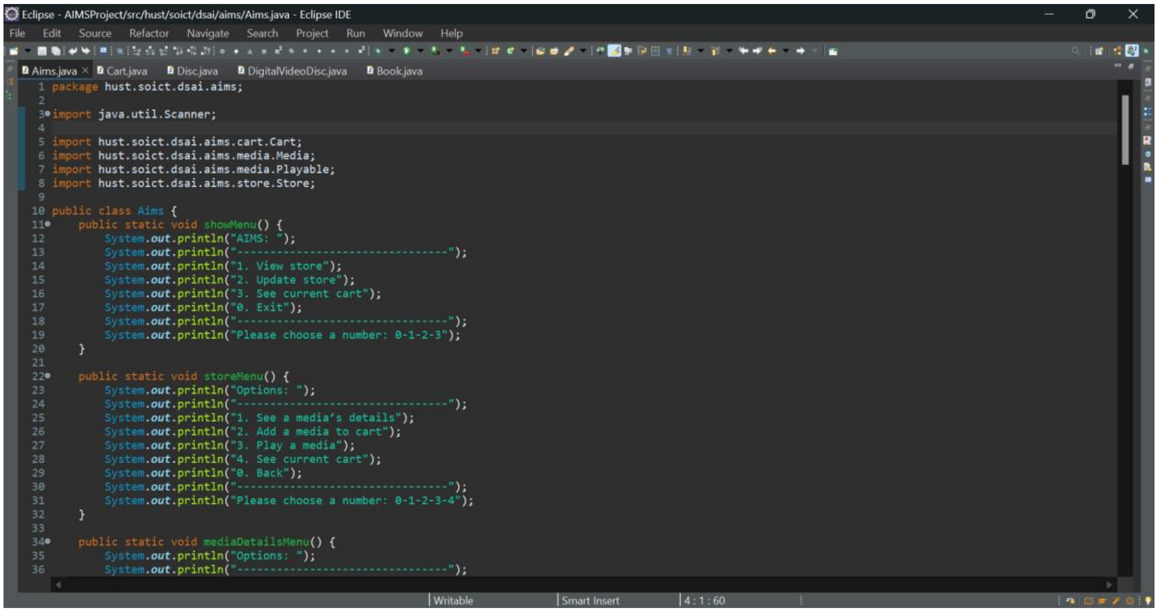
Report Lab04

Name: Nguyễn Đức Mạnh

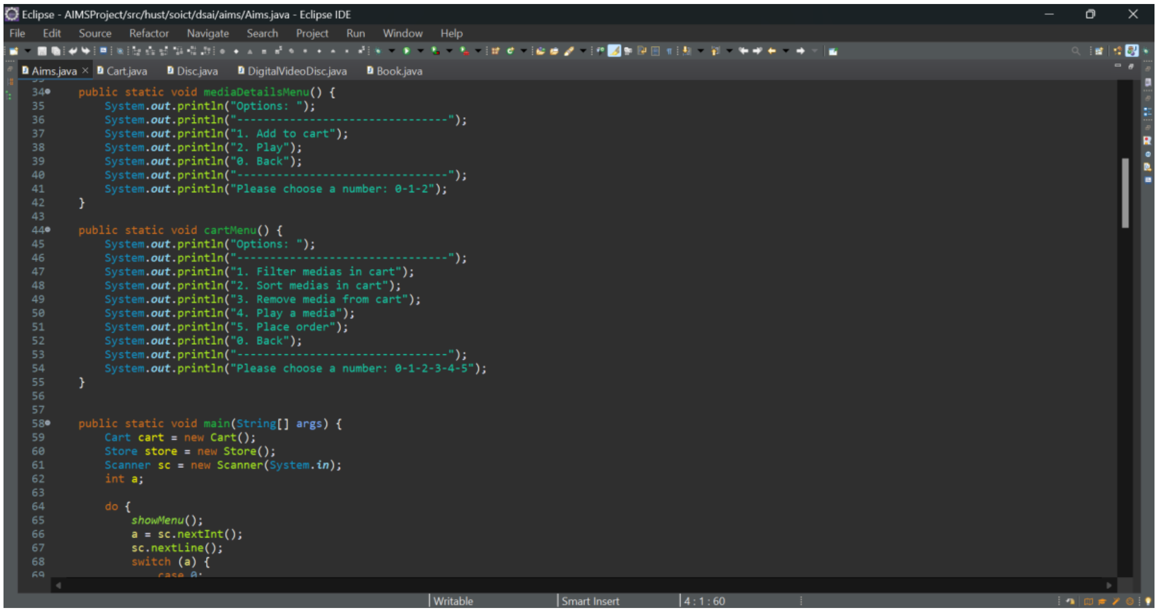
ID: 20235525

1. Source code:

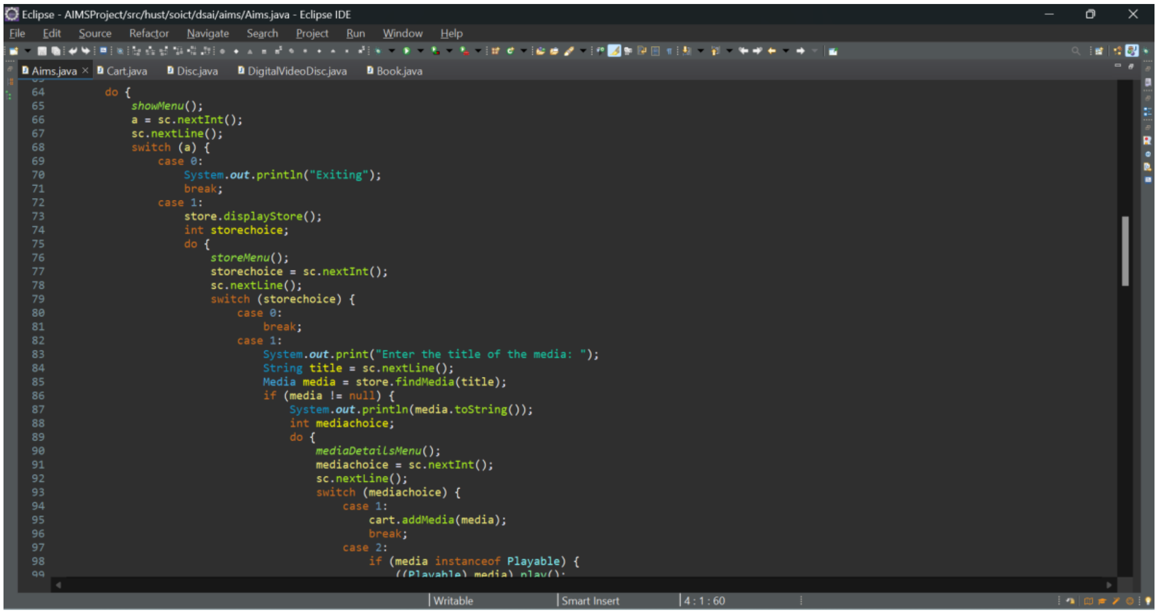
1.1. AIMS class:


Eclipse - AIMSProject/src/hust/soict/dsai/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
Writable
Smart Insert
4 : 1 : 60

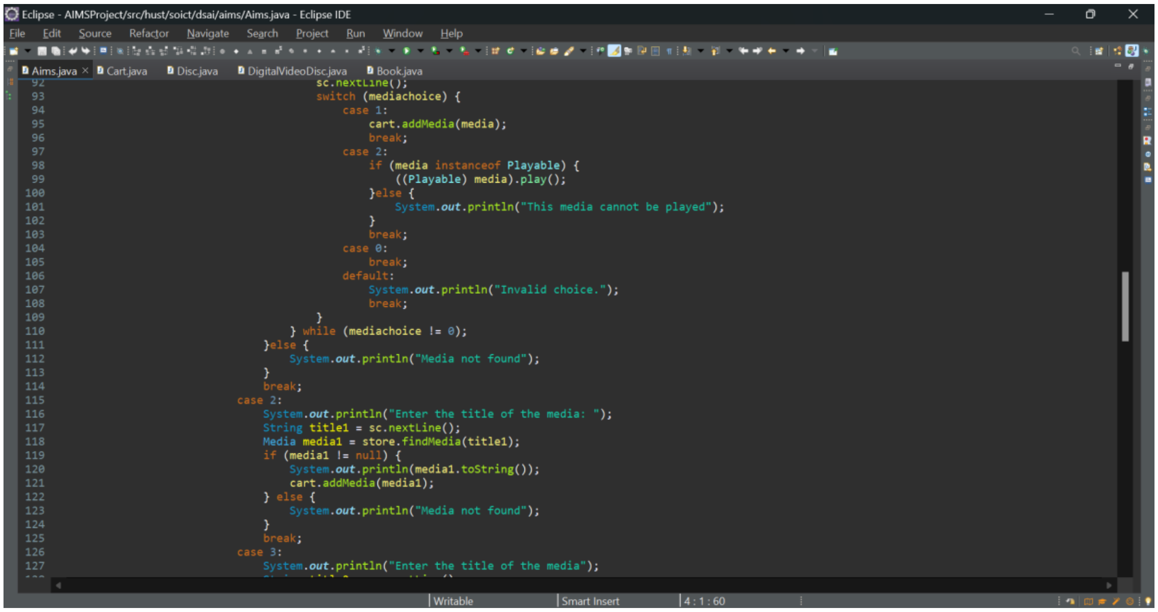
```java
package hust.soict.dsai.aims;

import java.util.Scanner;

import hust.soict.dsai.aims.cart.Cart;
import hust.soict.dsai.aims.media.Media;
import hust.soict.dsai.aims.media.Playable;
import hust.soict.dsai.aims.store.Store;

public class Aims {
    public static void showMenu() {
        System.out.println("AIMS: ");
        System.out.println("--------------------------------");
        System.out.println("1. View store");
        System.out.println("2. Update store");
        System.out.println("3. See current cart");
        System.out.println("0. Exit");
        System.out.println("--------------------------------");
        System.out.println("Please choose a number: 0-1-2-3");
    }

    public static void storeMenu() {
        System.out.println("Options: ");
        System.out.println("--------------------------------");
        System.out.println("1. See a media’s details");
        System.out.println("2. Add a media to cart");
        System.out.println("3. Play a media");
        System.out.println("4. See current cart");
        System.out.println("0. Back");
        System.out.println("--------------------------------");
        System.out.println("Please choose a number: 0-1-2-3-4");
    }

    public static void mediaDetailsMenu() {
        System.out.println("Options: ");
        System.out.println("--------------------------------");
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
Writable
Smart Insert
4 : 1 : 60

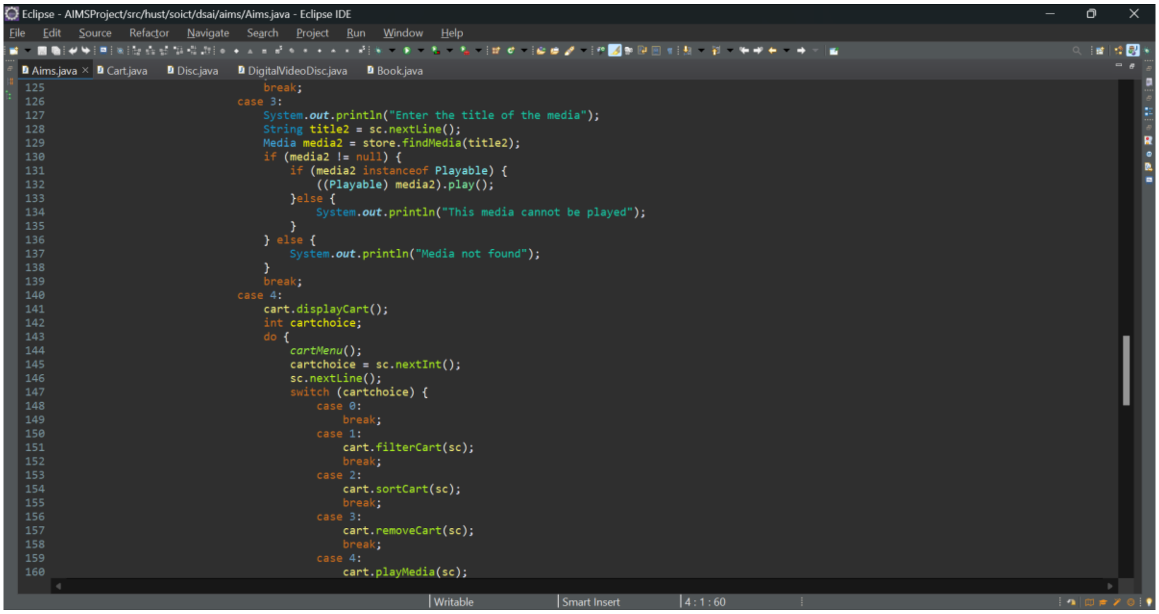
```java
    public static void mediaDetailsMenu() {
        System.out.println("Options: ");
        System.out.println("--------------------------------");
        System.out.println("1. Add to cart");
        System.out.println("2. Play");
        System.out.println("0. Back");
        System.out.println("--------------------------------");
        System.out.println("Please choose a number: 0-1-2");
    }

    public static void cartMenu() {
        System.out.println("Options: ");
        System.out.println("--------------------------------");
        System.out.println("1. Filter medias in cart");
        System.out.println("2. Sort medias in cart");
        System.out.println("3. Remove media from cart");
        System.out.println("4. Play a media");
        System.out.println("5. Place order");
        System.out.println("0. Back");
        System.out.println("--------------------------------");
        System.out.println("Please choose a number: 0-1-2-3-4-5");
    }


    public static void main(String[] args) {
        Cart cart = new Cart();
        Store store = new Store();
        Scanner sc = new Scanner(System.in);
        int a;

