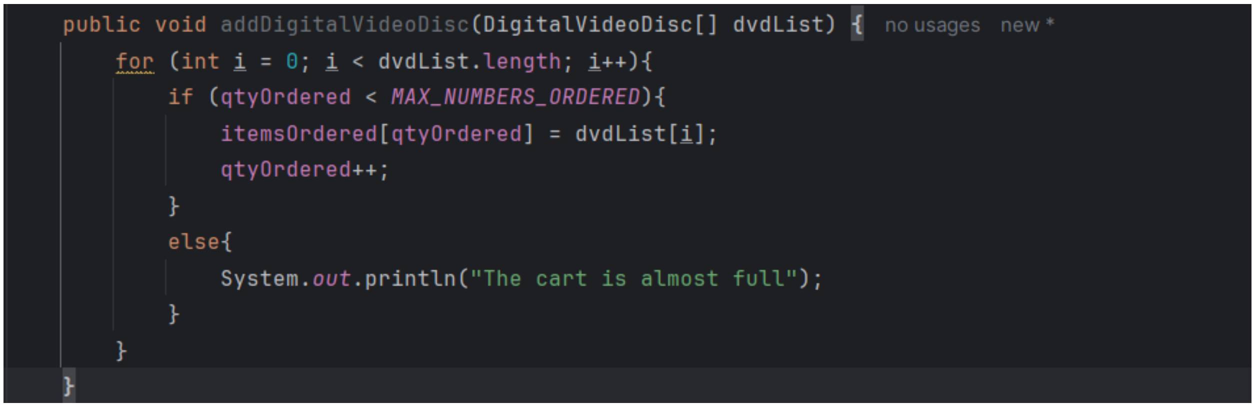
Object-Oriented Programming

Họ và tên: Lê Bá Ngọc Hiểu

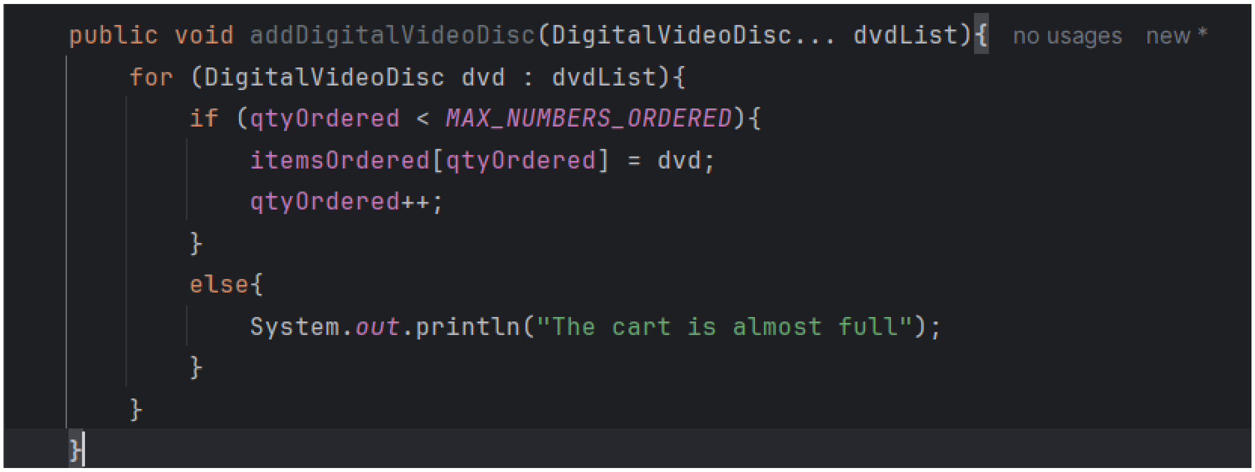
MSSV: 20225627

2. Working with method overloading

2.1 Overloading by differing types of parameter



```java
public void addDigitalVideoDisc(DigitalVideoDisc[] dvdList) {
    for (int i = 0; i < dvdList.length; i++){
        if (qtyOrdered < MAX_NUMBERS_ORDERED){
            itemsOrdered[qtyOrdered] = dvdList[i];
            qtyOrdered++;
        }
        else{
            System.out.println("The cart is almost full");
        }
    }
}
```

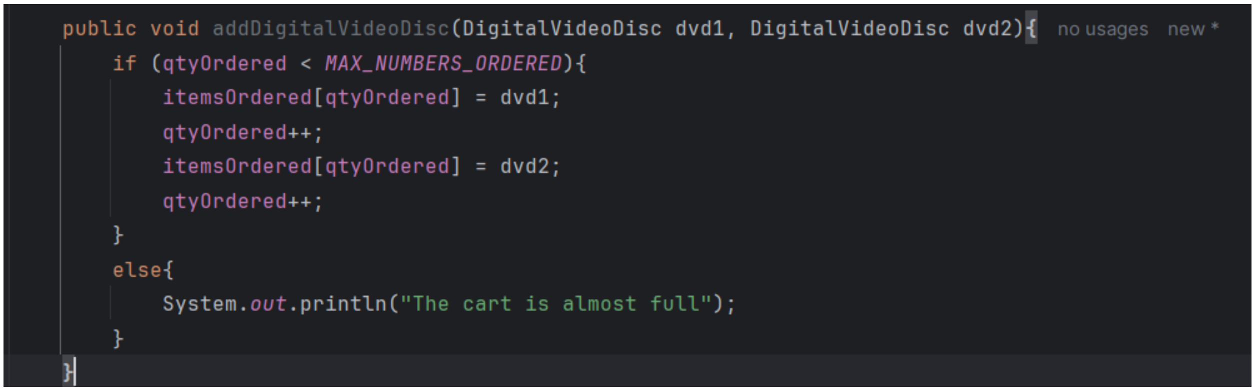


```java
public void addDigitalVideoDisc(DigitalVideoDisc... dvdList){
    for (DigitalVideoDisc dvd : dvdList){
        if (qtyOrdered < MAX_NUMBERS_ORDERED){
            itemsOrdered[qtyOrdered] = dvd;
            qtyOrdered++;
        }
        else{
            System.out.println("The cart is almost full");
        }
    }
}
```

