

# BÁO CÁO THỰC HÀNH LAB 02

## LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

### Mục lục

1. Đề bài.....	1
2. Yêu cầu.....	1
3. Use case diagram.....	2
4. Class diagram.....	2
5. Mã nguồn của chương trình.....	3
a. Create Aims class.....	3
b. Create the DigitalVideoDisc class and its attributes.....	3
c. Create accessors and mutators for the class DigitalVideoDisc.....	4
d. Create Constructor method.....	6
e. Create the Cart class to work with DigitalVideoDisc.....	7
f. Create Carts of DigitalVideoDiscs.....	8
g. Removing items from the cart.....	9
6. Demo chương trình.....	11

### Danh sách hình ảnh

Figure 1: Biểu đồ use case.....	3
Figure 2: Class diagram.....	4
Figure 3: Create Aims class.....	4
Figure 4: Create the DigitalVideoDisc class and its attributes.....	5
Figure 5: Create accessors and mutators for the class DigitalVideoDisc.....	6
Figure 6: Create Constructor method.....	7
Figure 7: Create the Cart class to work with DigitalVideoDisc.....	8
Figure 8: Create Carts of DigitalVideoDiscs.....	9
Figure 9: Create Carts of DigitalVideoDiscs.....	10
Figure 10: Kết quả.....	10
Figure 11: Removing items from the cart.....	11
Figure 12: Kết quả sau khi remove.....	11

## 1. Đề bài

Thiết kế hệ thống mới cho dự án AIMS(hiện tại chỉ có một phương tiện : DVD)

## 2. Yêu cầu

- **Đối với người mua hàng**
  - Duyệt danh sách các DVD có sẵn trong cửa hàng
  - tìm kiếm DVD theo : tiêu đề , danh mục , giá cả

- xem thông tin chi tiết của 1 DVD
- thêm DVD vào giỏ
- xem giỏ hàng
- sắp xếp DVD trong giỏ hàng theo tiêu đề hoặc chi phí
- cập nhật số lượng DVD trong giỏ hàng
- đặt hàng
- **Đối với người quản lý cửa hàng**
  - Đăng nhập kiểm tra quyền
  - xem danh sách các đơn hàng đang chờ xử lí
  - xem chi tiết 1 đơn hàng và xử lý nó (đồng ý / từ chối)
  - thêm , xóa DVD

