administrator\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_(
- 1. CD - Soledad - Ballad - null - null - 0s - $12.5
- 3. DVD - Final Fantasy X - Fantasy - null - 0s - $222.22
- 2. Book - Operating System Concepts - ICT - [] - $30.0
```

Create an ArrayList of Media, then add some media (CD, DVD or Book) into the list. Iterate through the list and print out the information of the media by using toString() method.

Observe what happens and explain in detail.

When calling this method, toString() will show information of each item depending on its object type. With Polymorphism, toString() method can be used with various object types.

12. Sort media in the cart

MediaComparatorByCostTitle.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;
import java.util.Comparator;

public class MediaComparatorByCostTitle implements Comparator<Media> {
  public int compare(Media media1, Media media2) {
        return Comparator.comparing(Media::getCost)
                .thenComparing(Media::getTitle)
                .compare(media1, media2);
    }
}
```

MediaComparatorByTitleCost.java

```java
/* Student: Nguyen Huu Duc - 20210192
 * Class: 732870
 */
package hust.soict.hedspi.aims.media;
import java.util.Comparator;

public class MediaComparatorByTitleCost implements Comparator<Media> {
  public int compare(Media media1, Media media2) {
        return Comparator.comparing(Media::getTitle)
                .thenComparing(Media::getCost)
                .compare(media1, media2);
    }
}
```

Question

Alternatively, to compare items in the cart, instead of using **Comparator**, we can use the **Comparable** interface and override the **compareTo()** method. You can refer to the Javadocs to see the information of this interface.

Suppose we are taking this **Comparable interface** approach.

- *What class should implement the Comparable interface?*

Media class (abstract class that contains compare objects) should implement.

- *In those classes, how should you implement the compareTo()method be to reflect the ordering that we want?*

We should compare objects' attributes.

For example, we can implement MediaComparatorByTitleCost as follow:

```
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());
}
```

- *Can we have two ordering rules of the item (by title then cost and by cost then title) if we use this Comparable interface approach?*

No, because compareTo() method only returns one int value.

- *Suppose the DVDs has a different ordering rule from the other media types, that is by title, then decreasing length, then cost. How would you modify your code to allow this?*

```
public class DVD extends Media implements Comparable {
        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());
        }
}
```

13. Create a complete console application in the Aims class

```
        Scanner aims = new Scanner(System.in);
        while (true) {
            showMenu();
            int option = aims.nextInt();
            aims.nextLine();
            switch (option) {
                case 1:
                    viewStore(aims);
                    break;
                case 2:
                    updateStore(aims);
                    break;
                case 3:
                    seeCurrentCart(aims);
                    break;
                case 0:
                    System.out.println("Exiting the application...");
                    aims.close();
                    return;
                default:
                    System.out.println("Invalid option. Please try again.");
                    break;
```

Problems @ Javadoc Declaration Console

```
<terminated> Aims [Java Application] C:\Users\Administrator\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86
The Fellowship of the Ring has been added to the store.
The Two Towers has been added to the store.
The Return of the King has been added to the store.
Test has been added to the store.
Test has been added to the cart.
Hi Nguyen Huu Duc (20210192),
AIMS:
--------------------------------
1.View store
2.Update store
3.See current cart
0. Exit
--------------------------------
Please choose a number: 0-1-2-3
```