Answer:

In this case, the **varargs method** is preferable due to its simplicity, flexibility, and cleaner syntax. It reduces boilerplate code for the caller and provides a better developer experience. However, the array parameter method could still be useful in scenarios where the data is already in an array format or when working with collections that need to be converted to arrays.

2.2. Overloading by differing the number of parameters



```
public void addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2){
    if (qtyOrdered < MAX_NUMBERS_ORDERED){
        itemsOrdered[qtyOrdered] = dvd1;
        qtyOrdered++;
        itemsOrdered[qtyOrdered] = dvd2;
        qtyOrdered++;
    }
    else{
        System.out.println("The cart is almost full");
    }
}
```

3. Passing parameter

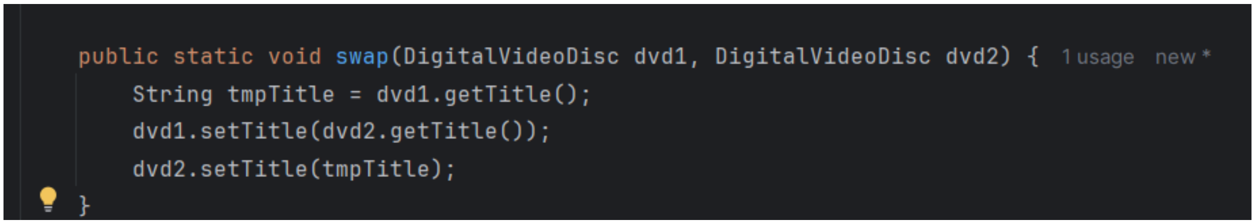
Answer:

Why do the titles of the objects remain unchanged after the call to swap(jungleDVD, cinderellaDVD)?

The swap method does not change the original references jungleDVD and cinderellaDVD. In Java, when you pass objects to a method, you pass the reference by value. Inside the swap method, the parameters o1 and o2 are local variables that hold copies of the original references. Swapping o1 and o2 affects only these local variables, not the original references outside the method. Therefore, jungleDVD and cinderellaDVD still point to their original objects, and their titles remain unchanged.

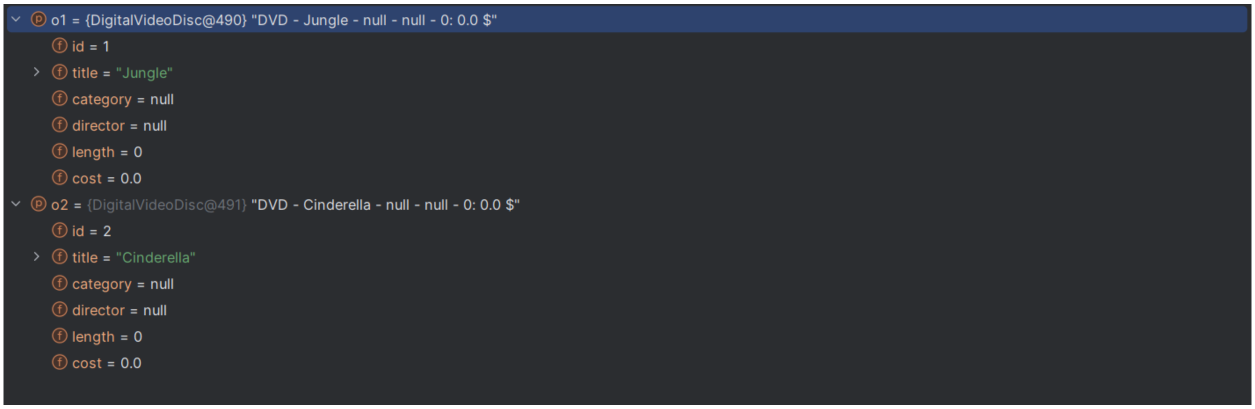
Why is the title of jungleDVD changed after the call to changeTitle(jungleDVD, cinderellaDVD.getTitle())?

In the changeTitle method, the reference dvd points to the same object as jungleDVD. When dvd.setTitle(title) is called, it modifies the title of the object that both dvd and jungleDVD refer to. Although a new DigitalVideoDisc object is created and assigned to dvd later in the method, this reassignment does not affect jungleDVD, as dvd is only a local copy of the reference. Hence, the change to the title persists in the original object referred to by jungleDVD.