### 3. Use case diagram

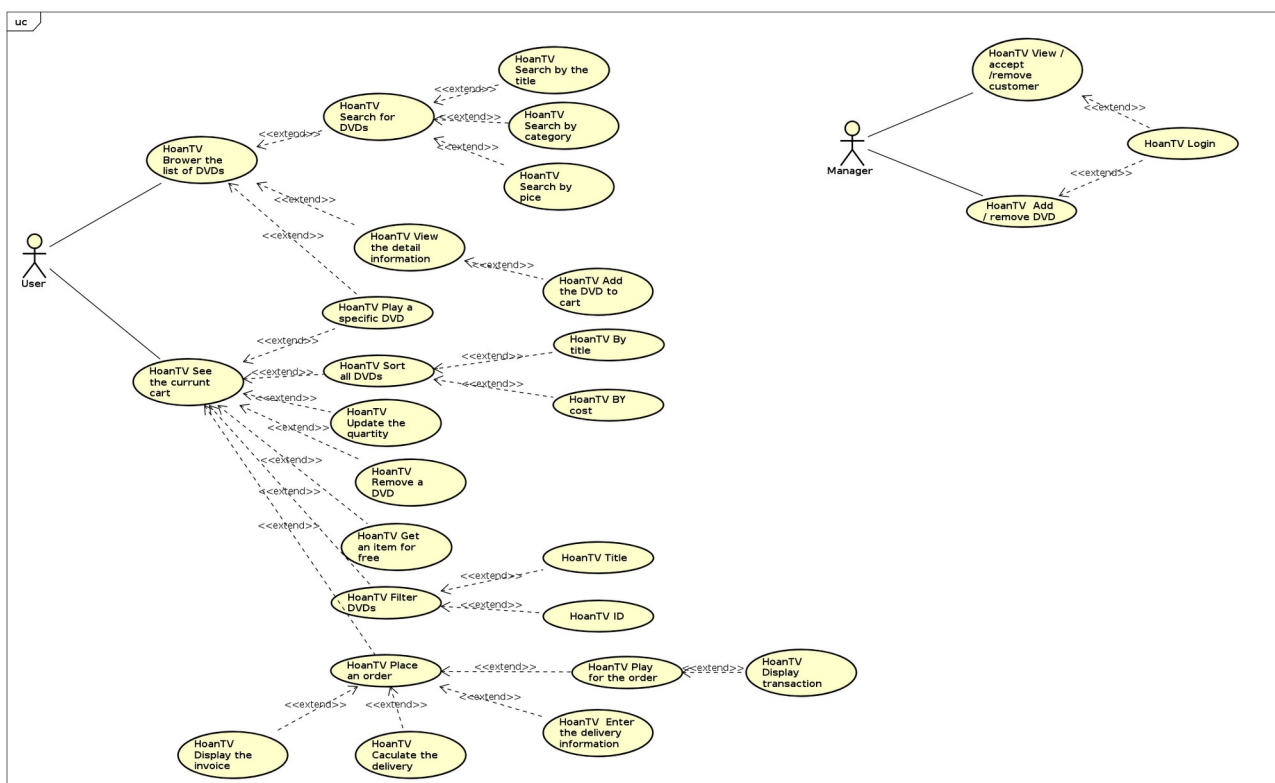


Figure 1: Biểu đồ use case

### 4. Class diagram

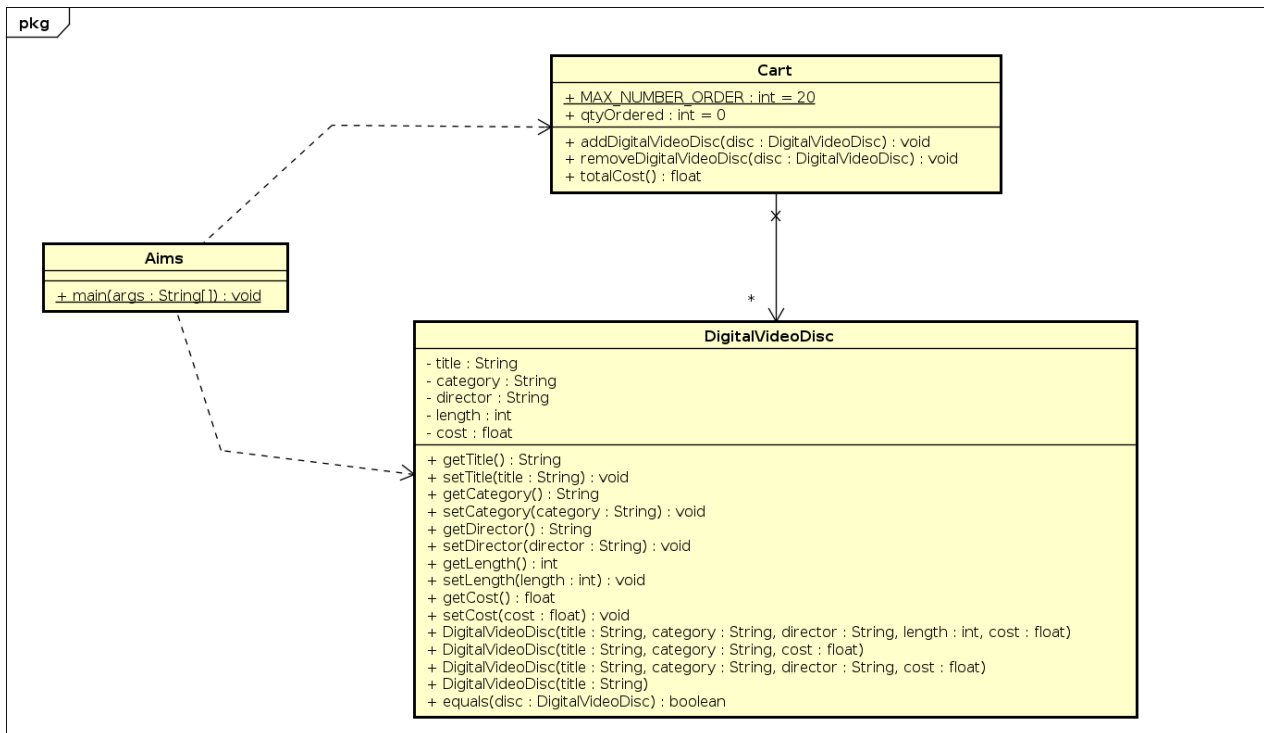


Figure 2: Class diagram

## 5. Mã nguồn của chương trình.

### a. Create Aims class

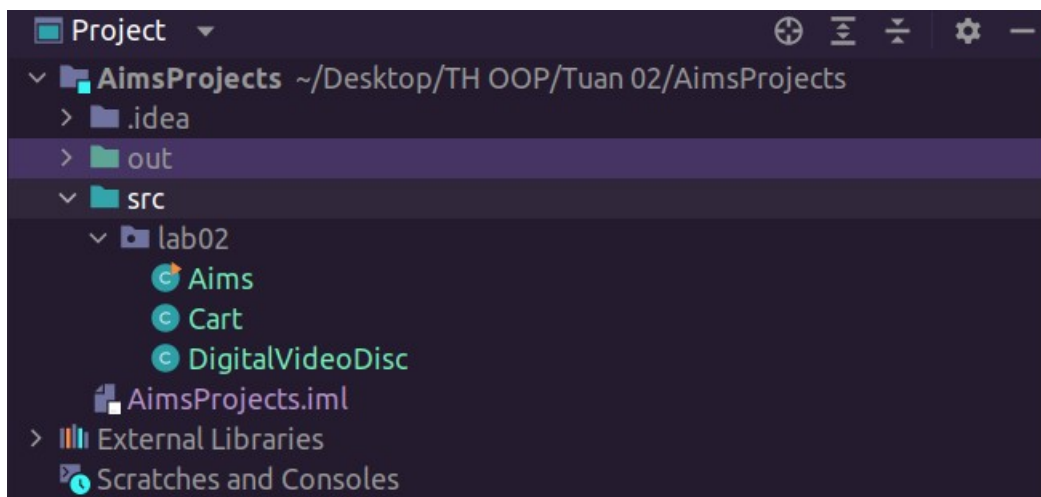
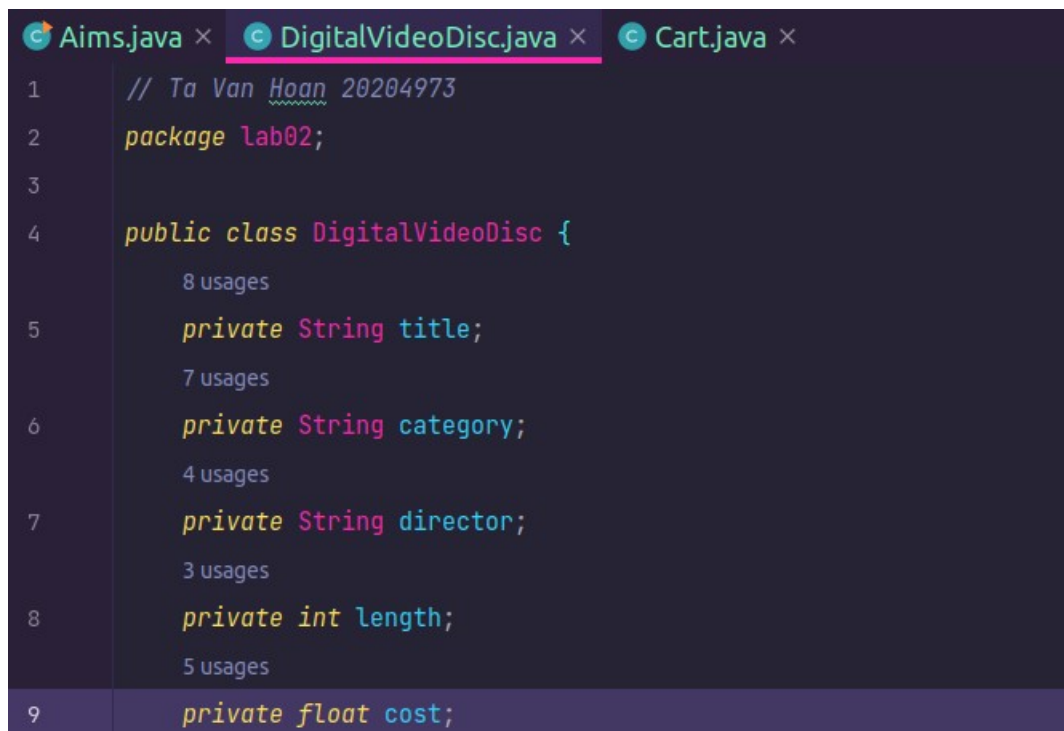


Figure 3: Create Aims class

### b. Create the DigitalVideoDisc class and its attributes

A screenshot of an IDE with three tabs: Aims.java, DigitalVideoDisc.java (selected), and Cart.java. The code in DigitalVideoDisc.java is as follows:

```
1 // Ta Van Hoan 20204973
2 package lab02;
3
4 public class DigitalVideoDisc {
5     private String title;
6     private String category;
7     private String director;
8     private int length;
9     private float cost;
```

Usage statistics are shown for some attributes: 'title' (8 usages), 'category' (7 usages), 'director' (4 usages), 'length' (3 usages), and 'cost' (5 usages).