        do {
            showMenu();
            a = sc.nextInt();
            sc.nextLine();
            switch (a) {
                case 0:
```



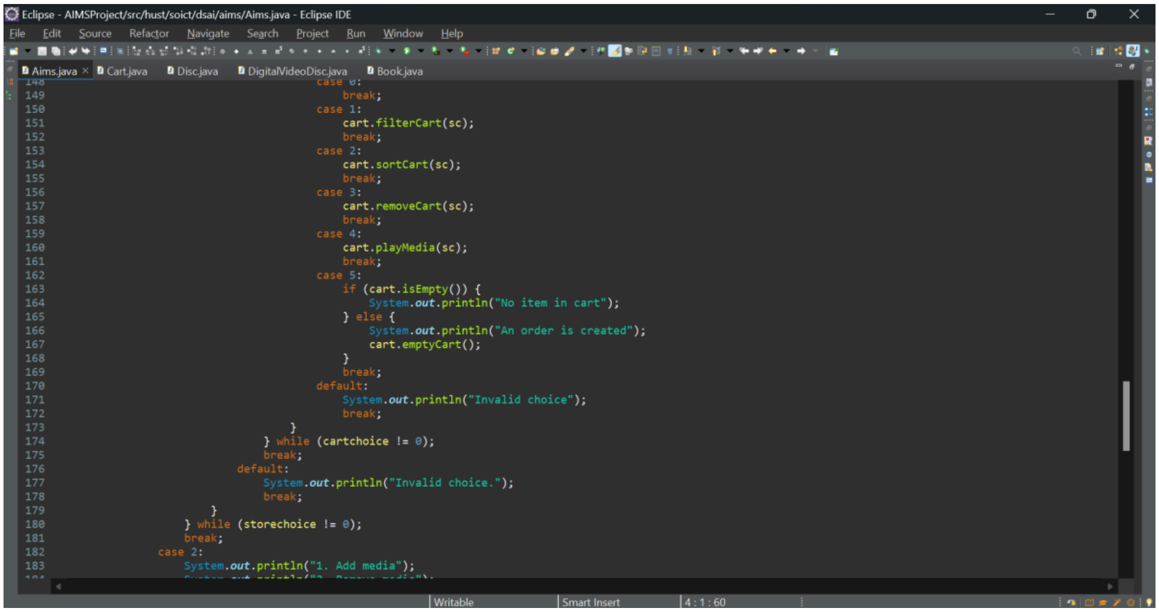
```
        do {
            showMenu();
            a = sc.nextInt();
            sc.nextLine();
            switch (a) {
                case 0:
                    System.out.println("Exiting");
                    break;
                case 1:
                    store.displayStore();
                    int storechoice;
                    do {
                        storeMenu();
                        storechoice = sc.nextInt();
                        sc.nextLine();
                        switch (storechoice) {
                            case 0:
                                break;
                            case 1:
                                System.out.print("Enter the title of the media: ");
                                String title = sc.nextLine();
                                Media media = store.findMedia(title);
                                if (media != null) {
                                    System.out.println(media.toString());
                                    int mediachoice;
                                    do {
                                        mediaDetailsMenu();
                                        mediachoice = sc.nextInt();
                                        sc.nextLine();
                                        switch (mediachoice) {
                                            case 1:
                                                cart.addMedia(media);
                                                break;
                                            case 2:
                                                if (media instanceof Playable) {
                                                    ((Playable) media).play();
```



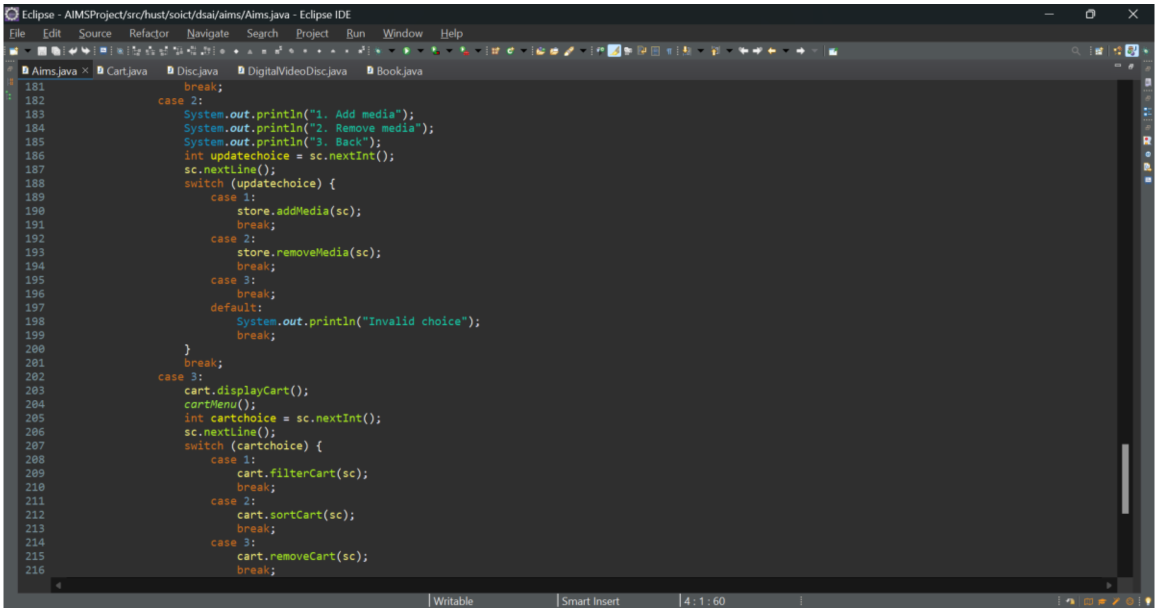
```
                                        sc.nextLine();
                                        switch (mediachoice) {
                                            case 1:
                                                cart.addMedia(media);
                                                break;
                                            case 2:
                                                if (media instanceof Playable) {
                                                    ((Playable) media).play();
                                                }else {
                                                    System.out.println("This media cannot be played");
                                                }
                                                break;
                                            case 0:
                                                break;
                                            default:
                                                System.out.println("Invalid choice.");
                                                break;
                                        }
                                    } while (mediachoice != 0);
                                }else {
                                    System.out.println("Media not found");
                                }
                                break;
                            case 2:
                                System.out.println("Enter the title of the media: ");
                                String title1 = sc.nextLine();
                                Media media1 = store.findMedia(title1);
                                if (media1 != null) {
                                    System.out.println(media1.toString());
                                    cart.addMedia(media1);
                                } else {
                                    System.out.println("Media not found");
                                }
                                break;
                            case 3:
                                System.out.println("Enter the title of the media");
```



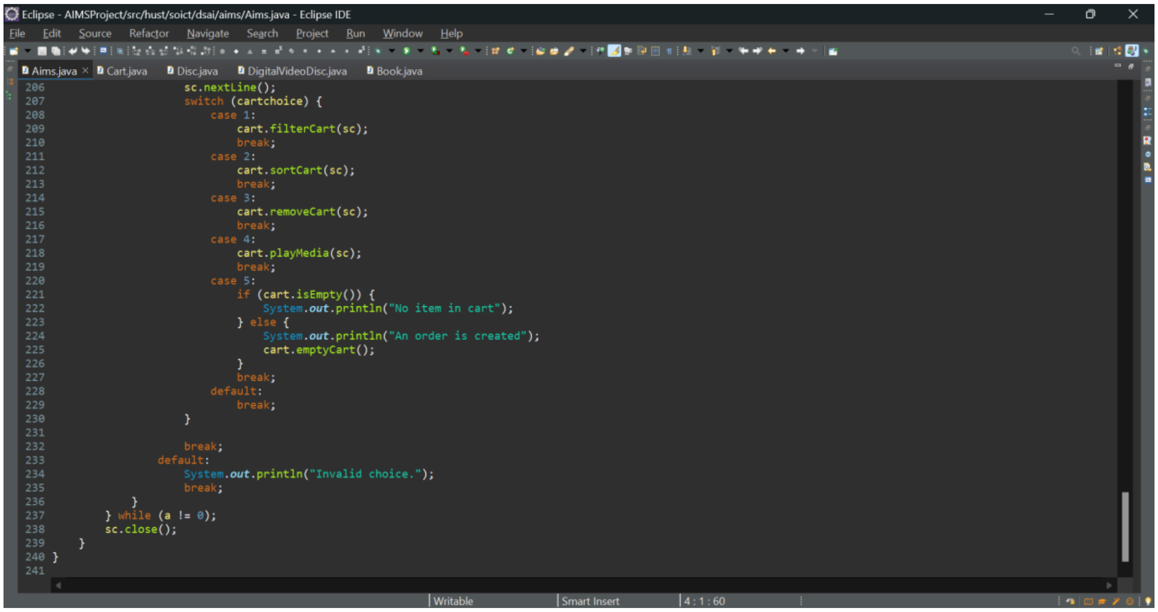
```
                                break;
                            case 3:
                                System.out.println("Enter the title of the media");
                                String title2 = sc.nextLine();
                                Media media2 = store.findMedia(title2);
                                if (media2 != null) {
                                    if (media2 instanceof Playable) {
                                        ((Playable) media2).play();
                                    }else {
                                        System.out.println("This media cannot be played");
                                    }
                                } else {
                                    System.out.println("Media not found");
                                }
                                break;
                            case 4:
                                cart.displayCart();
                                int cartchoice;
                                do {
                                    cartMenu();
                                    cartchoice = sc.nextInt();
                                    sc.nextLine();
                                    switch (cartchoice) {
                                        case 0:
                                            break;
                                        case 1:
                                            cart.filterCart(sc);
                                            break;
                                        case 2:
                                            cart.sortCart(sc);
                                            break;
                                        case 3:
                                            cart.removeCart(sc);
                                            break;
                                        case 4:
                                            cart.playMedia(sc);
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
Writable
Smart Insert
4 : 1 : 60

```
                                        case 0:
                                            break;
                                        case 1:
                                            cart.filterCart(sc);
                                            break;
                                        case 2:
                                            cart.sortCart(sc);
                                            break;
                                        case 3:
                                            cart.removeCart(sc);
                                            break;
                                        case 4:
                                            cart.playMedia(sc);
                                            break;
                                        case 5:
                                            if (cart.isEmpty()) {
                                                System.out.println("No item in cart");
                                            } else {
                                                System.out.println("An order is created");
                                                cart.emptyCart();
                                            }
                                            break;
                                        default:
                                            System.out.println("Invalid choice");
                                            break;
                                    }
                                } while (cartchoice != 0);
                                break;
                            default:
                                System.out.println("Invalid choice.");
                                break;
                        }
                    } while (storechoice != 0);
                    break;
                case 2:
                    System.out.println("1. Add media");
                    System.out.println("2. Remove media");
```

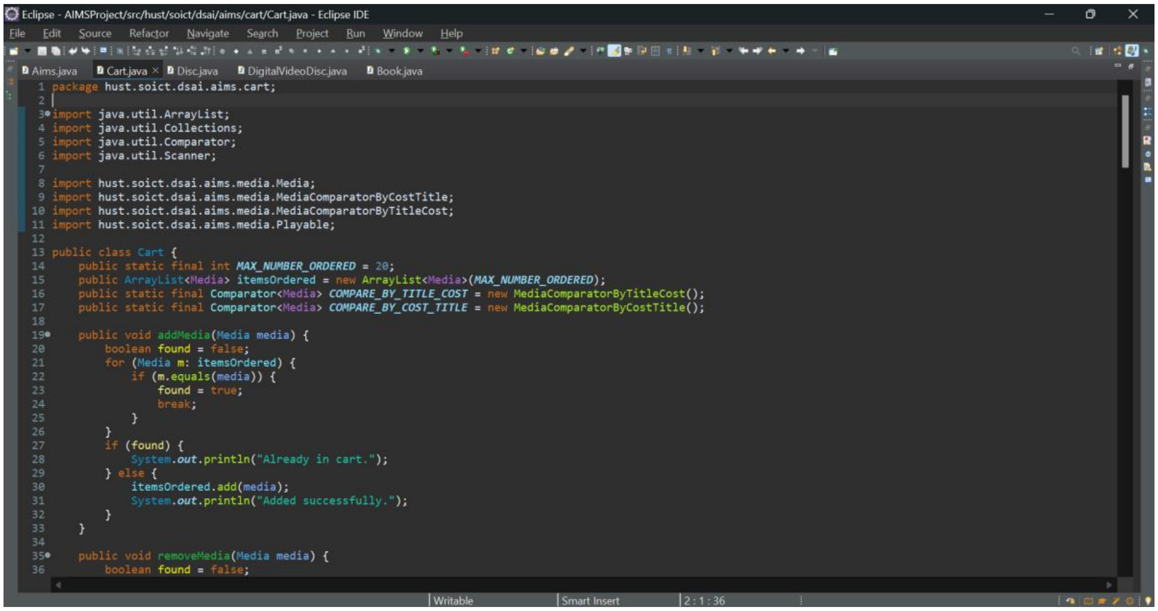

Eclipse - AIMSProject/src/hust/soict/dsai/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
Writable
Smart Insert
4 : 1 : 60

```
                    break;
                case 2:
                    System.out.println("1. Add media");
                    System.out.println("2. Remove media");
                    System.out.println("3. Back");
                    int updatechoice = sc.nextInt();
                    sc.nextLine();
                    switch (updatechoice) {
                        case 1:
                            store.addMedia(sc);
                            break;
                        case 2:
                            store.removeMedia(sc);
                            break;
                        case 3:
                            break;
                        default:
                            System.out.println("Invalid choice");
                            break;
                    }
                    break;
                case 3:
                    cart.displayCart();
                    cartMenu();
                    int cartchoice = sc.nextInt();
                    sc.nextLine();
                    switch (cartchoice) {
                        case 1:
                            cart.filterCart(sc);
                            break;
                        case 2:
                            cart.sortCart(sc);
                            break;
                        case 3:
                            cart.removeCart(sc);
                            break;
```