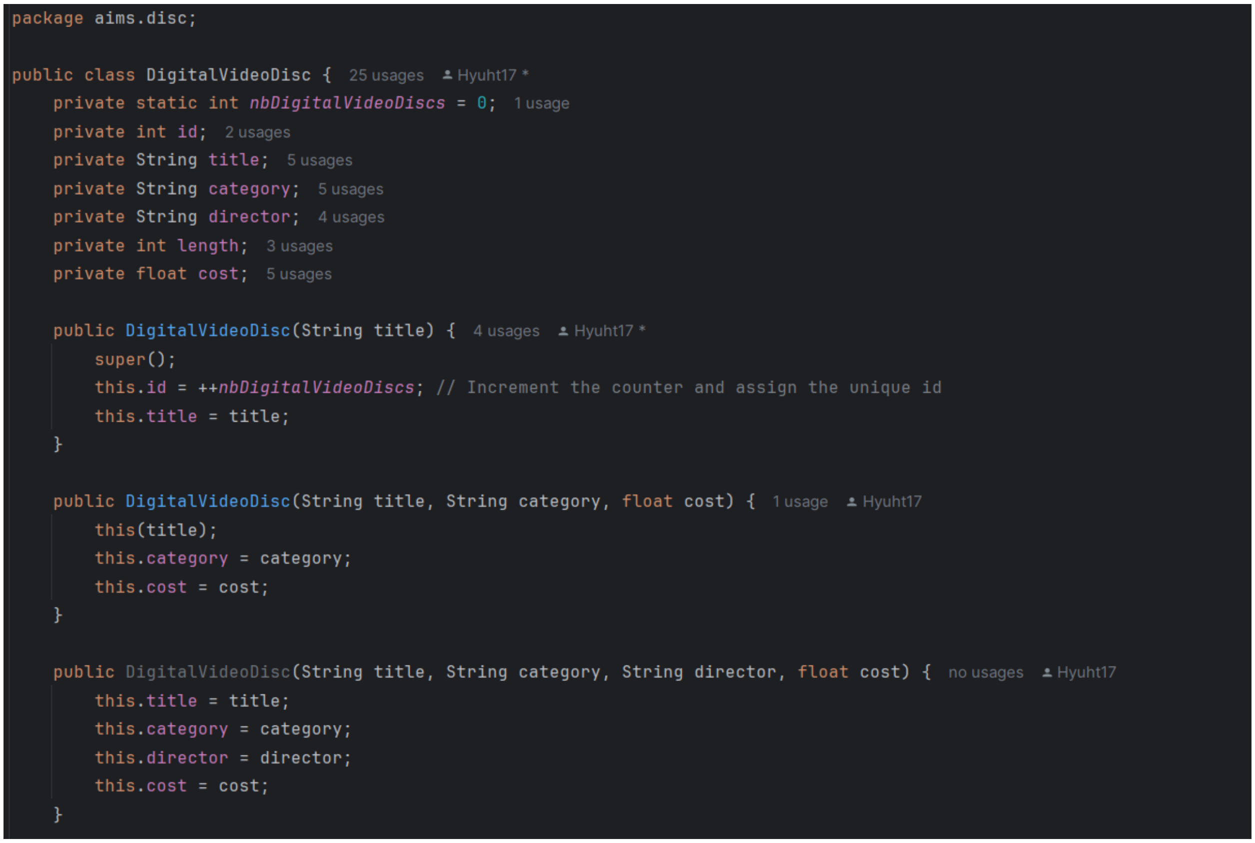
```java
public static void swap(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {
    String tmpTitle = dvd1.getTitle();
    dvd1.setTitle(dvd2.getTitle());
    dvd2.setTitle(tmpTitle);
}
```

4. Use debug run:

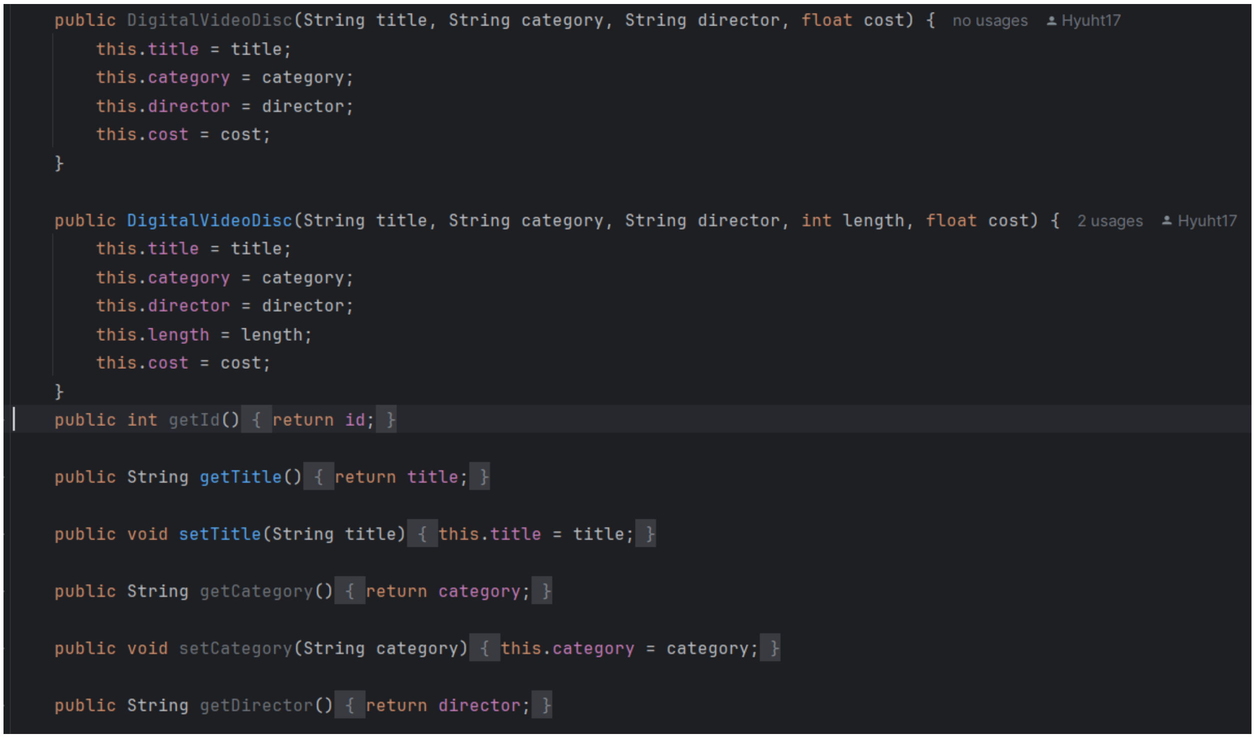


```
o1 = {DigitalVideoDisc@490} "DVD - Jungle - null - null - 0: 0.0 $"
    id = 1
  > title = "Jungle"
    category = null
    director = null
    length = 0
    cost = 0.0
o2 = {DigitalVideoDisc@491} "DVD - Cinderella - null - null - 0: 0.0 $"
    id = 2
  > title = "Cinderella"
    category = null
    director = null
    length = 0
    cost = 0.0
```