Figure 4: Create the DigitalVideoDisc class and its attributes

**c.**Create accessors and mutators for the class DigitalVideoDisc

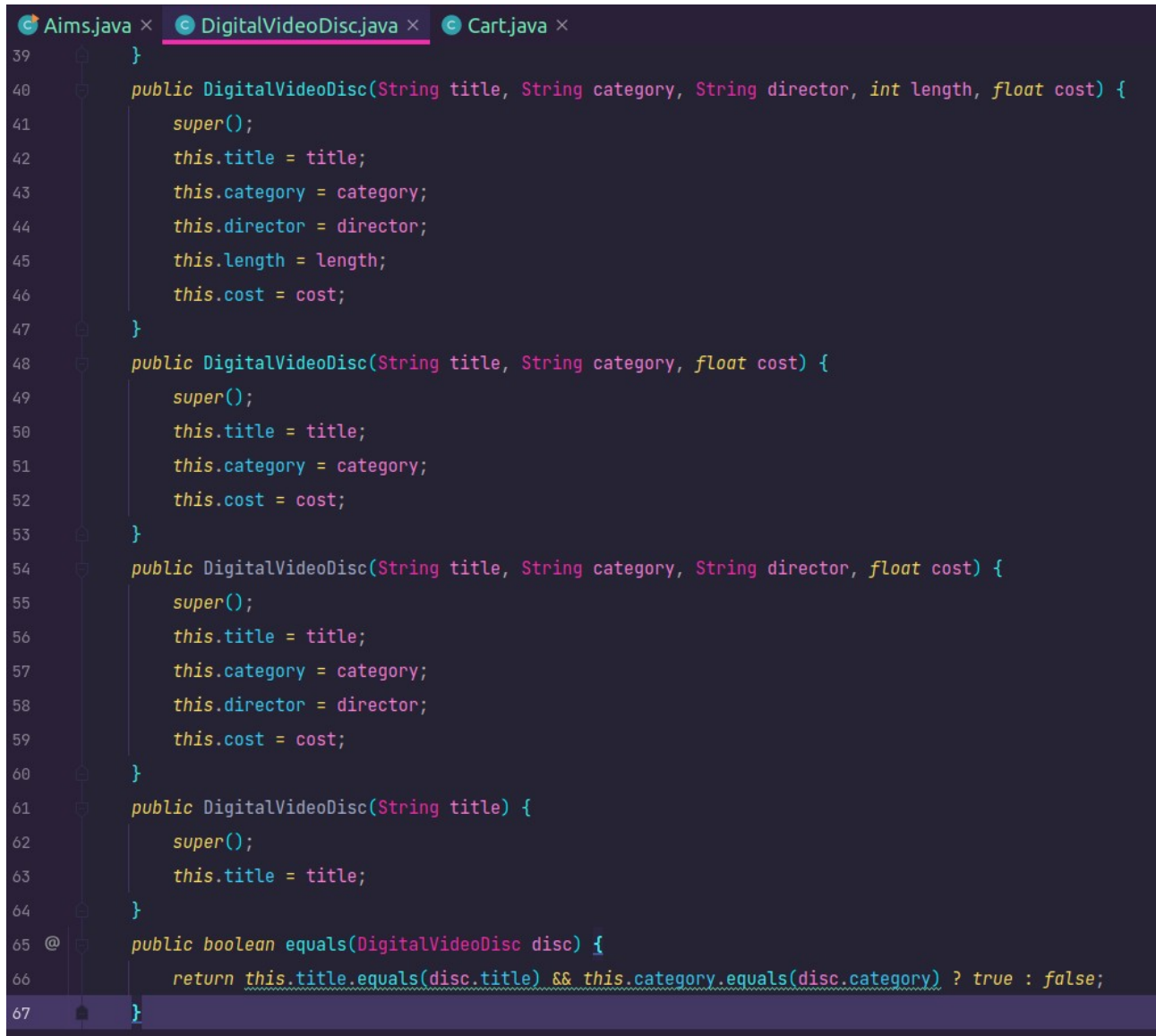


```
10 public String getTitle() {
11     return title;
12 }
13 public void setTitle(String title) {
14     this.title = title;
15 }
16 public String getCategory() {
17     return category;
18 }
19 public void setCategory(String category) {
20     this.category = category;
21 }
22 public String getDirector() {
23     return director;
24 }
25 public void setDirector(String director) {
26     this.director = director;
27 }
28 public int getLength() {
29     return length;
30 }
31 public void setLength(int length) {
32     this.length = length;
33 }
34 public float getCost() {
35     return cost;
36 }
37 public void setCost(float cost) {
38     this.cost = cost;
39 }
```