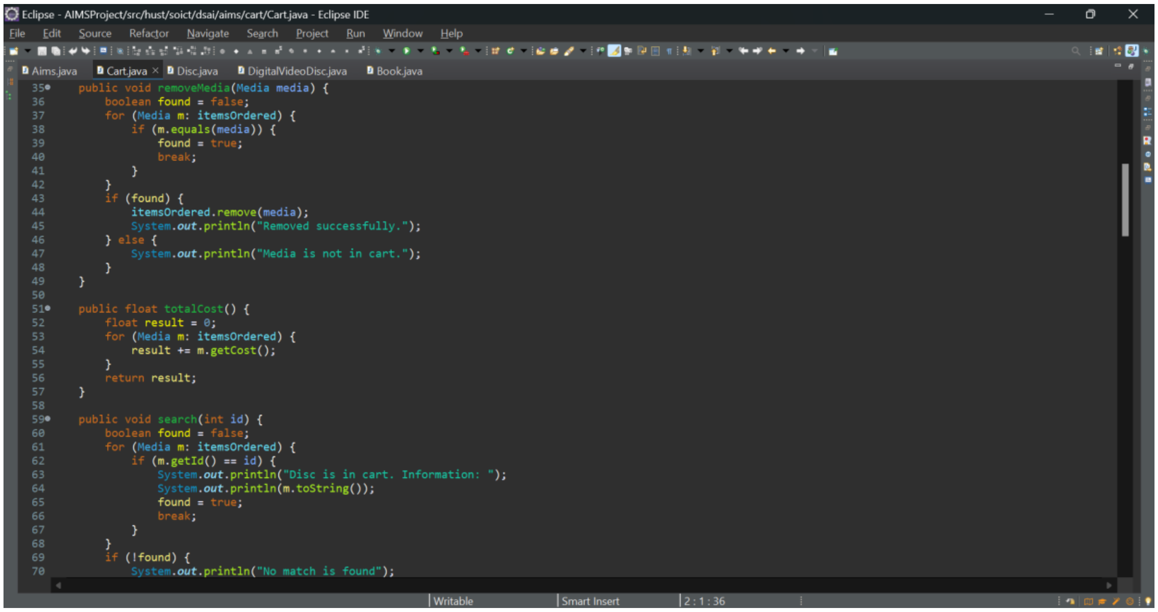

Eclipse - AIMSProject/src/hust/soict/dsai/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
Writable
Smart Insert
4 : 1 : 60

```
                    sc.nextLine();
                    switch (cartchoice) {
                        case 1:
                            cart.filterCart(sc);
                            break;
                        case 2:
                            cart.sortCart(sc);
                            break;
                        case 3:
                            cart.removeCart(sc);
                            break;
                        case 4:
                            cart.playMedia(sc);
                            break;
                        case 5:
                            if (cart.isEmpty()) {
                                System.out.println("No item in cart");
                            } else {
                                System.out.println("An order is created");
                                cart.emptyCart();
                            }
                            break;
                        default:
                            break;
                    }

                    break;
                default:
                    System.out.println("Invalid choice.");
                    break;
            }
        } while (a != 0);
        sc.close();
    }
}

```

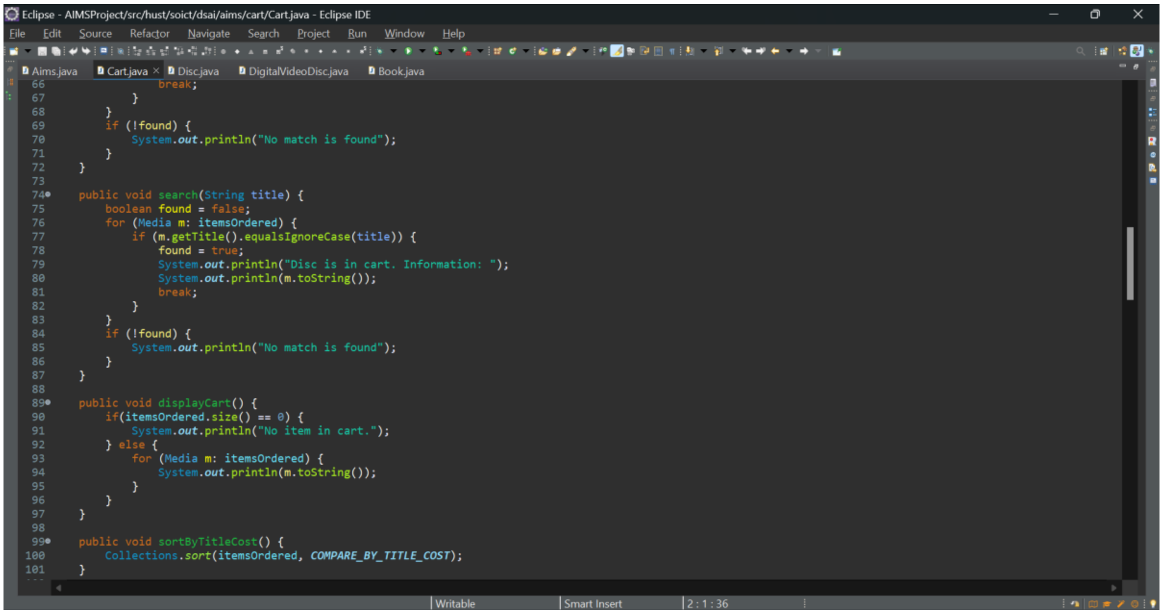
1.2. Cart class:


Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

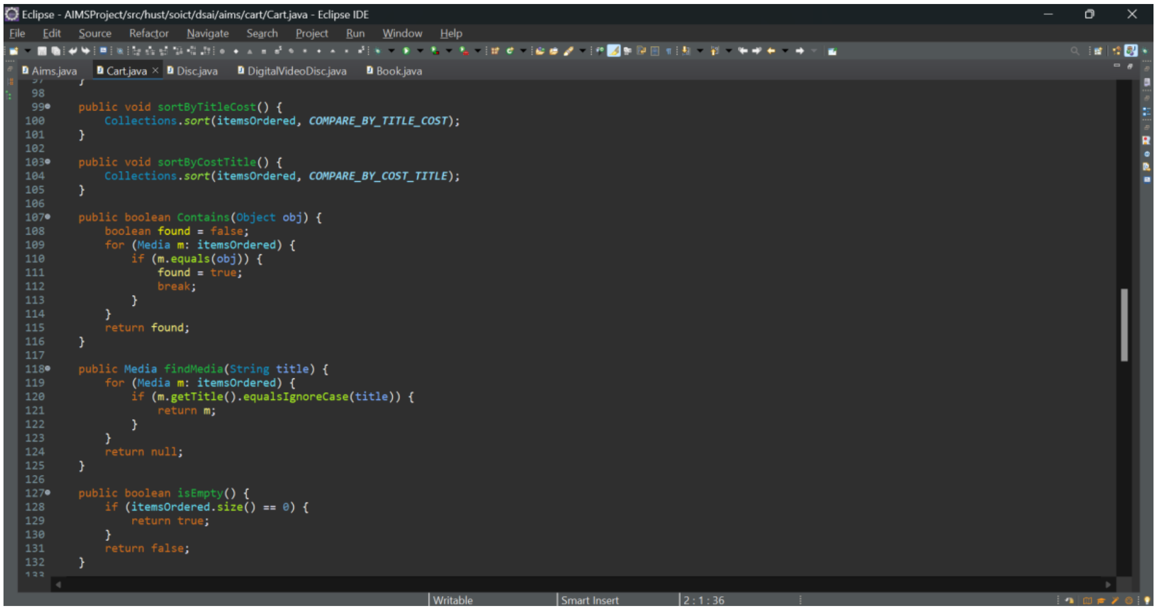
```java
package hust.soict.dsai.aims.cart;

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

import hust.soict.dsai.aims.media.Media;
import hust.soict.dsai.aims.media.MediaComparatorByCostTitle;
import hust.soict.dsai.aims.media.MediaComparatorByTitleCost;
import hust.soict.dsai.aims.media.Playable;

public class Cart {
    public static final int MAX_NUMBER_ORDERED = 20;
    public ArrayList<Media> itemsOrdered = new ArrayList<Media>(MAX_NUMBER_ORDERED);
    public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();
    public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();

    public void addMedia(Media media) {
        boolean found = false;
        for (Media m: itemsOrdered) {
            if (m.equals(media)) {
                found = true;
                break;
            }
        }
        if (found) {
            System.out.println("Already in cart.");
        } else {
            itemsOrdered.add(media);
            System.out.println("Added successfully.");
        }
    }

    public void removeMedia(Media media) {
        boolean found = false;
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

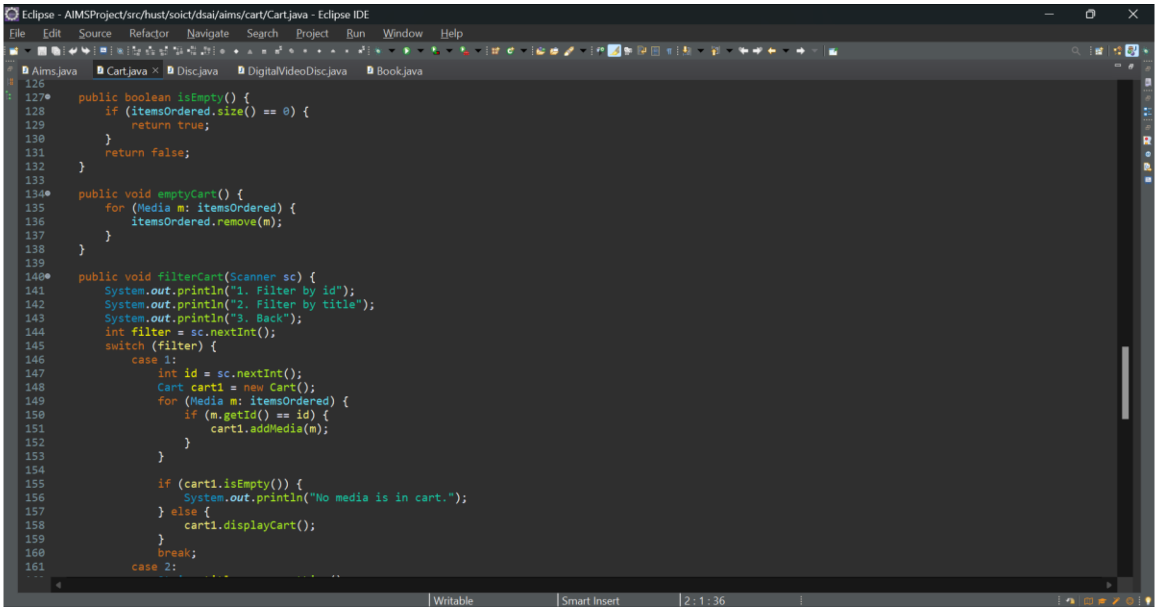
```java
    public void removeMedia(Media media) {
        boolean found = false;
        for (Media m: itemsOrdered) {
            if (m.equals(media)) {
                found = true;
                break;
            }
        }
        if (found) {
            itemsOrdered.remove(media);
            System.out.println("Removed successfully.");
        } else {
            System.out.println("Media is not in cart.");
        }
    }

    public float totalCost() {
        float result = 0;
        for (Media m: itemsOrdered) {
            result += m.getCost();
        }
        return result;
    }

    public void search(int id) {
        boolean found = false;
        for (Media m: itemsOrdered) {
            if (m.getId() == id) {
                System.out.println("Disc is in cart. Information: ");
                System.out.println(m.toString());
                found = true;
                break;
            }
        }
        if (!found) {
            System.out.println("No match is found");
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

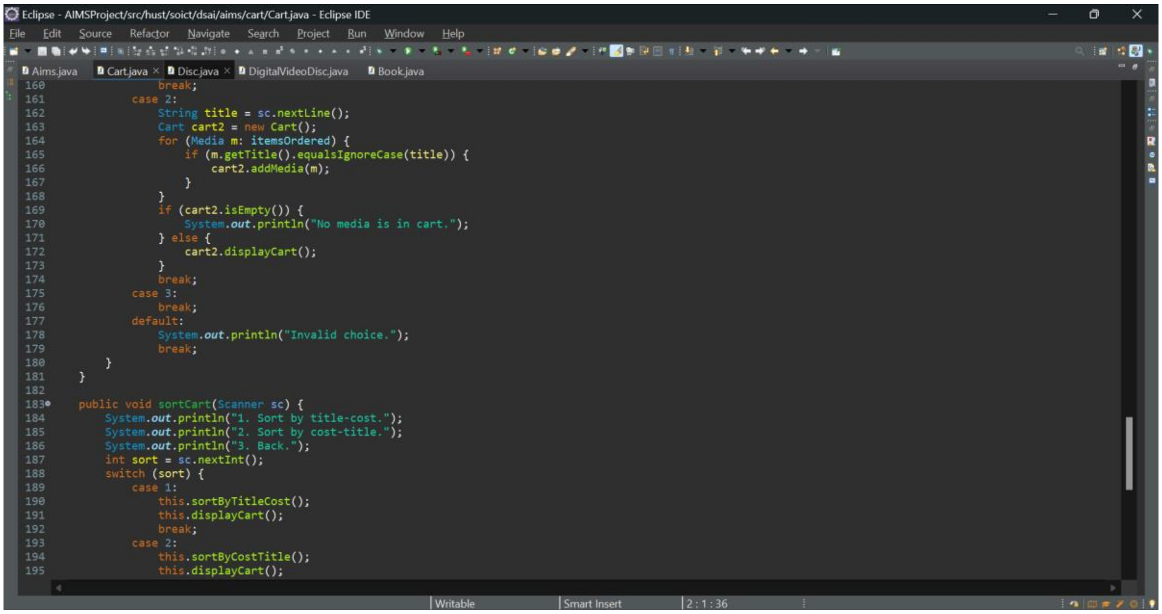
```java
                break;
            }
        }
        if (!found) {
            System.out.println("No match is found");
        }
    }

    public void search(String title) {
        boolean found = false;
        for (Media m: itemsOrdered) {
            if (m.getTitle().equalsIgnoreCase(title)) {
                found = true;
                System.out.println("Disc is in cart. Information: ");
                System.out.println(m.toString());
                break;
            }
        }
        if (!found) {
            System.out.println("No match is found");
        }
    }

    public void displayCart() {
        if(itemsOrdered.size() == 0) {
            System.out.println("No item in cart.");
        } else {
            for (Media m: itemsOrdered) {
                System.out.println(m.toString());
            }
        }
    }

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


Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

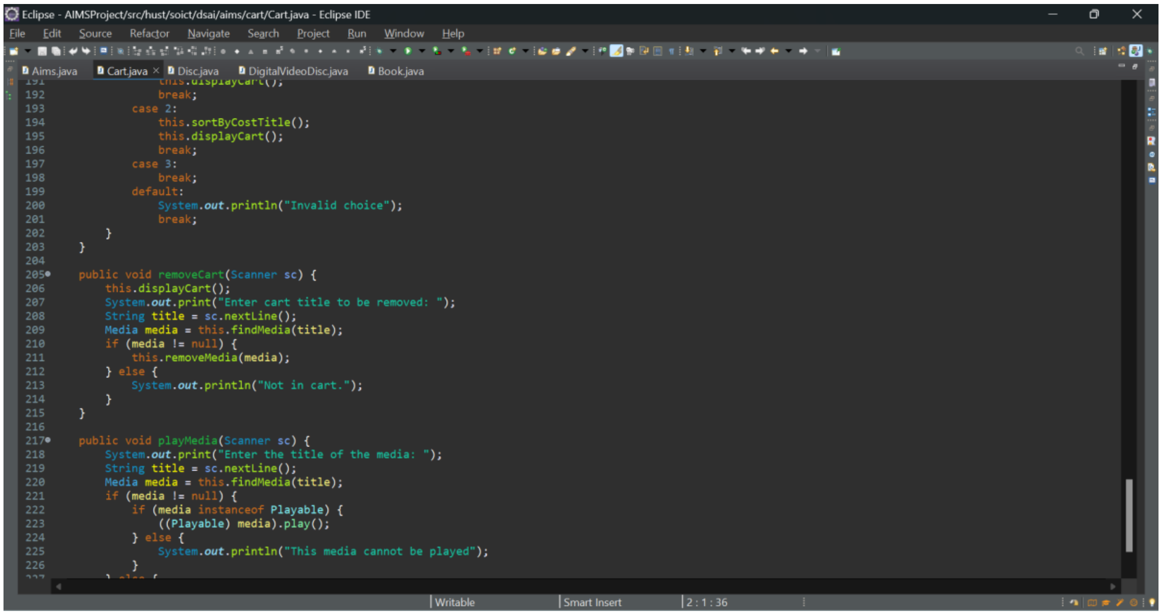
```java