5. Classifier Member and Instance Member



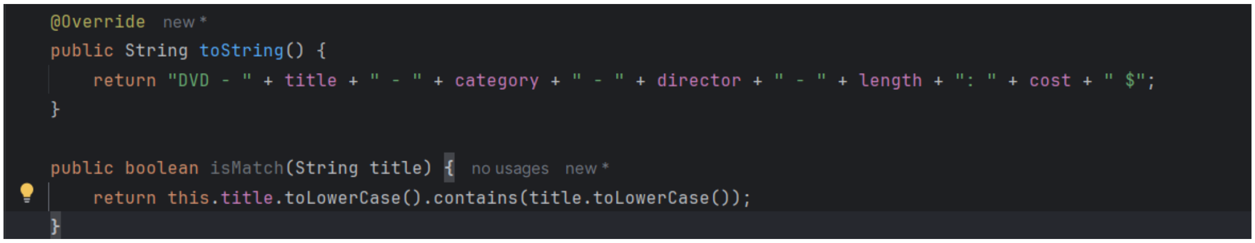
```
package aims.disc;

public class DigitalVideoDisc {  25 usages   Hyuht17 *
    private static int nbDigitalVideoDiscs = 0;  1 usage
    private int id;  2 usages
    private String title;  5 usages
    private String category;  5 usages
    private String director;  4 usages
    private int length;  3 usages
    private float cost;  5 usages

    public DigitalVideoDisc(String title) {  4 usages   Hyuht17 *
        super();
        this.id = ++nbDigitalVideoDiscs; // Increment the counter and assign the unique id
        this.title = title;
    }

    public DigitalVideoDisc(String title, String category, float cost) {  1 usage   Hyuht17
        this(title);
        this.category = category;
        this.cost = cost;
    }

    public DigitalVideoDisc(String title, String category, String director, float cost) {  no usages   Hyuht17
        this.title = title;
        this.category = category;
        this.director = director;
        this.cost = cost;
    }
```



```
    public DigitalVideoDisc(String title, String category, String director, float cost) {  no usages   Hyuht17
        this.title = title;
        this.category = category;
        this.director = director;
        this.cost = cost;
    }

    public DigitalVideoDisc(String title, String category, String director, int length, float cost) {  2 usages   Hyuht17
        this.title = title;
        this.category = category;
        this.director = director;
        this.length = length;
        this.cost = cost;
    }
    public int getId() { return 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 String getDirector() { return director; }
```

6. Open the `Cart` class

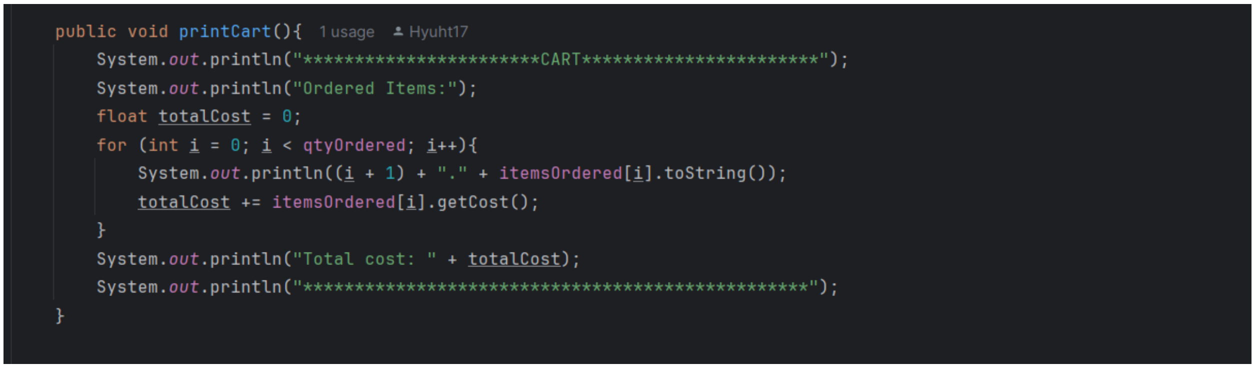
In DigitalVideoDisc class:



```
@Override  new *
public String toString() {
    return "DVD - " + title + " - " + category + " - " + director + " - " + length + ": " + cost + " $";
}

public boolean isMatch(String title) {  no usages  new *
    return this.title.toLowerCase().contains(title.toLowerCase());
}
```