**Reading Assignment:** When should accessor methods be used?

**Answer:** We need getters and setters or accessors and mutators to protect sensitive information in a class. The information is protected from Illegal use by using these standard methods.

## d.Create Constructor method



```
39  }
40  public DigitalVideoDisc(String title, String category, String director, int length, float cost) {
41      super();
42      this.title = title;
43      this.category = category;
44      this.director = director;
45      this.length = length;
46      this.cost = cost;
47  }
48  public DigitalVideoDisc(String title, String category, float cost) {
49      super();
50      this.title = title;
51      this.category = category;
52      this.cost = cost;
53  }
54  public DigitalVideoDisc(String title, String category, String director, float cost) {
55      super();
56      this.title = title;
57      this.category = category;
58      this.director = director;
59      this.cost = cost;
60  }
61  public DigitalVideoDisc(String title) {
62      super();
63      this.title = title;
64  }
65  @ public boolean equals(DigitalVideoDisc disc) {
66      return this.title.equals(disc.title) && this.category.equals(disc.category) ? true : false;
67  }
```

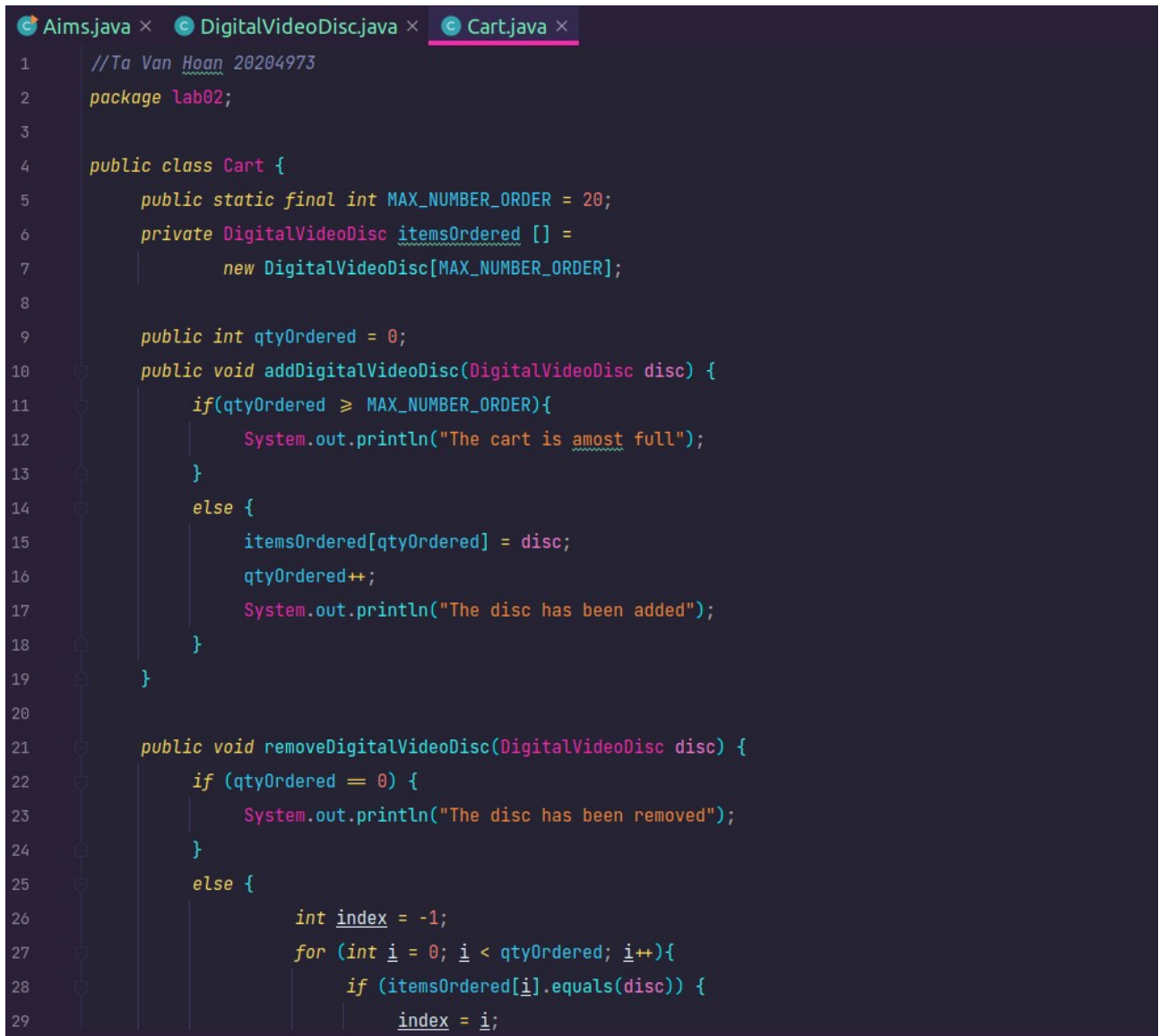
Figure 6: Create Constructor method

**Question:** If you create a constructor method to build a DVD by title then create a constructor method to build a DVD by category. Does JAVA allow you to do this?

**Answer:** Yes, we can do this. This is an overload.



### e. Create the Cart class to work with DigitalVideoDisc



```