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

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

    public boolean Contains(Object obj) {
        boolean found = false;
        for (Media m: itemsOrdered) {
            if (m.equals(obj)) {
                found = true;
                break;
            }
        }
        return found;
    }

    public Media findMedia(String title) {
        for (Media m: itemsOrdered) {
            if (m.getTitle().equalsIgnoreCase(title)) {
                return m;
            }
        }
        return null;
    }

    public boolean isEmpty() {
        if (itemsOrdered.size() == 0) {
            return true;
        }
        return false;
    }
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

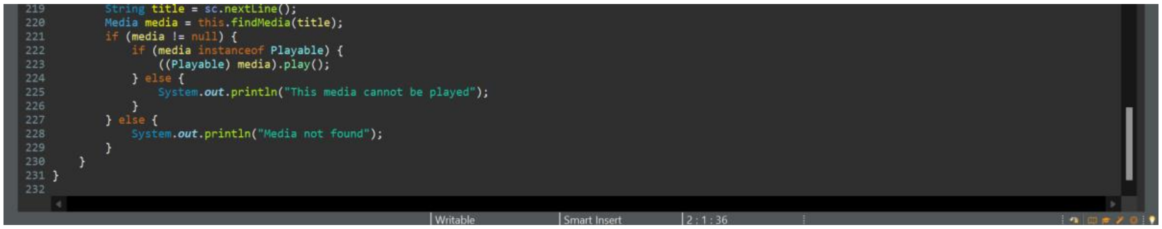
```java

    public boolean isEmpty() {
        if (itemsOrdered.size() == 0) {
            return true;
        }
        return false;
    }

    public void emptyCart() {
        for (Media m: itemsOrdered) {
            itemsOrdered.remove(m);
        }
    }

    public void filterCart(Scanner sc) {
        System.out.println("1. Filter by id");
        System.out.println("2. Filter by title");
        System.out.println("3. Back");
        int filter = sc.nextInt();
        switch (filter) {
            case 1:
                int id = sc.nextInt();
                Cart cart1 = new Cart();
                for (Media m: itemsOrdered) {
                    if (m.getId() == id) {
                        cart1.addMedia(m);
                    }
                }

                if (cart1.isEmpty()) {
                    System.out.println("No media is in cart.");
                } else {
                    cart1.displayCart();
                }
                break;
            case 2:
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

```java
                break;
            case 2:
                String title = sc.nextLine();
                Cart cart2 = new Cart();
                for (Media m: itemsOrdered) {
                    if (m.getTitle().equalsIgnoreCase(title)) {
                        cart2.addMedia(m);
                    }
                }
                if (cart2.isEmpty()) {
                    System.out.println("No media is in cart.");
                } else {
                    cart2.displayCart();
                }
                break;
            case 3:
                break;
            default:
                System.out.println("Invalid choice.");
                break;
        }
    }

    public void sortCart(Scanner sc) {
        System.out.println("1. Sort by title-cost.");
        System.out.println("2. Sort by cost-title.");
        System.out.println("3. Back.");
        int sort = sc.nextInt();
        switch (sort) {
            case 1:
                this.sortByTitleCost();
                this.displayCart();
                break;
            case 2:
                this.sortByCostTitle();
                this.displayCart();
```

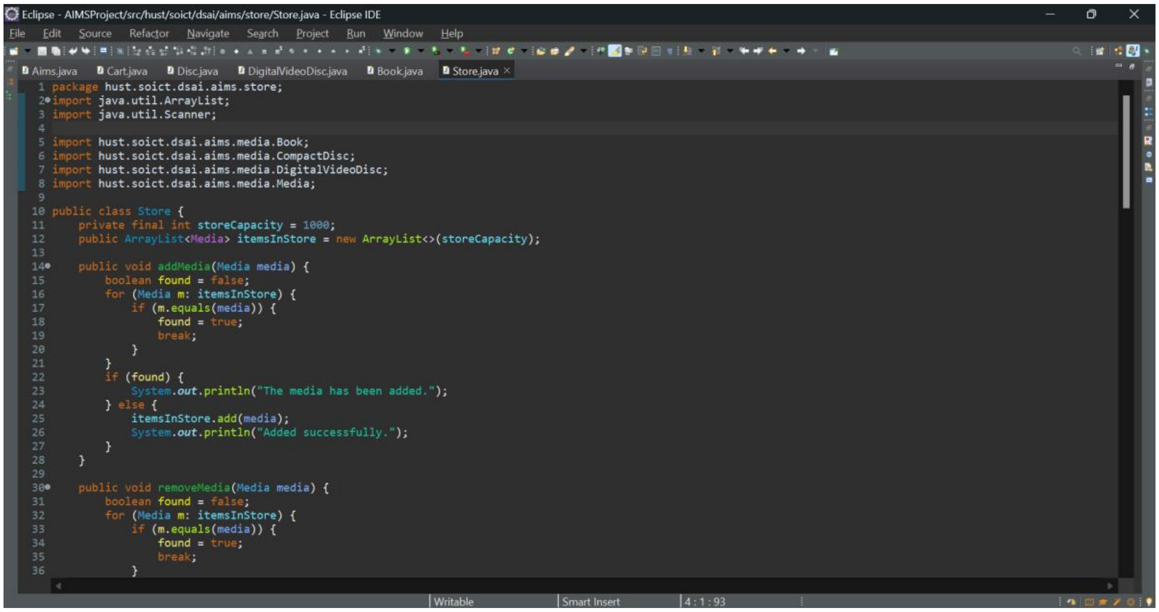

Eclipse - AIMSProject/src/hust/soict/dsai/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Writable
Smart Insert
2 : 1 : 36

```java
                this.displayCart();
                break;
            case 2:
                this.sortByCostTitle();
                this.displayCart();
                break;
            case 3:
                break;
            default:
                System.out.println("Invalid choice");
                break;
        }
    }

    public void removeCart(Scanner sc) {
        this.displayCart();
        System.out.print("Enter cart title to be removed: ");
        String title = sc.nextLine();
        Media media = this.findMedia(title);
        if (media != null) {
            this.removeMedia(media);
        } else {
            System.out.println("Not in cart.");
        }
    }

    public void playMedia(Scanner sc) {
        System.out.print("Enter the title of the media: ");
        String title = sc.nextLine();
        Media media = this.findMedia(title);
        if (media != null) {
            if (media instanceof Playable) {
                ((Playable) media).play();
            } else {
                System.out.println("This media cannot be played");
            }
        } else {
```

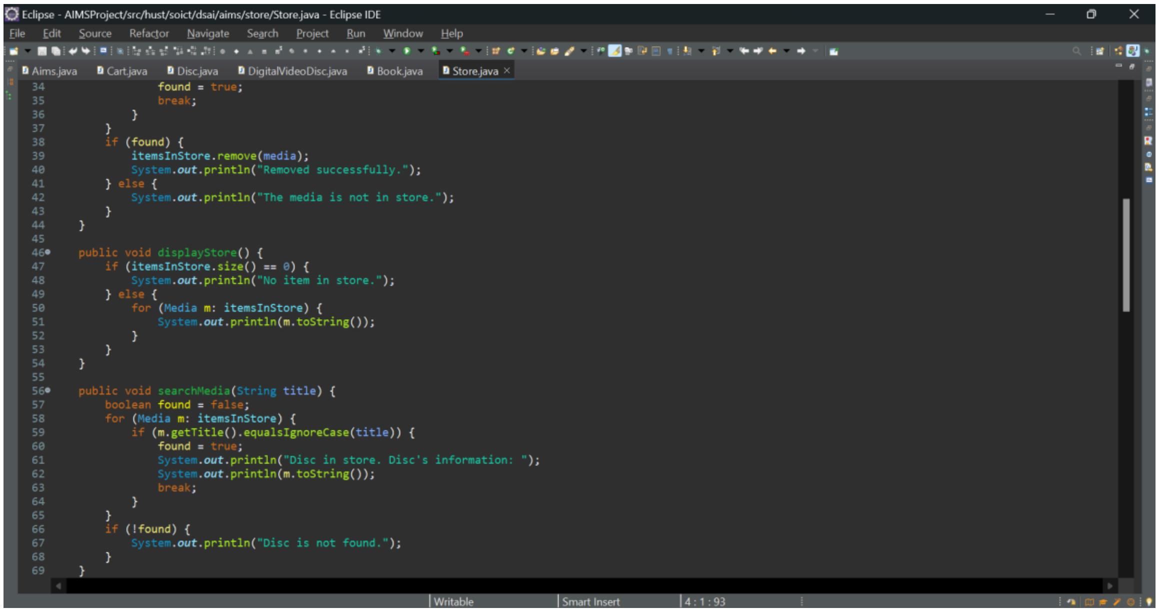


```java
        String title = sc.nextLine();
        Media media = this.findMedia(title);
        if (media != null) {
            if (media instanceof Playable) {
                ((Playable) media).play();
            } else {
                System.out.println("This media cannot be played");
            }
        } else {
            System.out.println("Media not found");
        }
    }
}
```

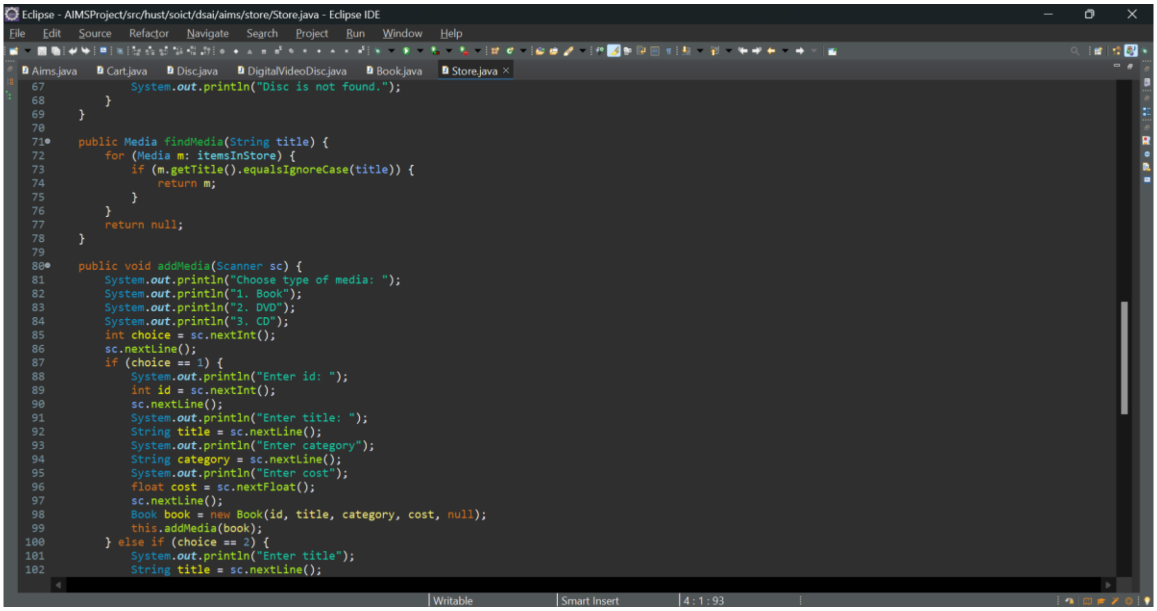
1.3. Store class:


Eclipse - AIMSProject/src/hust/soict/dsai/aims/store/Store.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Writable
Smart Insert
4 : 1 : 93

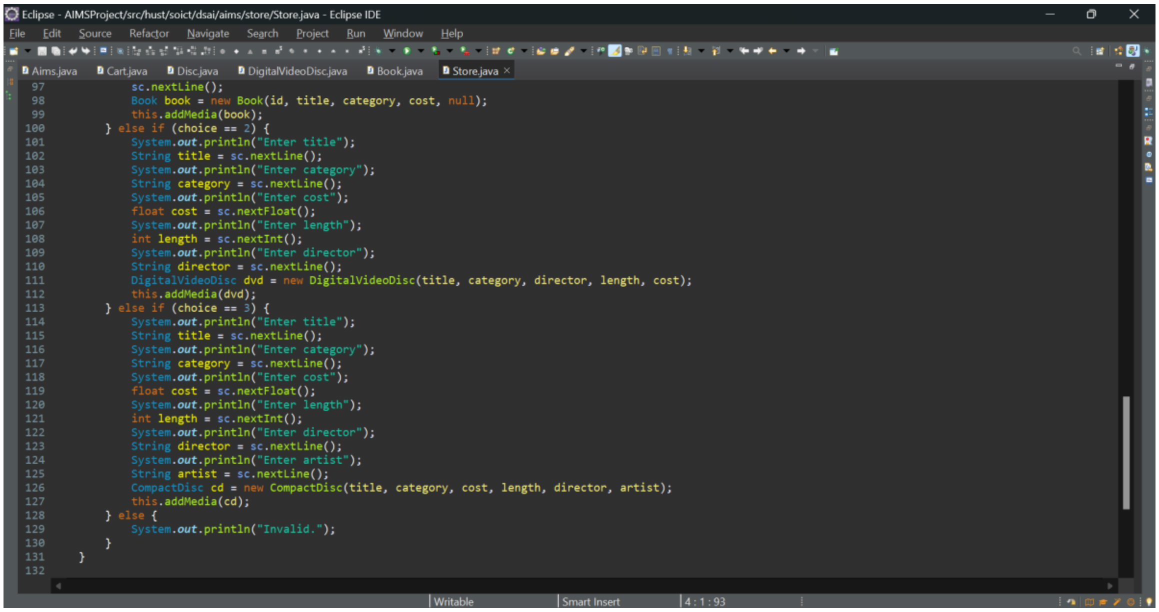
```java
package hust.soict.dsai.aims.store;
import java.util.ArrayList;
import java.util.Scanner;