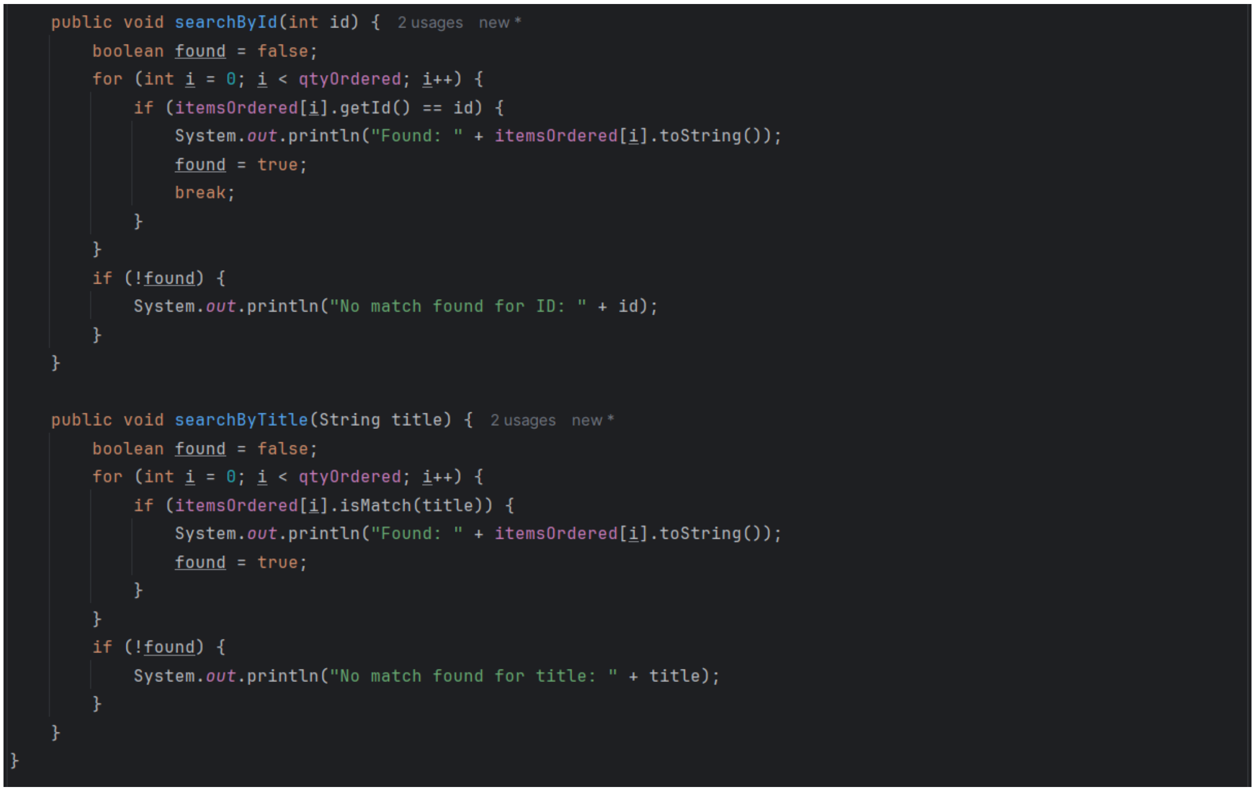
In cart class:

Print cart function:



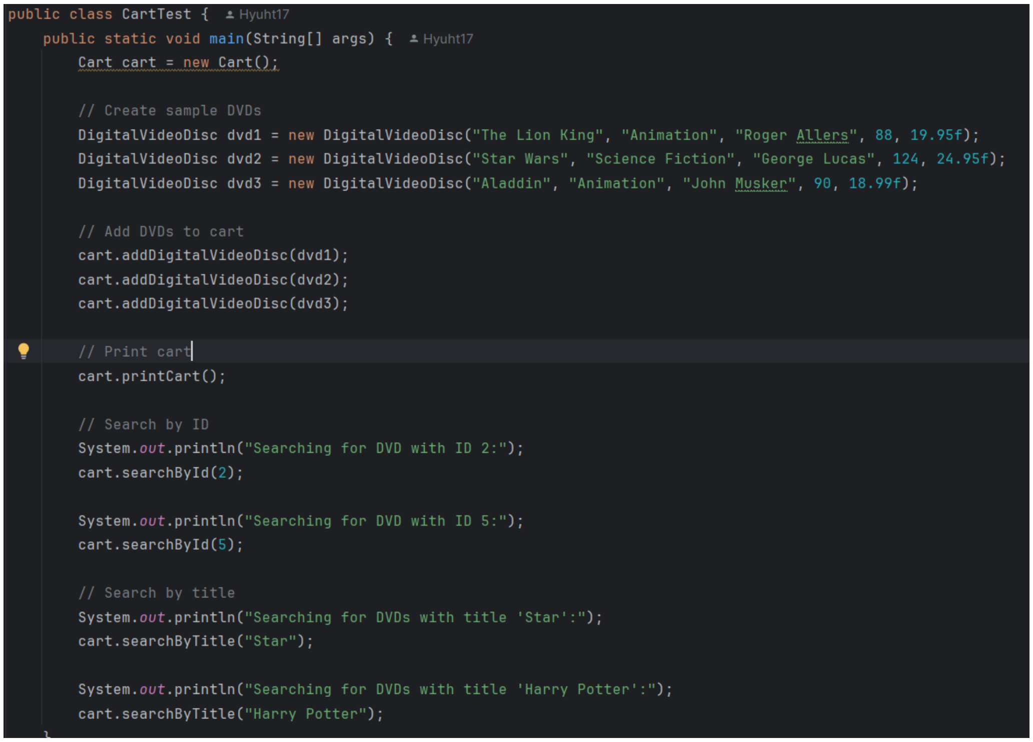
```
public void printCart(){  1 usage  Hyuht17
    System.out.println("***********************CART************************");
    System.out.println("Ordered Items:");
    float totalCost = 0;
    for (int i = 0; i < qtyOrdered; i++){
        System.out.println((i + 1) + "." + itemsOrdered[i].toString());
        totalCost += itemsOrdered[i].getCost();
    }
    System.out.println("Total cost: " + totalCost);
    System.out.println("**************************************************");
}
```