1 //Ta Van Hoan 20204973
2 package lab02;
3
4 public class Cart {
5     public static final int MAX_NUMBER_ORDER = 20;
6     private DigitalVideoDisc itemsOrdered [] =
7         new DigitalVideoDisc[MAX_NUMBER_ORDER];
8
9     public int qtyOrdered = 0;
10    public void addDigitalVideoDisc(DigitalVideoDisc disc) {
11        if(qtyOrdered >= MAX_NUMBER_ORDER){
12            System.out.println("The cart is almost full");
13        }
14        else {
15            itemsOrdered[qtyOrdered] = disc;
16            qtyOrdered++;
17            System.out.println("The disc has been added");
18        }
19    }
20
21    public void removeDigitalVideoDisc(DigitalVideoDisc disc) {
22        if (qtyOrdered == 0) {
23            System.out.println("The disc has been removed");
24        }
25        else {
26            int index = -1;
27            for (int i = 0; i < qtyOrdered; i++){
28                if (itemsOrdered[i].equals(disc)) {
29                    index = i;
```

Figure 7: Create the Cart class to work with DigitalVideoDisc

```
44
45     public float totalCost() {
46         float total = 0;
47         for (int i = 0; i < qtyOrdered; i++){
48             total += itemsOrdered[i].getCost();
49         }
50     }
51     return total;
52 }
53 }
```

## f. Create Carts of DigitalVideoDiscs

```
30     }
31 }
32 if(index != -1) {
33     for(int i = 0; i < qtyOrdered; i++){
34         itemsOrdered[i] = itemsOrdered[i+1];
35     }
36     System.out.println("The disc has been removed");
37     qtyOrdered--;
38 }
39 else {
40     System.out.println("can't find the disc");
41 }
42 }
43 }
44 }
```