import hust.soict.dsai.aims.media.Book;
import hust.soict.dsai.aims.media.CompactDisc;
import hust.soict.dsai.aims.media.DigitalVideoDisc;
import hust.soict.dsai.aims.media.Media;

public class Store {
    private final int storeCapacity = 1000;
    public ArrayList<Media> itemsInStore = new ArrayList<>(storeCapacity);

    public void addMedia(Media media) {
        boolean found = false;
        for (Media m: itemsInStore) {
            if (m.equals(media)) {
                found = true;
                break;
            }
        }
        if (found) {
            System.out.println("The media has been added.");
        } else {
            itemsInStore.add(media);
            System.out.println("Added successfully.");
        }
    }

    public void removeMedia(Media media) {
        boolean found = false;
        for (Media m: itemsInStore) {
            if (m.equals(media)) {
                found = true;
                break;
            }
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/store/Store.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Writable
Smart Insert
4 : 1 : 93

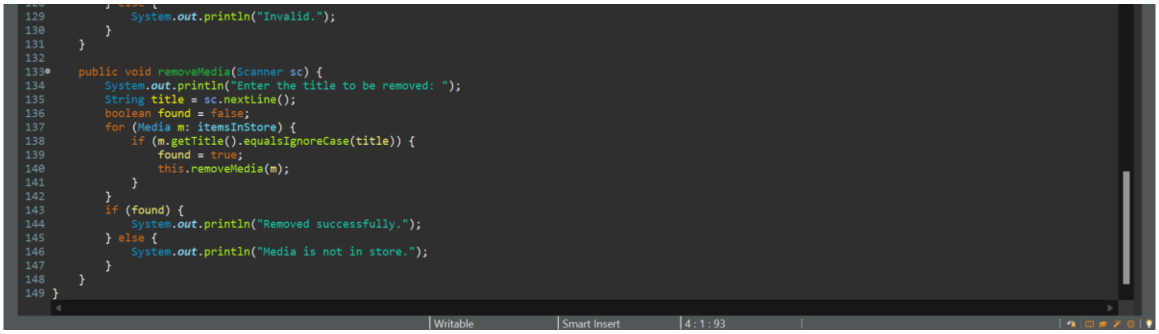
```java
                found = true;
                break;
            }
        }
        if (found) {
            itemsInStore.remove(media);
            System.out.println("Removed successfully.");
        } else {
            System.out.println("The media is not in store.");
        }
    }

    public void displayStore() {
        if (itemsInStore.size() == 0) {
            System.out.println("No item in store.");
        } else {
            for (Media m: itemsInStore) {
                System.out.println(m.toString());
            }
        }
    }

    public void searchMedia(String title) {
        boolean found = false;
        for (Media m: itemsInStore) {
            if (m.getTitle().equalsIgnoreCase(title)) {
                found = true;
                System.out.println("Disc in store. Disc's information: ");
                System.out.println(m.toString());
                break;
            }
        }
        if (!found) {
            System.out.println("Disc is not found.");
        }
    }
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/store/Store.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Writable
Smart Insert
4 : 1 : 93

```java
            System.out.println("Disc is not found.");
        }
    }

    public Media findMedia(String title) {
        for (Media m: itemsInStore) {
            if (m.getTitle().equalsIgnoreCase(title)) {
                return m;
            }
        }
        return null;
    }

    public void addMedia(Scanner sc) {
        System.out.println("Choose type of media: ");
        System.out.println("1. Book");
        System.out.println("2. DVD");
        System.out.println("3. CD");
        int choice = sc.nextInt();
        sc.nextLine();
        if (choice == 1) {
            System.out.println("Enter id: ");
            int id = sc.nextInt();
            sc.nextLine();
            System.out.println("Enter title: ");
            String title = sc.nextLine();
            System.out.println("Enter category");
            String category = sc.nextLine();
            System.out.println("Enter cost");
            float cost = sc.nextFloat();
            sc.nextLine();
            Book book = new Book(id, title, category, cost, null);
            this.addMedia(book);
        } else if (choice == 2) {
            System.out.println("Enter title");
            String title = sc.nextLine();
```

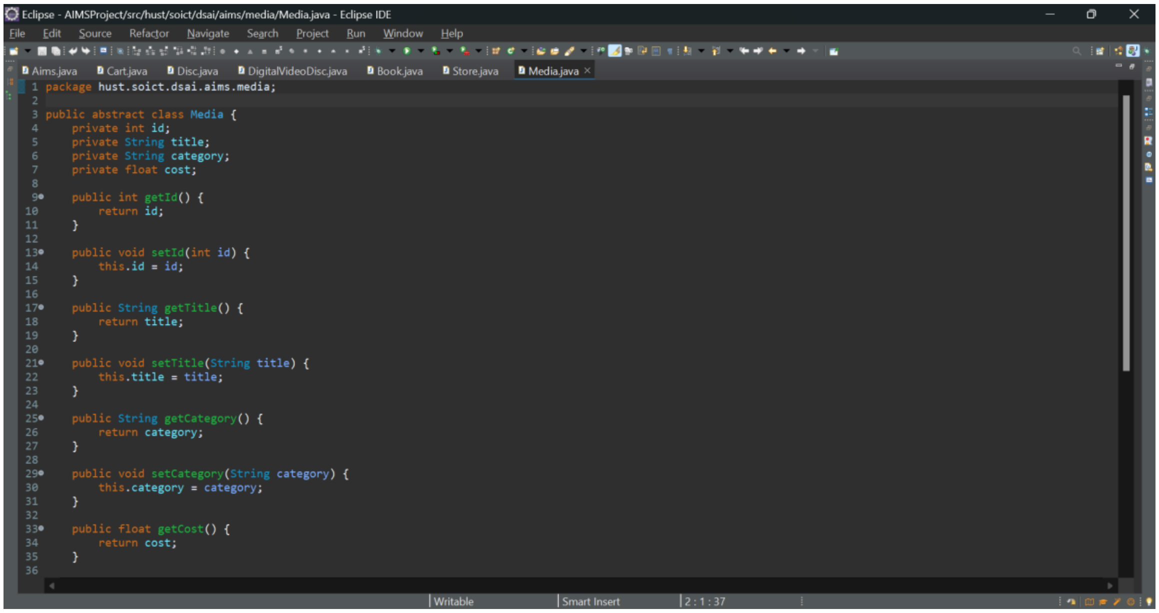

Eclipse - AIMSProject/src/hust/soict/dsai/aims/store/Store.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Writable
Smart Insert
4 : 1 : 93

```java
            sc.nextLine();
            Book book = new Book(id, title, category, cost, null);
            this.addMedia(book);
        } else if (choice == 2) {
            System.out.println("Enter title");
            String title = sc.nextLine();
            System.out.println("Enter category");
            String category = sc.nextLine();
            System.out.println("Enter cost");
            float cost = sc.nextFloat();
            System.out.println("Enter length");
            int length = sc.nextInt();
            System.out.println("Enter director");
            String director = sc.nextLine();
            DigitalVideoDisc dvd = new DigitalVideoDisc(title, category, director, length, cost);
            this.addMedia(dvd);
        } else if (choice == 3) {
            System.out.println("Enter title");
            String title = sc.nextLine();
            System.out.println("Enter category");
            String category = sc.nextLine();
            System.out.println("Enter cost");
            float cost = sc.nextFloat();
            System.out.println("Enter length");
            int length = sc.nextInt();
            System.out.println("Enter director");
            String director = sc.nextLine();
            System.out.println("Enter artist");
            String artist = sc.nextLine();
            CompactDisc cd = new CompactDisc(title, category, cost, length, director, artist);
            this.addMedia(cd);
        } else {
            System.out.println("Invalid.");
        }
    }

```

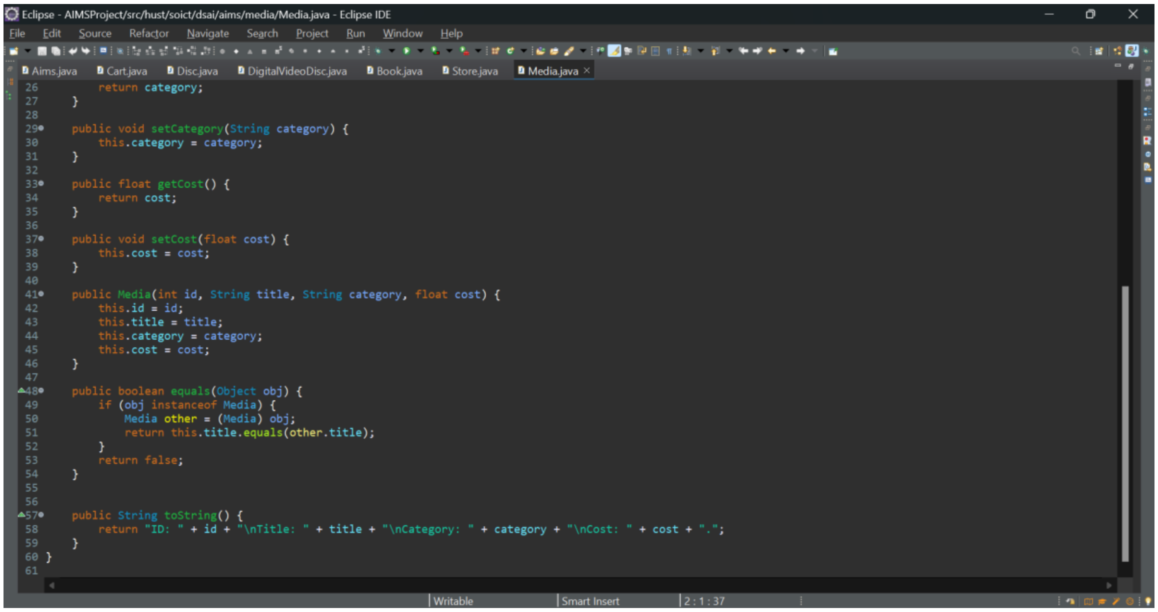


```java
            System.out.println("Invalid.");
        }
    }

    public void removeMedia(Scanner sc) {
        System.out.println("Enter the title to be removed: ");
        String title = sc.nextLine();
        boolean found = false;
        for (Media m: itemsInStore) {
            if (m.getTitle().equalsIgnoreCase(title)) {
                found = true;
                this.removeMedia(m);
            }
        }
        if (found) {
            System.out.println("Removed successfully.");
        } else {
            System.out.println("Media is not in store.");
        }
    }
}
```

1.4. Media class:

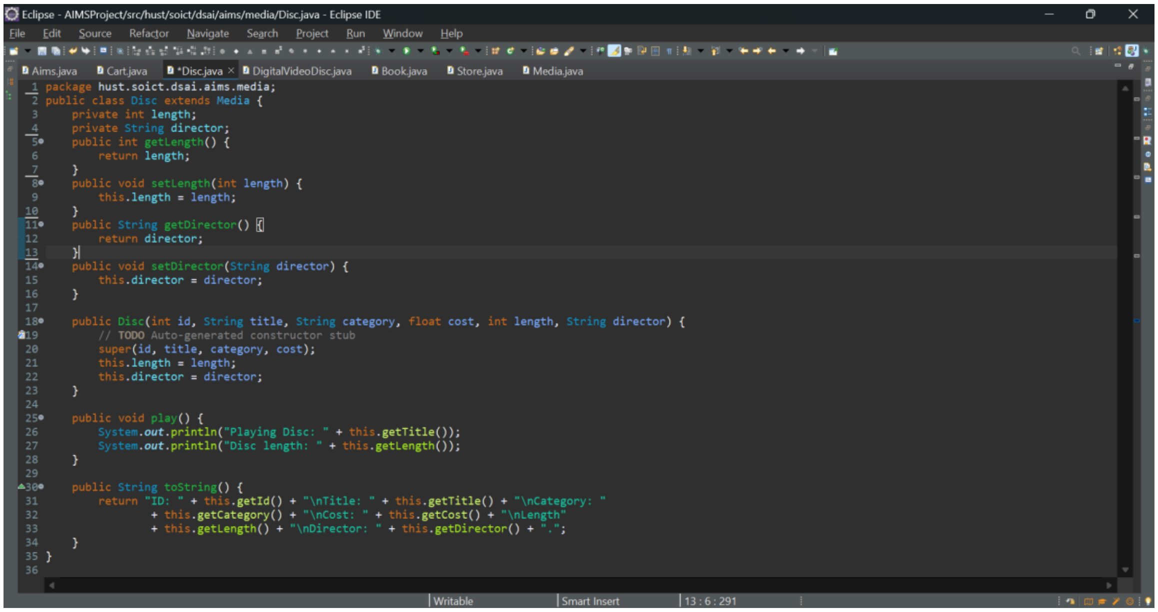

Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/Media.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Media.java
Writable
Smart Insert
2 : 1 : 37

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

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

```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/Media.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Media.java
Writable
Smart Insert
2 : 1 : 37