Search function:



```
    public void searchById(int id) {  2 usages   new *
        boolean found = false;
        for (int i = 0; i < qtyOrdered; i++) {
            if (itemsOrdered[i].getId() == id) {
                System.out.println("Found: " + itemsOrdered[i].toString());
                found = true;
                break;
            }
        }
        if (!found) {
            System.out.println("No match found for ID: " + id);
        }
    }

    public void searchByTitle(String title) {  2 usages   new *
        boolean found = false;
        for (int i = 0; i < qtyOrdered; i++) {
            if (itemsOrdered[i].isMatch(title)) {
                System.out.println("Found: " + itemsOrdered[i].toString());
                found = true;
            }
        }
        if (!found) {
            System.out.println("No match found for title: " + title);
        }
    }
}
```

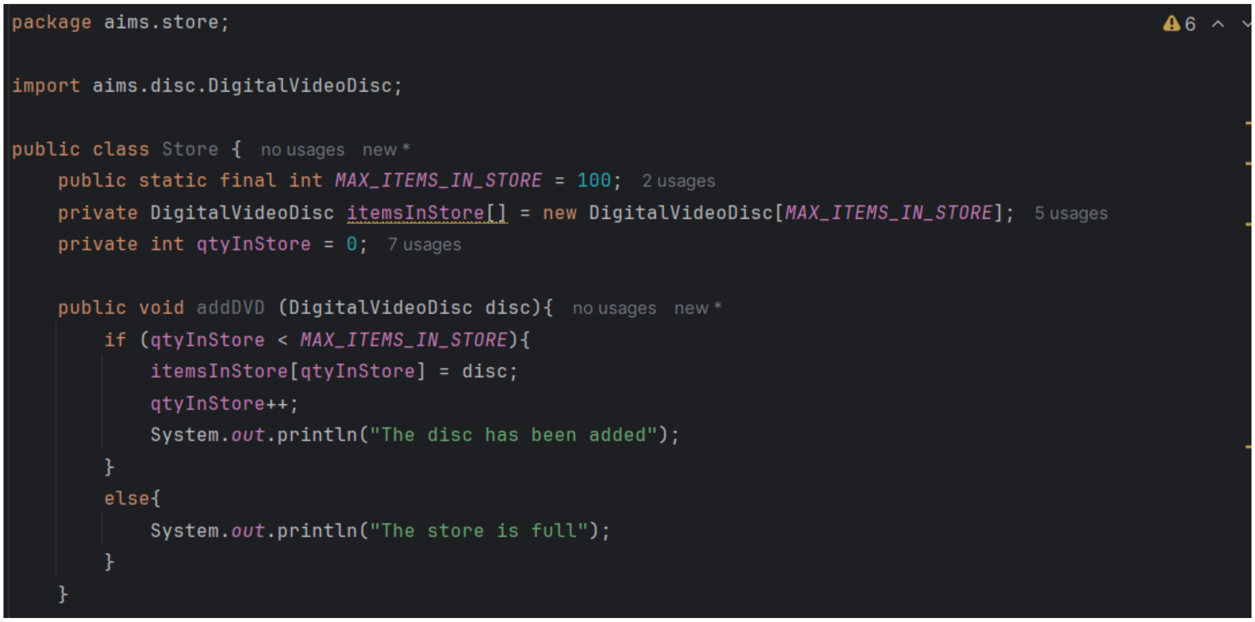
CartTest:



```
public class CartTest {  Hyuht17
    public static void main(String[] args) {  Hyuht17
        Cart cart = new Cart();

        // Create sample DVDs
        DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 88, 19.95f);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 124, 24.95f);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladdin", "Animation", "John Musker", 90, 18.99f);

        // Add DVDs to cart
        cart.addDigitalVideoDisc(dvd1);
        cart.addDigitalVideoDisc(dvd2);
        cart.addDigitalVideoDisc(dvd3);

        // Print cart
        cart.printCart();

        // Search by ID
        System.out.println("Searching for DVD with ID 2:");
        cart.searchById(2);

        System.out.println("Searching for DVD with ID 5:");
        cart.searchById(5);

        // Search by title
        System.out.println("Searching for DVDs with title 'Star':");
        cart.searchByTitle("Star");

        System.out.println("Searching for DVDs with title 'Harry Potter':");
        cart.searchByTitle("Harry Potter");
    }
```