Figure 8: Create Carts of DigitalVideoDiscs

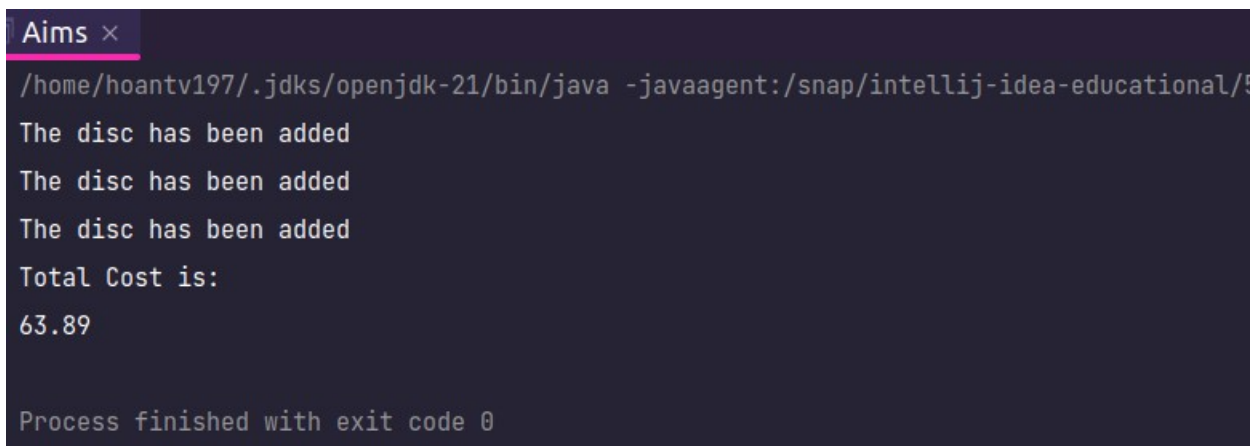




```
1 // Ta Van Hoan 20204973
2 package lab02;
3
4 public class Aims {
5
6     public static void main(String[] args) {
7
8         //Create a new cart
9         Cart anOrder = new Cart();
10
11         //Create new dvd object and add them to the cart
12         DigitalVideoDisc dvd1 = new DigitalVideoDisc( title: "The Lion King", category: "Animation", director: "Roger Allers", length: 87, cost: 19.95f);
13         anOrder.addDigitalVideoDisc(dvd1);
14
15         DigitalVideoDisc dvd2 = new DigitalVideoDisc( title: "Star Wars" , category: "Science Fiction", director: "George Lucas", length: 87, cost: 24.95f);
16         anOrder.addDigitalVideoDisc(dvd2);
17
18         DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin", category: "Animation", cost: 18.99f);
19         anOrder.addDigitalVideoDisc(dvd3);
20
21         System.out.println("Total Cost is: ");
22         System.out.println(anOrder.totalCost());
23
24     }
25 }
```

Figure 9: Create Carts of DigitalVideoDiscs

Result :

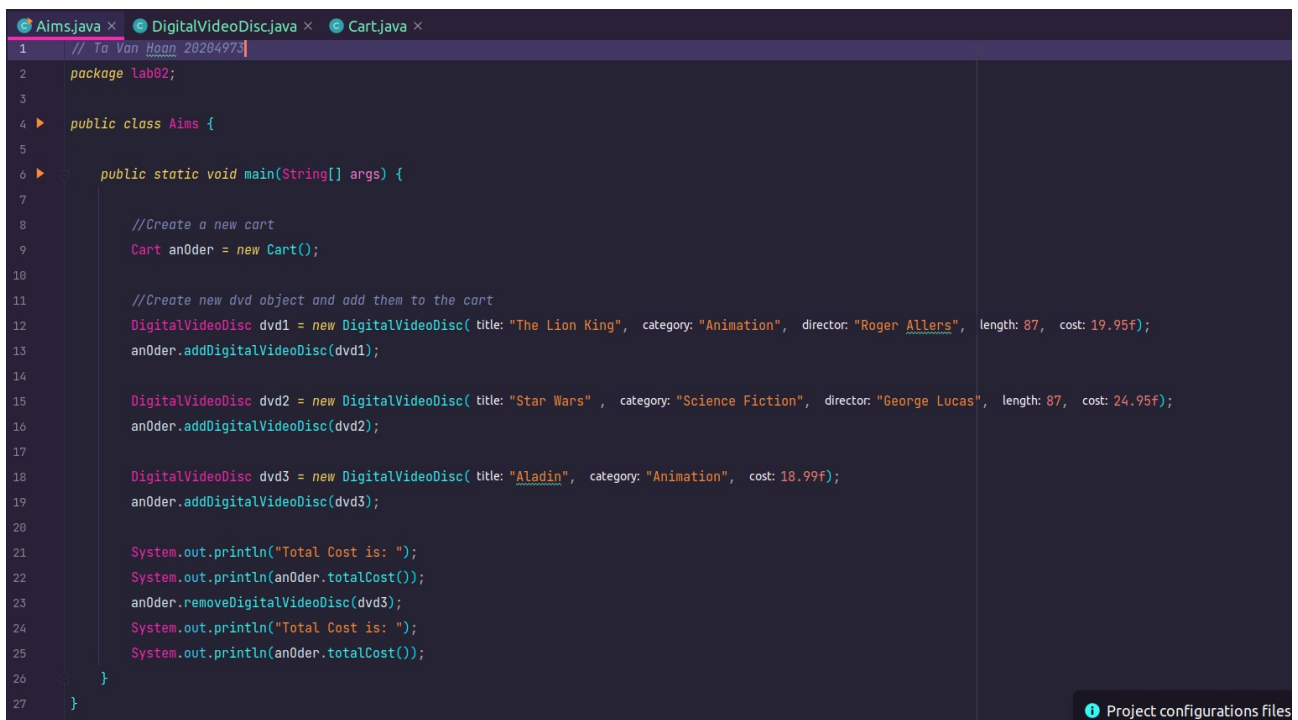


```
Aims x
/home/hoantv197/.jdk/openjdk-21/bin/java -javaagent:/snap/intellij-idea-educational/
The disc has been added
The disc has been added
The disc has been added
Total Cost is:
63.89

Process finished with exit code 0
```