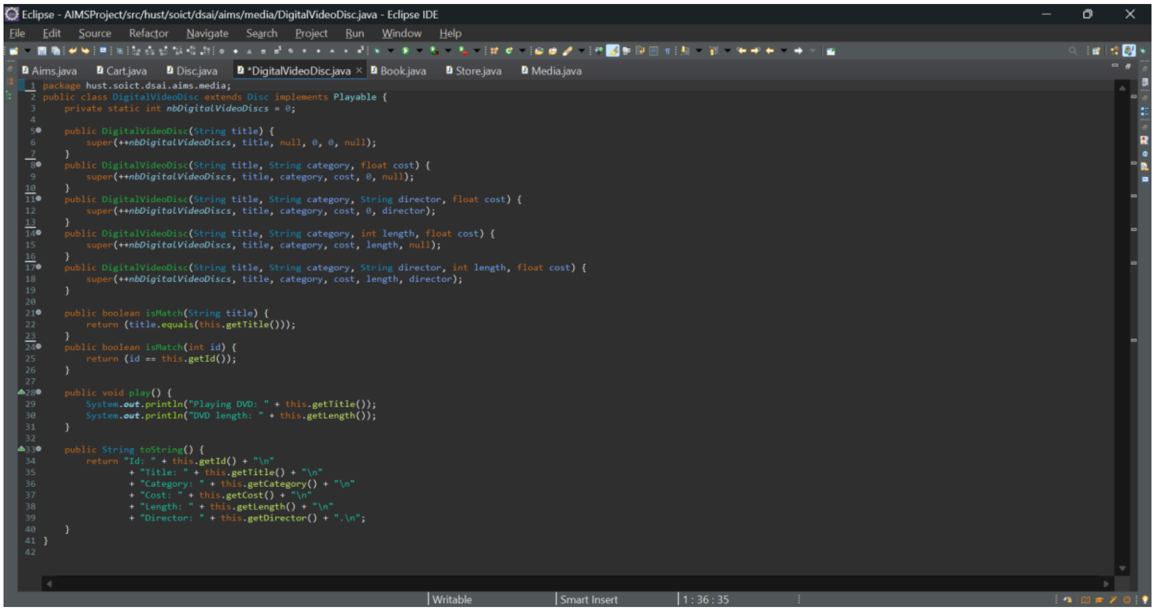
```java
        return category;
    }

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

    public float getCost() {
        return cost;
    }

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

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

    public boolean equals(Object obj) {
        if (obj instanceof Media) {
            Media other = (Media) obj;
            return this.title.equals(other.title);
        }
        return false;
    }


    public String toString() {
        return "ID: " + id + "\nTitle: " + title + "\nCategory: " + category + "\nCost: " + cost + ".";
    }
}
```

1.5. Disc class:


Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/Disc.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java
Cart.java
*Disc.java
DigitalVideoDisc.java
Book.java
Store.java
Media.java
Writable
Smart Insert
13 : 6 : 291

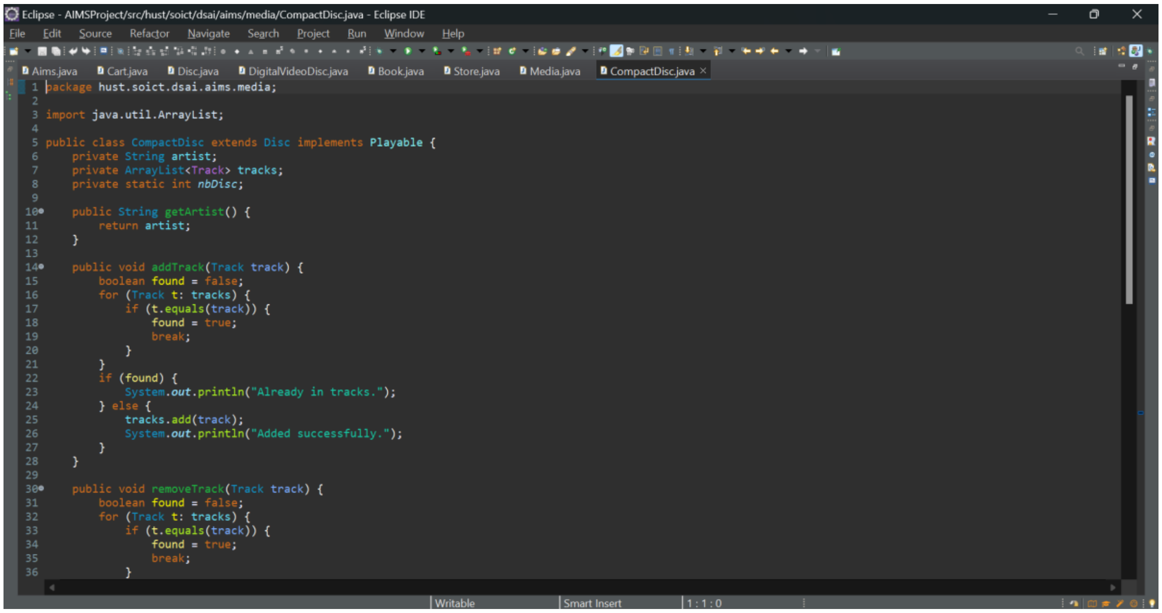
```java
package hust.soict.dsai.aims.media;
public class Disc extends Media {
    private int length;
    private String director;
    public int getLength() {
        return length;
    }
    public void setLength(int length) {
        this.length = length;
    }
    public String getDirector() {
        return director;
    }
    public void setDirector(String director) {
        this.director = director;
    }

    public Disc(int id, String title, String category, float cost, int length, String director) {
        // TODO Auto-generated constructor stub
        super(id, title, category, cost);
        this.length = length;
        this.director = director;
    }

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

    public String toString() {
        return "ID: " + this.getId() + "\nTitle: " + this.getTitle() + "\nCategory: "
                + this.getCategory() + "\nCost: " + this.getCost() + "\nLength"
                + this.getLength() + "\nDirector: " + this.getDirector() + ".";
    }
}
```

1.6. DigitalVideoDisc class:

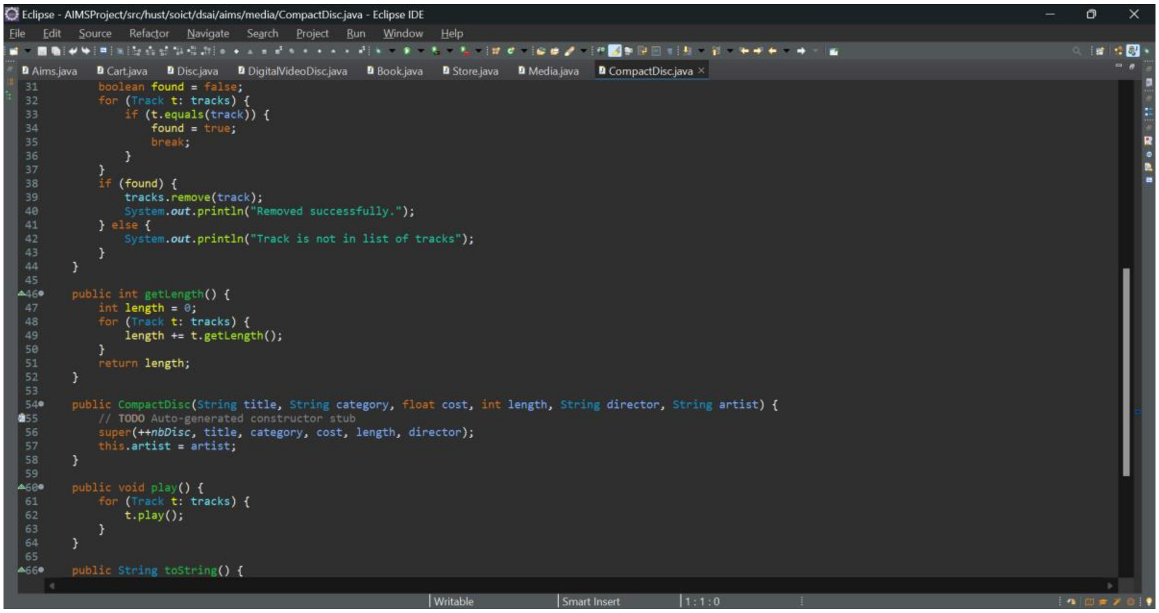

Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/DigitalVideoDisc.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java *DigitalVideoDisc.java Book.java Store.java Media.java
Writable Smart Insert 1 : 36 : 35

```java
package hust.soict.dsai.aims.media;
public class DigitalVideoDisc extends Disc implements Playable {
    private static int nbDigitalVideoDiscs = 0;

    public DigitalVideoDisc(String title) {
        super(++nbDigitalVideoDiscs, title, null, 0, 0, null);
    }
    public DigitalVideoDisc(String title, String category, float cost) {
        super(++nbDigitalVideoDiscs, title, category, cost, 0, null);
    }
    public DigitalVideoDisc(String title, String category, String director, float cost) {
        super(++nbDigitalVideoDiscs, title, category, cost, 0, director);
    }
    public DigitalVideoDisc(String title, String category, int length, float cost) {
        super(++nbDigitalVideoDiscs, title, category, cost, length, null);
    }
    public DigitalVideoDisc(String title, String category, String director, int length, float cost) {
        super(++nbDigitalVideoDiscs, title, category, cost, length, director);
    }

    public boolean isMatch(String title) {
        return (title.equals(this.getTitle()));
    }
    public boolean isMatch(int id) {
        return (id == this.getId());
    }

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

    public String toString() {
        return "Id: " + this.getId() + "\n"
                + "Title: " + this.getTitle() + "\n"
                + "Category: " + this.getCategory() + "\n"
                + "Cost: " + this.getCost() + "\n"
                + "Length: " + this.getLength() + "\n"
                + "Director: " + this.getDirector() + ".\n";
    }
}
```

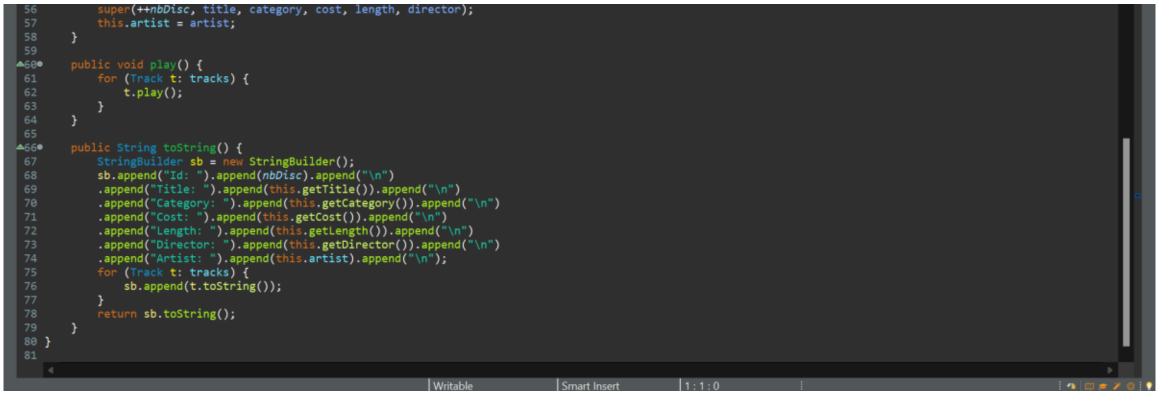
1.7. CompactDisc class:


Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/CompactDisc.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java Store.java Media.java CompactDisc.java
Writable Smart Insert 1 : 1 : 0

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

import java.util.ArrayList;

public class CompactDisc extends Disc implements Playable {
    private String artist;
    private ArrayList<Track> tracks;
    private static int nbDisc;

    public String getArtist() {
        return artist;
    }

    public void addTrack(Track track) {
        boolean found = false;
        for (Track t: tracks) {
            if (t.equals(track)) {
                found = true;
                break;
            }
        }
        if (found) {
            System.out.println("Already in tracks.");
        } else {
            tracks.add(track);
            System.out.println("Added successfully.");
        }
    }

    public void removeTrack(Track track) {
        boolean found = false;
        for (Track t: tracks) {
            if (t.equals(track)) {
                found = true;
                break;
            }
```

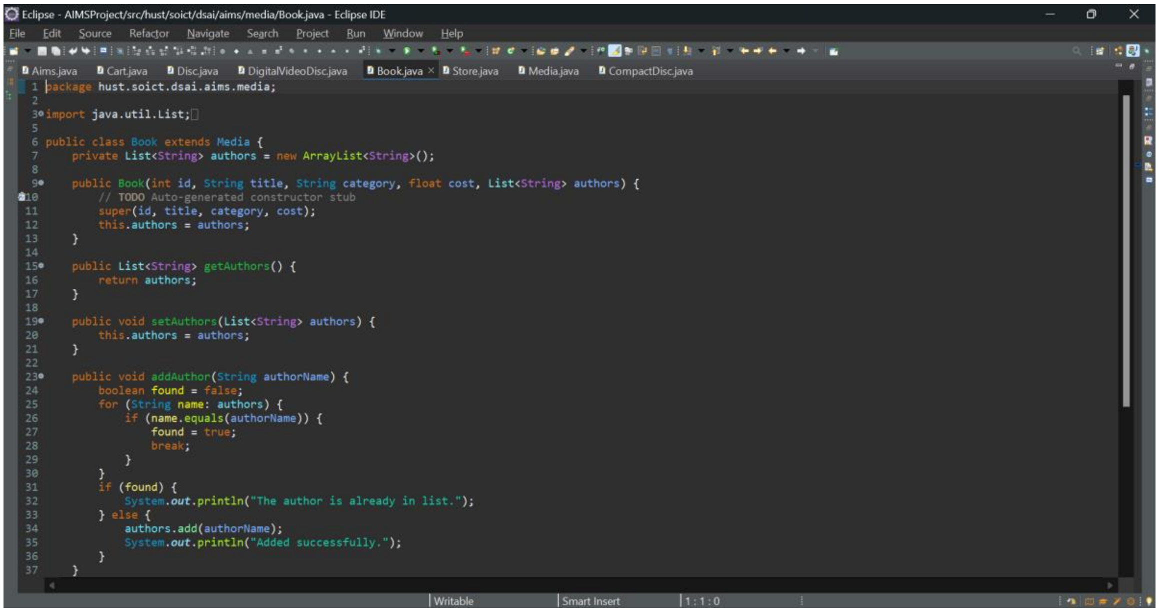

Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/CompactDisc.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java Store.java Media.java CompactDisc.java
Writable Smart Insert 1 : 1 : 0