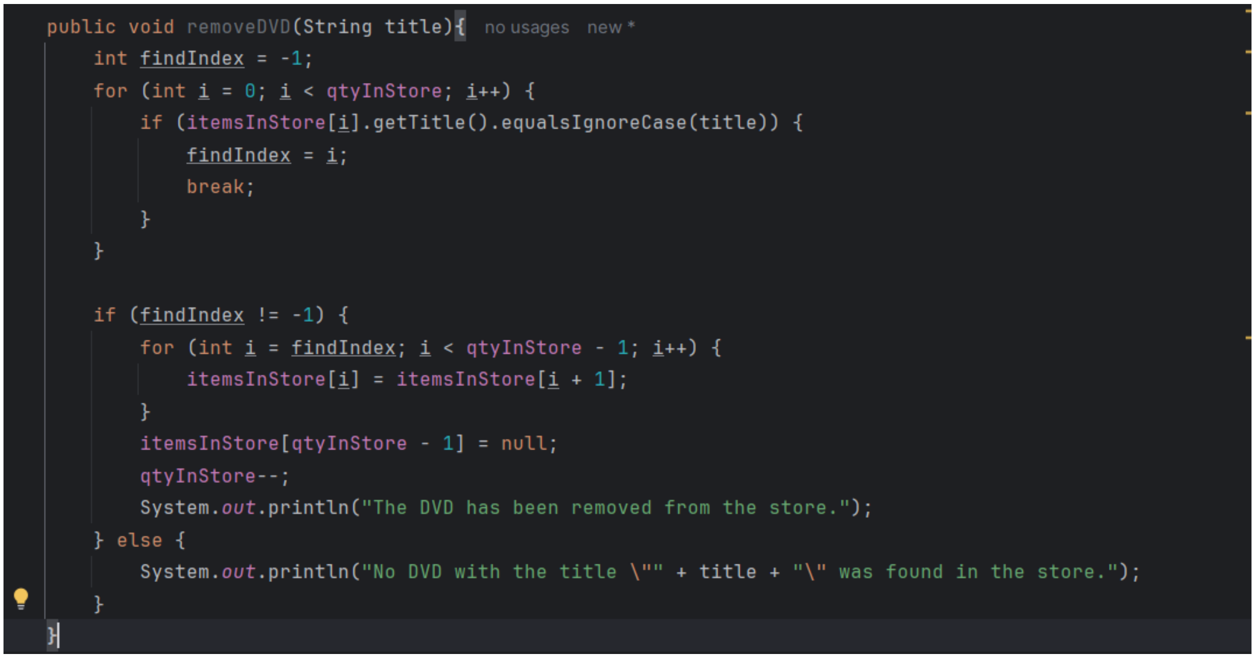
7. Implement the `Store` class

Add DVD:



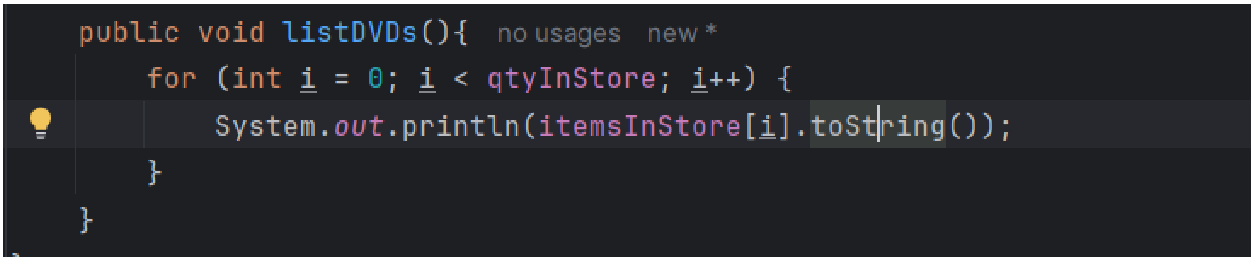
```
package aims.store;

import aims.disc.DigitalVideoDisc;

public class Store {
    public static final int MAX_ITEMS_IN_STORE = 100;
    private DigitalVideoDisc itemsInStore[] = new DigitalVideoDisc[MAX_ITEMS_IN_STORE];
    private int qtyInStore = 0;

    public void addDVD (DigitalVideoDisc disc){
        if (qtyInStore < MAX_ITEMS_IN_STORE){
            itemsInStore[qtyInStore] = disc;
            qtyInStore++;
            System.out.println("The disc has been added");
        }
        else{
            System.out.println("The store is full");
        }
    }
```

Remove DVD:



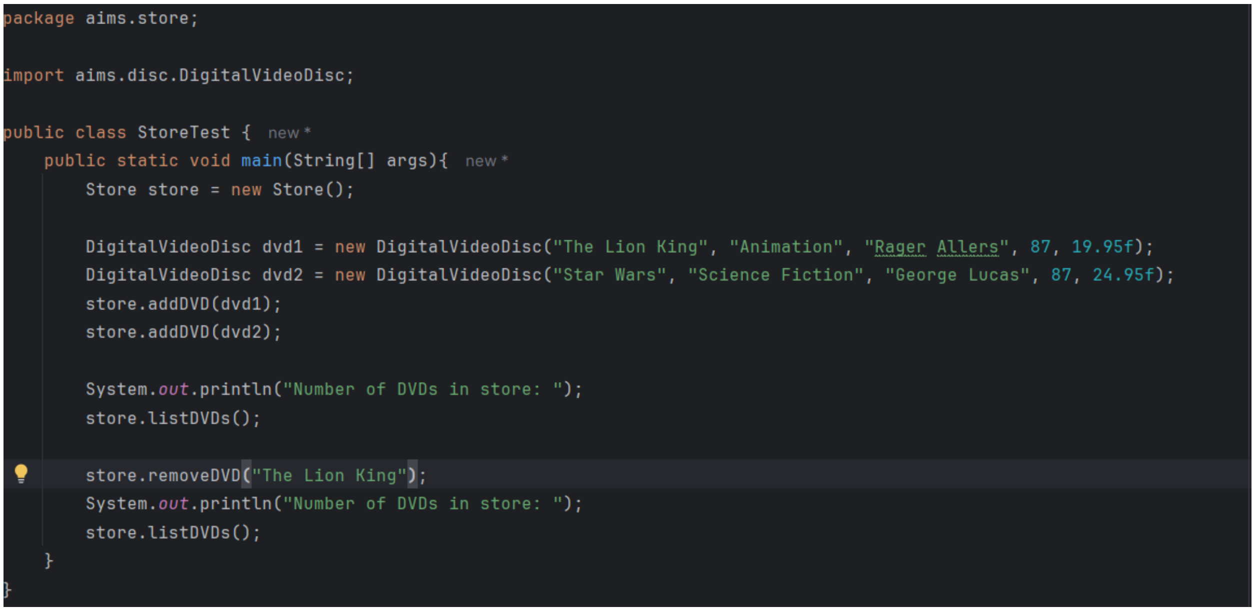
```
    public void removeDVD(String title){
        int findIndex = -1;
        for (int i = 0; i < qtyInStore; i++) {
            if (itemsInStore[i].getTitle().equalsIgnoreCase(title)) {
                findIndex = i;
                break;
            }
        }

        if (findIndex != -1) {
            for (int i = findIndex; i < qtyInStore - 1; i++) {
                itemsInStore[i] = itemsInStore[i + 1];
            }
            itemsInStore[qtyInStore - 1] = null;
            qtyInStore--;
            System.out.println("The DVD has been removed from the store.");
        } else {
            System.out.println("No DVD with the title \"" + title + "\" was found in the store.");
        }
    }
```