Figure 10: Kết quả

## g.Removing items from the cart



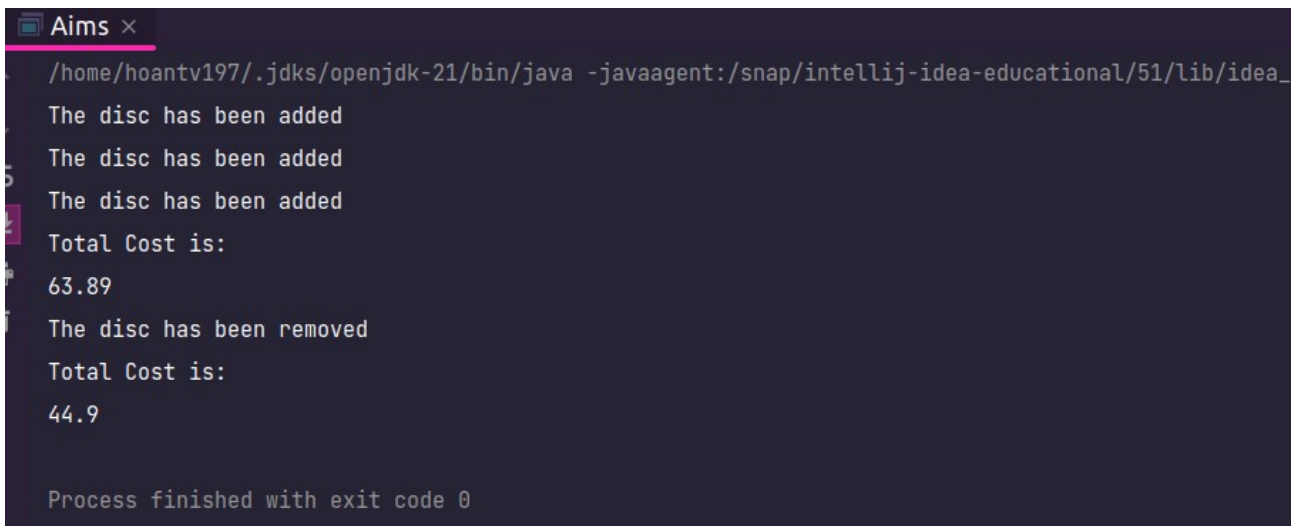
```

1 // Ta Van Hoan 20204973
2 package lab02;
3
4 public class Aims {
5
6     public static void main(String[] args) {
7
8         //Create a new cart
9         Cart anOrder = new Cart();
10
11         //Create new dvd object and add them to the cart
12         DigitalVideoDisc dvd1 = new DigitalVideoDisc( title: "The Lion King", category: "Animation", director: "Roger Allers", length: 87, cost: 19.95f);
13         anOrder.addDigitalVideoDisc(dvd1);
14
15         DigitalVideoDisc dvd2 = new DigitalVideoDisc( title: "Star Wars" , category: "Science Fiction", director: "George Lucas", length: 87, cost: 24.95f);
16         anOrder.addDigitalVideoDisc(dvd2);
17
18         DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin", category: "Animation", cost: 18.99f);
19         anOrder.addDigitalVideoDisc(dvd3);
20
21         System.out.println("Total Cost is: ");
22         System.out.println(anOrder.totalCost());
23         anOrder.removeDigitalVideoDisc(dvd3);
24         System.out.println("Total Cost is: ");
25         System.out.println(anOrder.totalCost());
26     }
27 }

```

Figure 11: Removing items from the cart

Result :



```

Aims x
/home/hoantv197/.jdk/openjdk-21/bin/java -javaagent:/snap/intellij-idea-educational/51/lib/idea_
The disc has been added
The disc has been added
The disc has been added
Total Cost is:
63.89
The disc has been removed
Total Cost is:
44.9

Process finished with exit code 0

```

Figure 12: Kết quả sau khi remove

## 6. Demo chương trình

Em đã quay video chạy chương trình và có thể mở bằng link dưới ạ  
<https://drive.google.com/file/d/1t2O5nPMiKlz9xINyrlEoCFPwBpvLJPnC/view?usp=sharing>