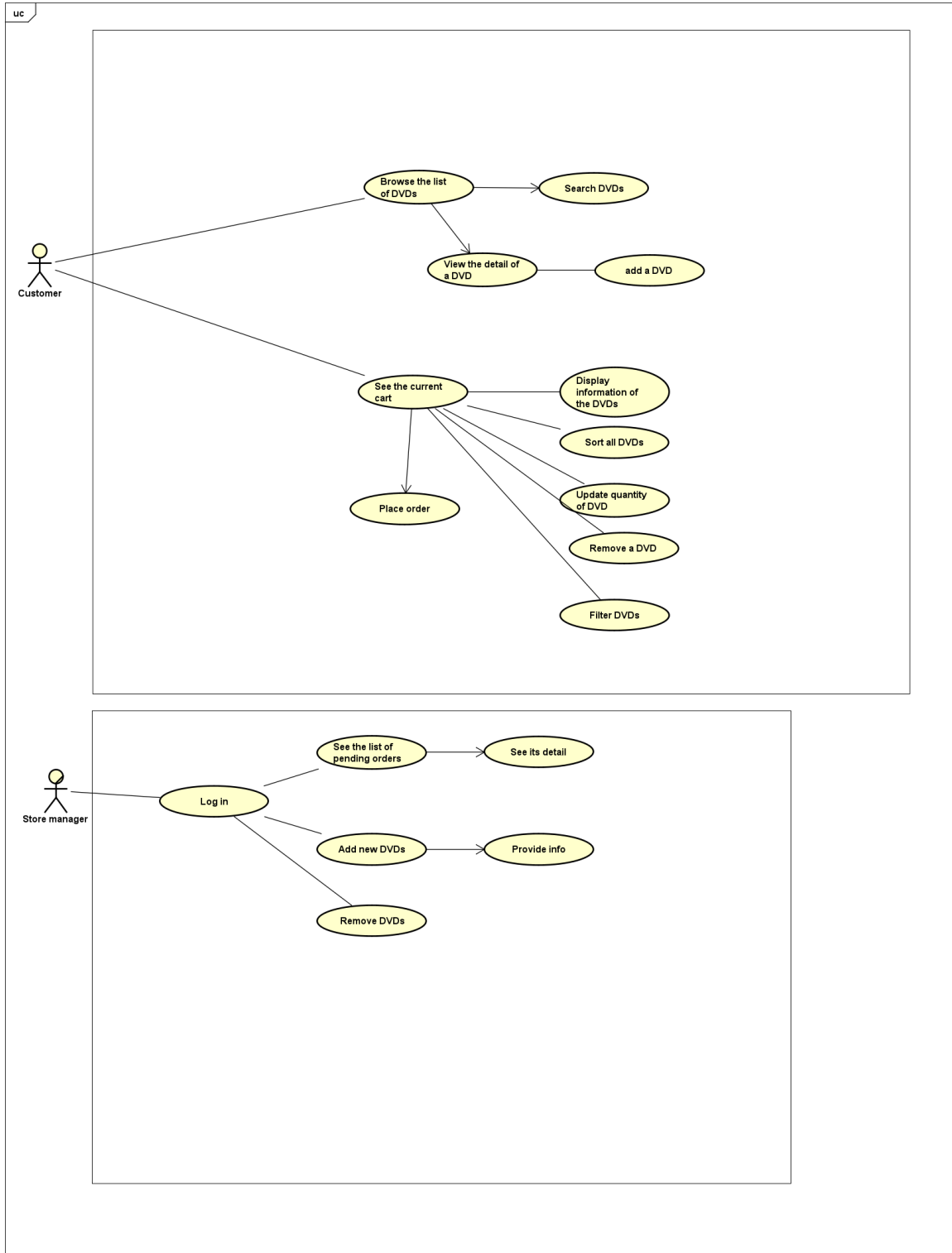


Thực hành lập trình hướng đối tượng

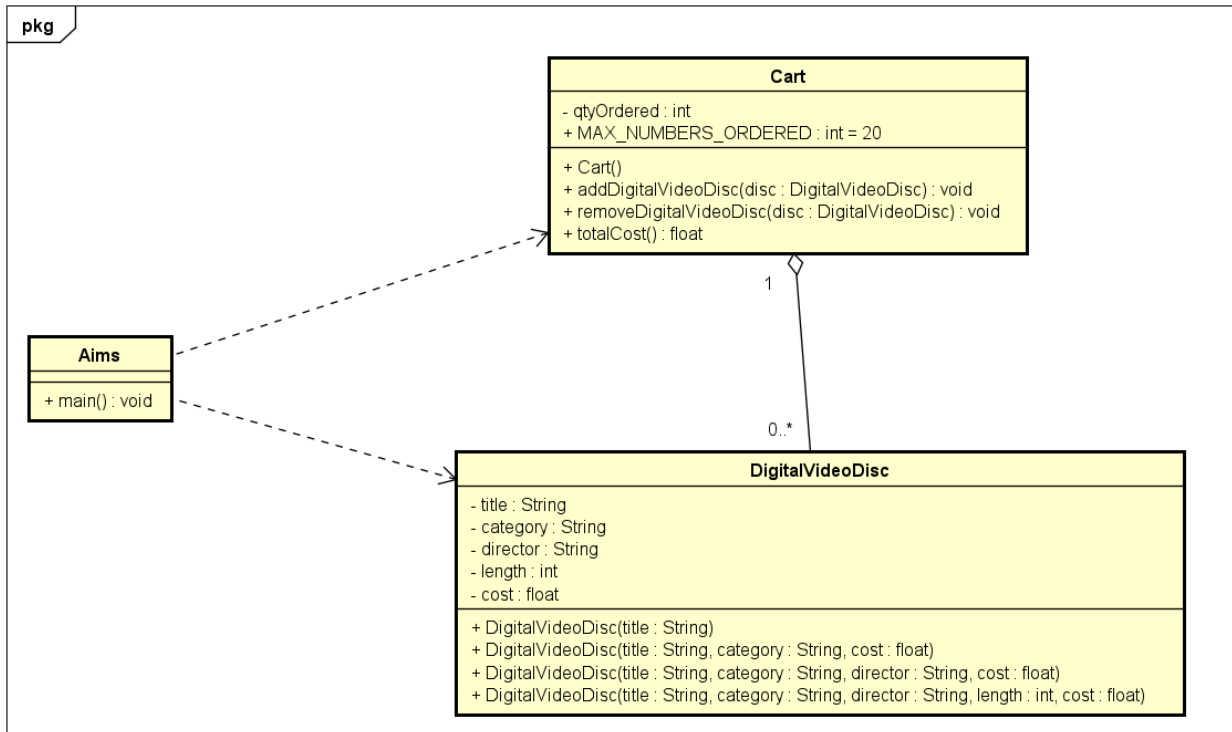
Báo cáo thực hành Lab02

Trần Nhật Minh – 20215284

1. Use case diagram

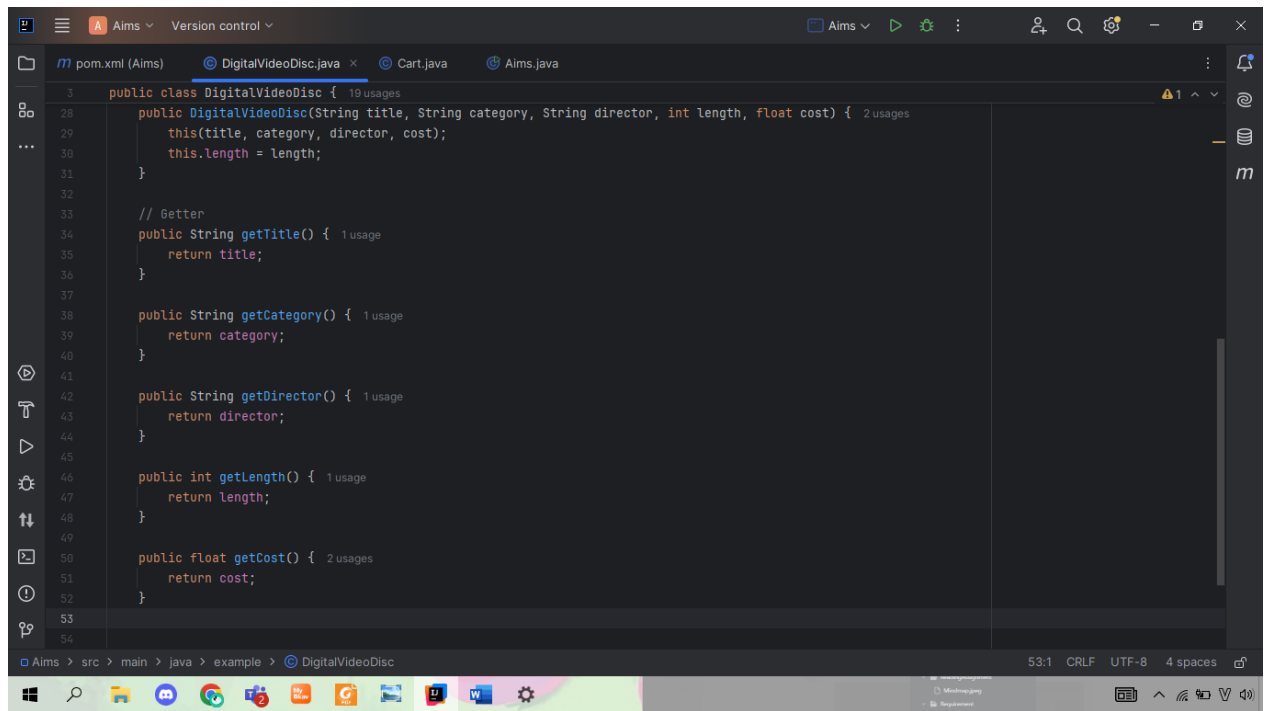


2. Class diagram



3. Lớp DigitalVideoDisc

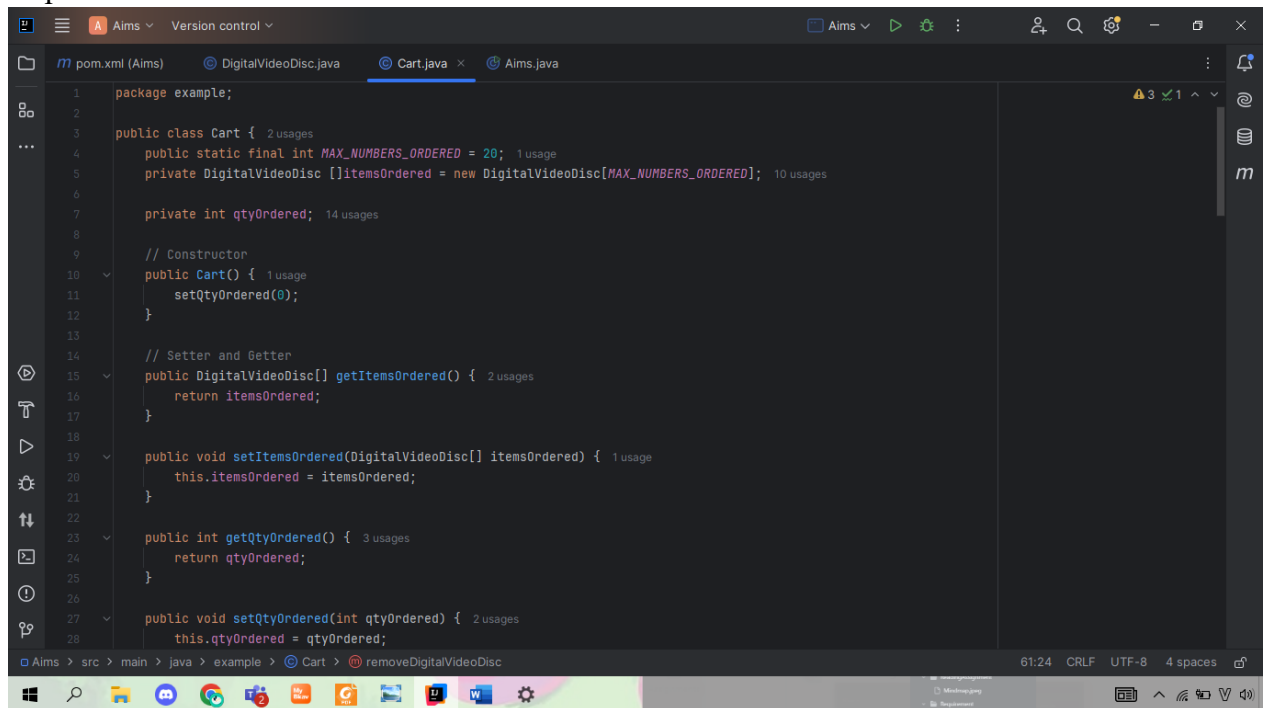
```
1 package example;
2
3 public class DigitalVideoDisc { 19 usages
4     // Attribute
5     private String title; 2 usages
6     private String category; 2 usages
7     private String director; 2 usages
8     private int length; 2 usages
9     private float cost; 2 usages
10
11     // Constructor
12     public DigitalVideoDisc(String title) { 1 usage
13         this.title = title;
14     }
15
16
17     public DigitalVideoDisc(String title, String category, float cost) { 2 usages
18         this(title);
19         this.category = category;
20         this.cost = cost;
21     }
22
23     public DigitalVideoDisc(String title, String category, String director, float cost) { 1 usage
24         this(title, category, cost);
25         this.director = director;
26     }
27
28     public DigitalVideoDisc(String title, String category, String director, int length, float cost) { 2 usages
```



The screenshot shows an IDE window with the file `DigitalVideoDisc.java` open. The code defines a `DigitalVideoDisc` class with a constructor and several getter methods. The IDE interface includes a sidebar with project files, a main editor area with line numbers, and a bottom status bar showing file encoding and line count.