```java
        boolean found = false;
        for (Track t: tracks) {
            if (t.equals(track)) {
                found = true;
                break;
            }
        }
        if (found) {
            tracks.remove(track);
            System.out.println("Removed successfully.");
        } else {
            System.out.println("Track is not in list of tracks");
        }
    }

    public int getLength() {
        int length = 0;
        for (Track t: tracks) {
            length += t.getLength();
        }
        return length;
    }

    public CompactDisc(String title, String category, float cost, int length, String director, String artist) {
        // TODO Auto-generated constructor stub
        super(++nbDisc, title, category, cost, length, director);
        this.artist = artist;
    }

    public void play() {
        for (Track t: tracks) {
            t.play();
        }
    }

    public String toString() {
```

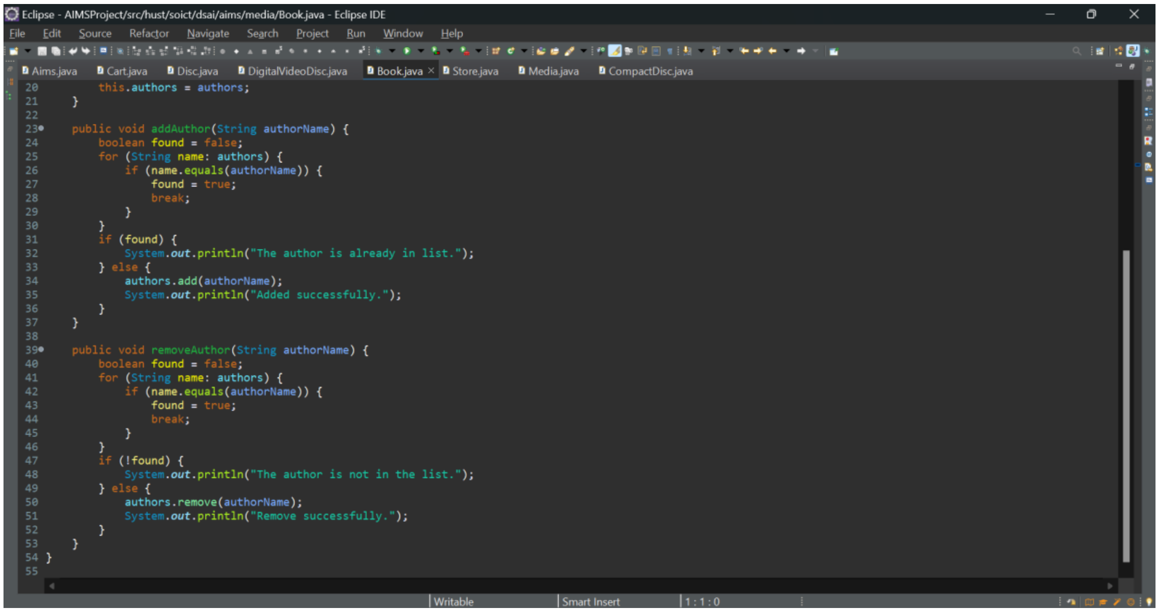


```java
        super(++nbDisc, title, category, cost, length, director);
        this.artist = artist;
    }

    public void play() {
        for (Track t: tracks) {
            t.play();
        }
    }

    public String toString() {
        StringBuilder sb = new StringBuilder();
        sb.append("Id: ").append(nbDisc).append("\n")
        .append("Title: ").append(this.getTitle()).append("\n")
        .append("Category: ").append(this.getCategory()).append("\n")
        .append("Cost: ").append(this.getCost()).append("\n")
        .append("Length: ").append(this.getLength()).append("\n")
        .append("Director: ").append(this.getDirector()).append("\n")
        .append("Artist: ").append(this.artist).append("\n");
        for (Track t: tracks) {
            sb.append(t.toString());
        }
        return sb.toString();
    }
}
```

1.8. Book class:

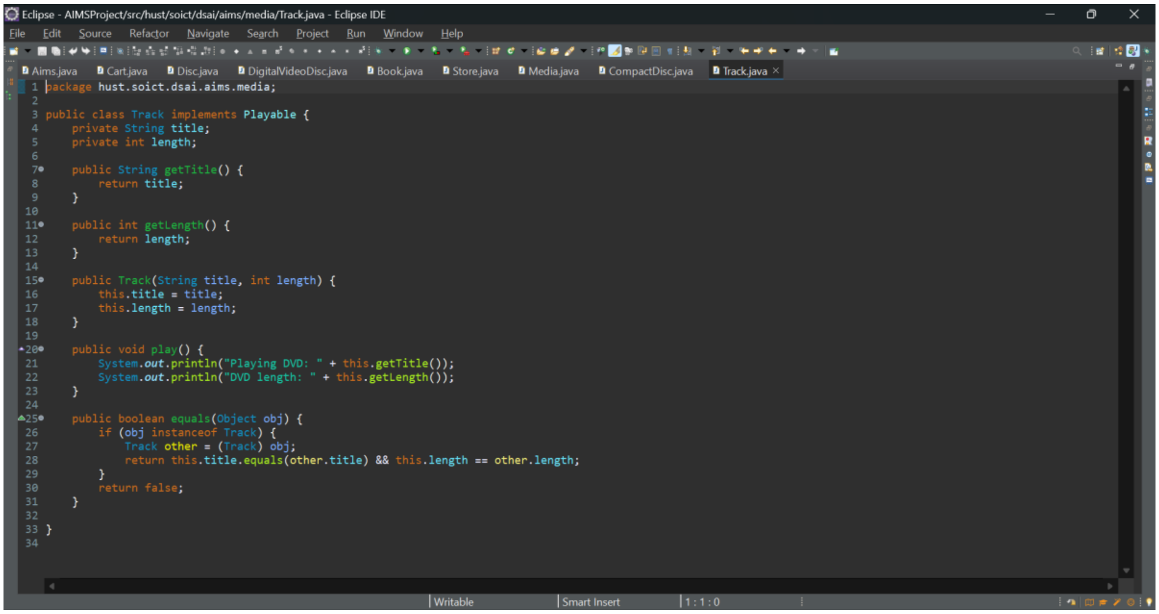

Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/Book.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java Store.java Media.java CompactDisc.java
Writable Smart Insert 1 : 1 : 0

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

import java.util.List;

public class Book extends Media {
    private List<String> authors = new ArrayList<String>();

    public Book(int id, String title, String category, float cost, List<String> authors) {
        // TODO Auto-generated constructor stub
        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) {
        boolean found = false;
        for (String name: authors) {
            if (name.equals(authorName)) {
                found = true;
                break;
            }
        }
        if (found) {
            System.out.println("The author is already in list.");
        } else {
            authors.add(authorName);
            System.out.println("Added successfully.");
        }
    }
```


Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/Book.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java Store.java Media.java CompactDisc.java
Writable Smart Insert 1 : 1 : 0

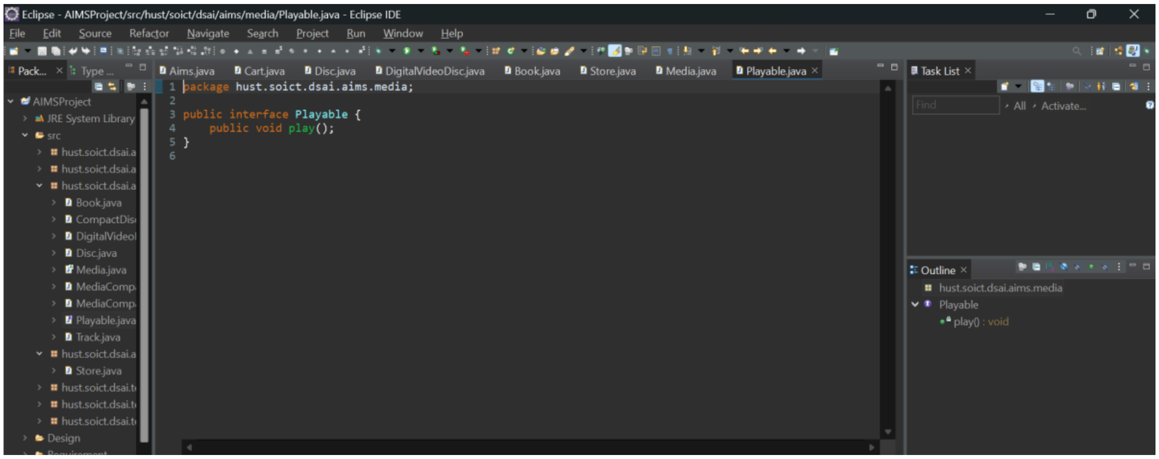
```java
        this.authors = authors;
    }

    public void addAuthor(String authorName) {
        boolean found = false;
        for (String name: authors) {
            if (name.equals(authorName)) {
                found = true;
                break;
            }
        }
        if (found) {
            System.out.println("The author is already in list.");
        } else {
            authors.add(authorName);
            System.out.println("Added successfully.");
        }
    }

    public void removeAuthor(String authorName) {
        boolean found = false;
        for (String name: authors) {
            if (name.equals(authorName)) {
                found = true;
                break;
            }
        }
        if (!found) {
            System.out.println("The author is not in the list.");
        } else {
            authors.remove(authorName);
            System.out.println("Remove successfully.");
        }
    }
}
```

1.9. Track class:


Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/Track.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java Store.java Media.java CompactDisc.java Track.java
Writable Smart Insert 1 : 1 : 0

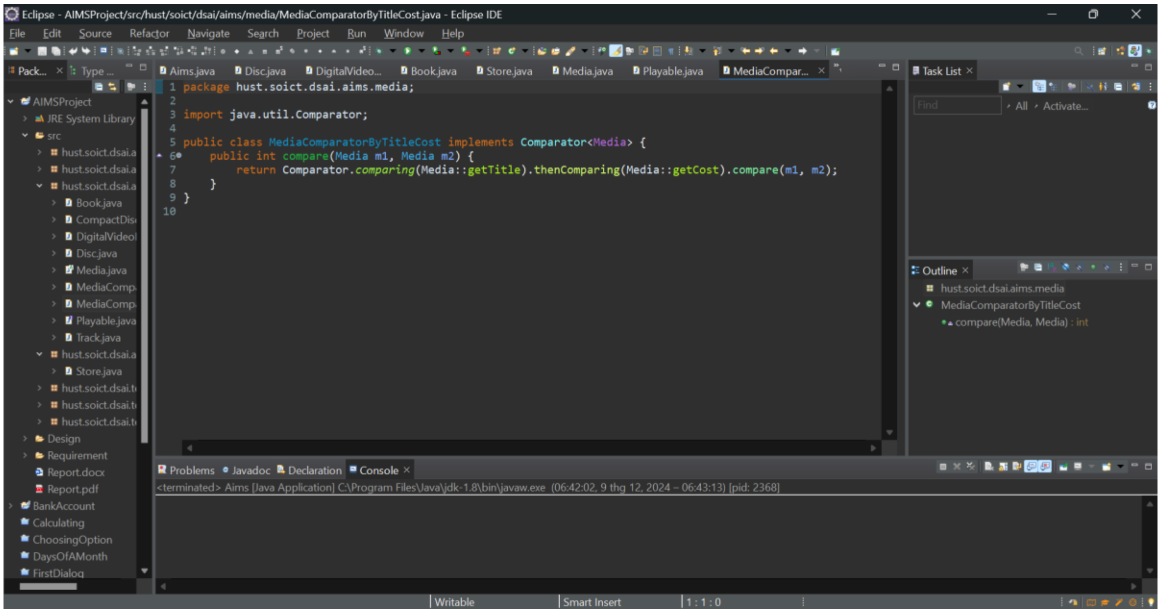
```java
package hust.soict.dsai.aims.media;

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

    public String getTitle() {
        return title;
    }

    public int getLength() {
        return length;
    }

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

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

    public boolean equals(Object obj) {
        if (obj instanceof Track) {
            Track other = (Track) obj;
            return this.title.equals(other.title) && this.length == other.length;
        }
        return false;
    }

}
```

1.10. Interface Playable:



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

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

1.11. MediaComparatorByTitleCost:

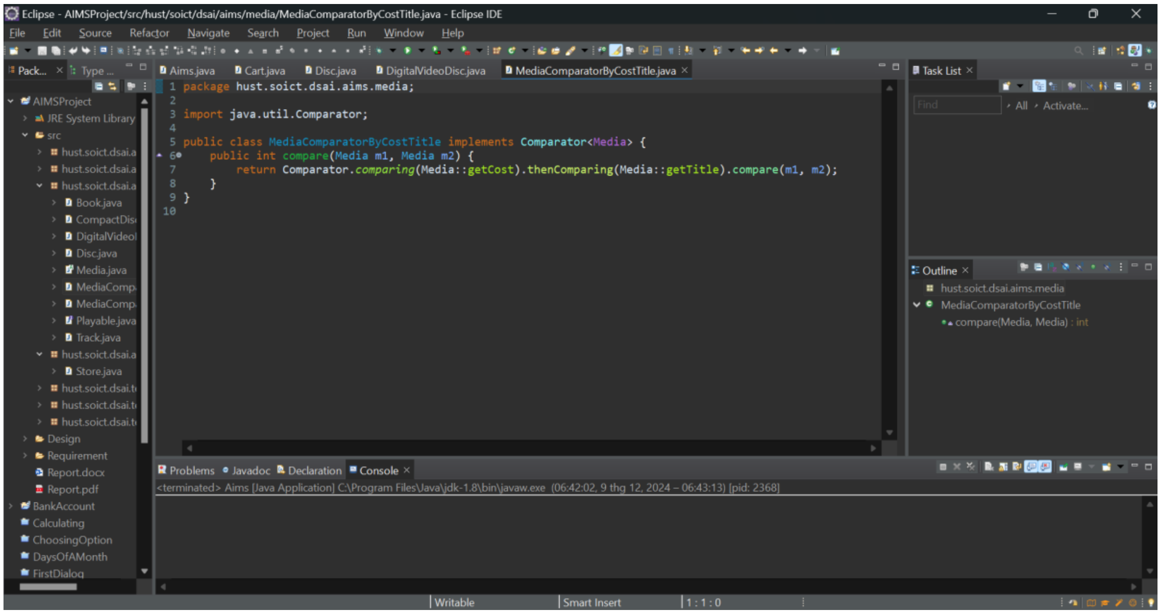


Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/MediaComparatorByTitleCost.java - Eclipse IDE