List DVD in Store:



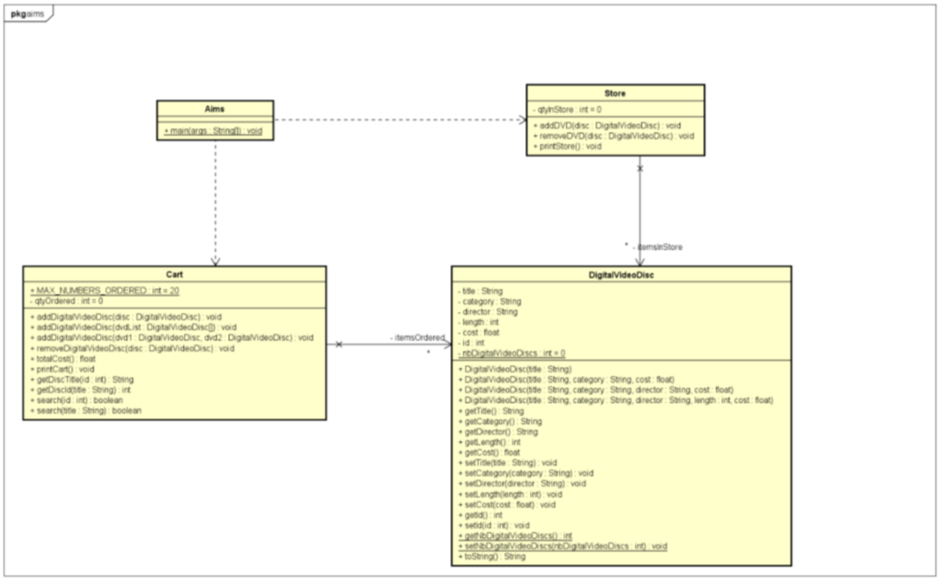
```
public void listDVDs(){  no usages  new *
    for (int i = 0; i < qtyInStore; i++) {
        System.out.println(itemsInStore[i].toString());
    }
}
```

Store Test:



```
package aims.store;

import aims.disc.DigitalVideoDisc;

public class StoreTest {  new *
    public static void main(String[] args){  new *
        Store store = new Store();

        DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King", "Animation", "Rager Allers", 87, 19.95f);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f);
        store.addDVD(dvd1);
        store.addDVD(dvd2);

        System.out.println("Number of DVDs in store: ");
        store.listDVDs();

        store.removeDVD("The Lion King");
        System.out.println("Number of DVDs in store: ");
        store.listDVDs();
    }
}
```

New Class Diagram:


pkg aims
Aims
+ main(args : String[]) : void
Store
- qtyInStore : int = 0
+ addDVD(disc : DigitalVideoDisc) : void
+ removeDVD(disc : DigitalVideoDisc) : void
+ printStore() : void
* - itemsInStore
Cart
+ MAX_NUMBERS_ORDERED : int = 20
- qtyOrdered : int = 0
+ addDigitalVideoDisc(disc : DigitalVideoDisc) : void
+ addDigitalVideoDisc(dvdList : DigitalVideoDisc[]) : void
+ addDigitalVideoDisc(dvd1 : DigitalVideoDisc, dvd2 : DigitalVideoDisc) : void
+ removeDigitalVideoDisc(disc : DigitalVideoDisc) : void
+ totalCost() : float
+ printCart() : void
+ getDiscTitle(id : int) : String
+ getDiscId(title : String) : int
+ search(id : int) : boolean
+ search(title : String) : boolean
- itemsOrdered
*
DigitalVideoDisc
- title : String
- category : String
- director : String
- length : int
- cost : float
- id : int
- nbDigitalVideoDiscs : int = 0
+ DigitalVideoDisc(title : String)
+ DigitalVideoDisc(title : String, category : String, cost : float)
+ DigitalVideoDisc(title : String, category : String, director : String, cost : float)
+ DigitalVideoDisc(title : String, category : String, director : String, length : int, cost : float)
+ getTitle() : String
+ getCategory() : String
+ getDirector() : String
+ getLength() : int
+ getCost() : float
+ setTitle(title : String) : void
+ setCategory(category : String) : void
+ setDirector(director : String) : void
+ setLength(length : int) : void
+ setCost(cost : float) : void
+ getId() : int
+ setId(id : int) : void
+ getNbDigitalVideoDiscs() : int
+ setNbDigitalVideoDiscs(nbDigitalVideoDiscs : int) : void
+ toString() : String