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

import java.util.Comparator;

public class MediaComparatorByTitleCost implements Comparator<Media> {
    public int compare(Media m1, Media m2) {
        return Comparator.comparing(Media::getTitle).thenComparing(Media::getCost).compare(m1, m2);
    }
}
```

```
<terminated> Aims [Java Application] C:\Program Files\Java\jdk-1.8\bin\javaw.exe  (06:42:02, 9 thg 12, 2024 – 06:43:13) [pid: 2368]
```

1.12. MediaComparatorByCostTitle:



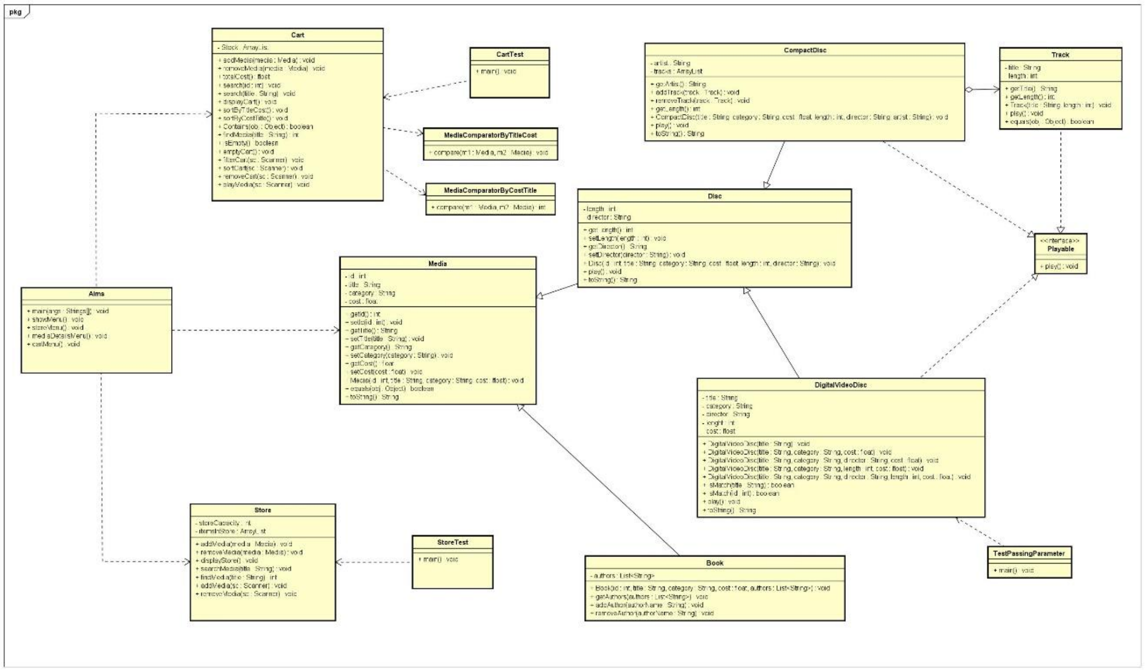
Eclipse - AIMSProject/src/hust/soict/dsai/aims/media/MediaComparatorByCostTitle.java - Eclipse IDE

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

import java.util.Comparator;

public class MediaComparatorByCostTitle implements Comparator<Media> {
    public int compare(Media m1, Media m2) {
        return Comparator.comparing(Media::getCost).thenComparing(Media::getTitle).compare(m1, m2);
    }
}
```

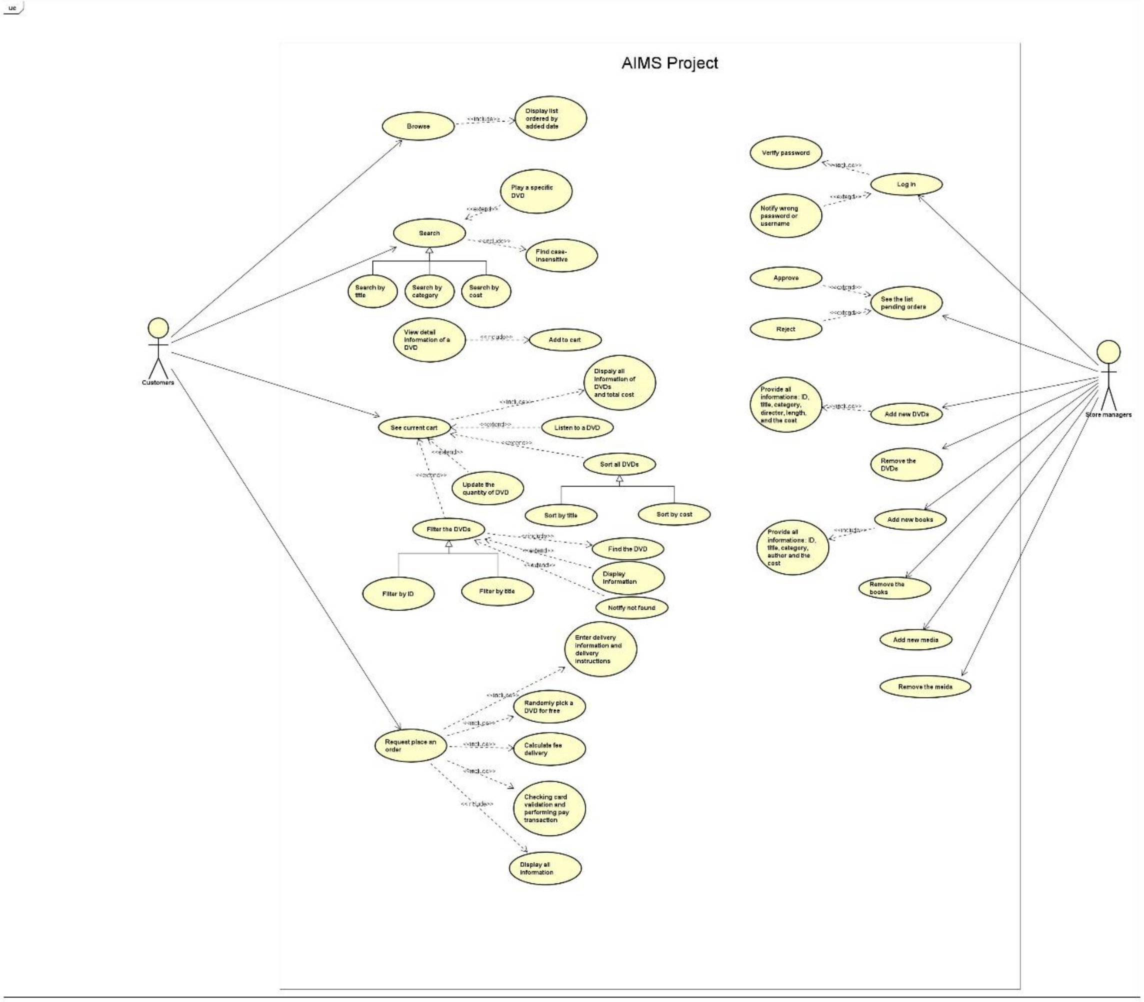
```
<terminated> Aims [Java Application] C:\Program Files\Java\jdk-1.8\bin\javaw.exe  (06:42:02, 9 thg 12, 2024 – 06:43:13) [pid: 2368]
```

2. Diagram:

2.1. Class diagram:


pkg
Cart
- Stock : ArrayList
+ addMedia(media : Media) : void
+ removeMedia(media : Media) : void
+ totalCost() : float
+ search(id : int) : void
+ search(title : String) : void
+ displayCart() : void
+ sortByTitleCost() : void
+ sortByCostTitle() : void
+ Contains(obj : Object) : boolean
+ findMedia(title : String) : int
+ isEmpty() : boolean
+ emptyCart() : void
+ filterCart(sc : Scanner) : void
+ sortCart(sc : Scanner) : void
+ removeCart(sc : Scanner) : void
+ playMedia(sc : Scanner) : void
CartTest
+ main() : void
MediaComparatorByTitleCost
+ compare(m1 : Media, m2 : Media) : void
MediaComparatorByCostTitle
+ compare(m1 : Media, m2 : Media) : int
CompactDisc
- artist : String
- tracks : ArrayList
+ getArtist() : String
+ addTrack(track : Track) : void
+ removeTrack(track : Track) : void
+ getLength() : int
+ CompactDisc(title : String, category : String, cost : float, length : int, director : String, artist : String) : void
+ play() : void
+ toString() : String
Track
- title : String
length : int
+ getTitle() : String
+ getLength() : int
+ Track(title : String, length : int) : void
+ play() : void
+ equals(obj : Object) : boolean
Disc
- length : int
director : String
+ getLength() : int
+ setLength(length : int) : void
+ getDirector() : String
+ setDirector(director : String) : void
+ Disc(id : int, title : String, category : String, cost : float, length : int, director : String) : void
+ play() : void
+ toString() : String
<<interface>>
Playable
+ play() : void
Aims
+ main(args : Strings[]) : void
+ showMenu() : void
+ storeMenu() : void
+ mediaDetailsMenu() : void
+ cartMenu() : void
Media
- id : int
- title : String
- category : String
- cost : float
getId() : int
- setId(id : int) : void
- getTitle() : String
- setTitle(title : String) : void
- getCategory() : String
- setCategory(category : String) : void
- getCost() : float
setCost(cost : float) : void
Media(id : int, title : String, category : String, cost : float) : void
- equals(obj : Object) : boolean
- toString() : String
DigitalVideoDisc
- title : String
- category : String
- director : String
- lenght : int
cost : float
+ DigitalVideoDisc(title : String) : void
+ DigitalVideoDisc(title : String, category : String, cost : float) : void
+ DigitalVideoDisc(title : String, category : String, director : String, cost : float) : void
+ DigitalVideoDisc(title : String, category : String, length : int, cost : float) : void
+ DigitalVideoDisc(title : String, category : String, director : String, length : int, cost : float) : void
+ isMatch(title : String) : boolean
+ isMatch(id : int) : boolean
+ play() : void
+ toString() : String
Store
- storeCapacity : int
- itemsInStore : ArrayList
+ addMedia(media : Media) : void
+ removeMedia(media : Media) : void
+ displayStore() : void
+ searchMedia(title : String) : void
+ findMedia(title : String) : int
+ addMedia(sc : Scanner) : void
+ removeMedia(sc : Scanner) : void
StoreTest
+ main() : void
Book
- authors : List<String>
+ Book(id : int, title : String, category : String, cost : float, authors : List<String>) : void
+ getAuthors(authors : List<String>) : void
+ addAuthor(authorName : String) : void
+ removeAuthor(authorName : String) : void
TestPassingParameter
+ main() : void

2.2. Usecase diagram:



3. Answer:

3.1 Which classes are aggregates of other classes? Checking all constructors of whole classes if they initialize for their parts?

- Cart aggregates Media objects
- Store aggregates Media objects
- CompactDisc aggregates Track objects

3.2 When overriding the equals() method of the Object class, you will have to cast the Object parameter obj to the type of Object that you are dealing with. For example, in the Media class, you must cast the Object obj to a Media, and then check the equality of the two objects' attributes as the above requirements (i.e. title for Media; title and length for Track). If the passing object is not an instance of Media, what happens?

- If the passing object is not an instance of Media, it will return false.

3.3 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 Java docs to see the information of this interface.

Suppose we are taking this Comparable interface approach.

· What class should implement the Comparable interface?

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

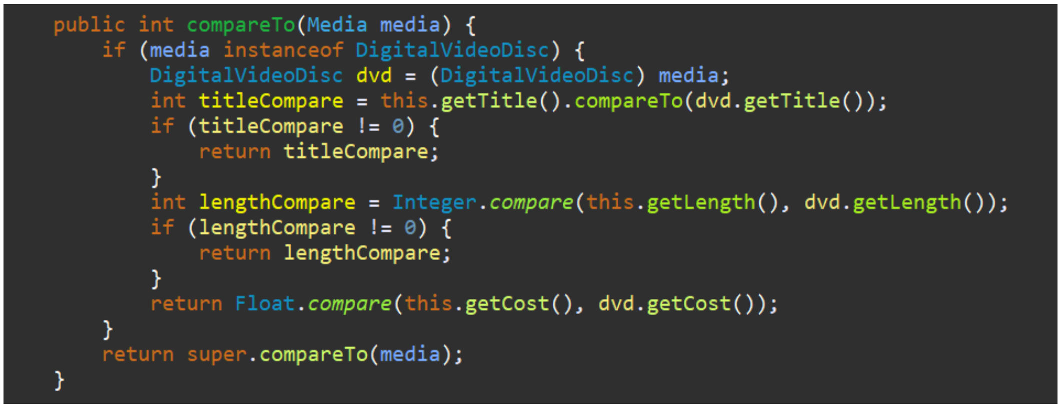
· Can we have two ordering rules of the item (by title then cost and by cost then title) if we use this Comparable interface approach?

· 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?

- Class media should implement the Comparable interface and subclass of media class such as Book, Disc, DVD, CD should also override that method.
- Each class should override the compareTo() method to define the natural ordering, for example by title, by cost, …

- Using Comparable interface, a class can define only one ordering using compareTo() method.

- We should override this method for DVD class:



```
    public int compareTo(Media media) {
        if (media instanceof DigitalVideoDisc) {
            DigitalVideoDisc dvd = (DigitalVideoDisc) media;
            int titleCompare = this.getTitle().compareTo(dvd.getTitle());
            if (titleCompare != 0) {
                return titleCompare;
            }
            int lengthCompare = Integer.compare(this.getLength(), dvd.getLength());
            if (lengthCompare != 0) {
                return lengthCompare;
            }
            return Float.compare(this.getCost(), dvd.getCost());
        }
        return super.compareTo(media);
    }
```