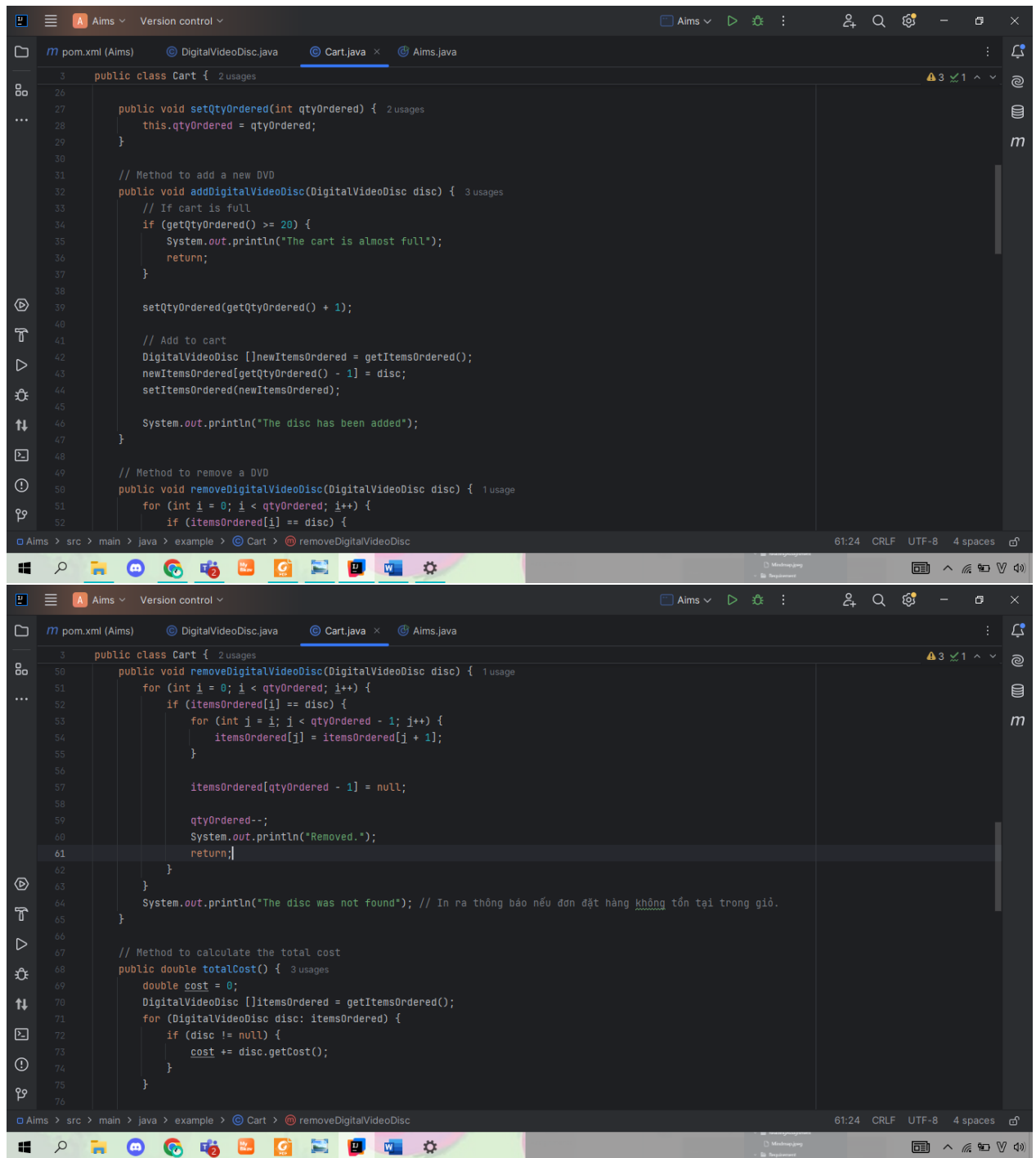
```
3 public class DigitalVideoDisc { 19 usages
28     public DigitalVideoDisc(String title, String category, String director, int length, float cost) { 2 usages
29         this(title, category, director, cost);
30         this.length = length;
31     }
32
33     // Getter
34     public String getTitle() { 1 usage
35         return title;
36     }
37
38     public String getCategory() { 1 usage
39         return category;
40     }
41
42     public String getDirector() { 1 usage
43         return director;
44     }
45
46     public int getLength() { 1 usage
47         return length;
48     }
49
50     public float getCost() { 2 usages
51         return cost;
52     }
53
54 }
```

4. Lớp Cart



The screenshot shows an IDE window with the file `Cart.java` open. The code defines a `Cart` class with a static constant, a private array of `DigitalVideoDisc` objects, and several methods for managing the cart. The IDE interface is similar to the previous screenshot, showing the same sidebar and status bar.

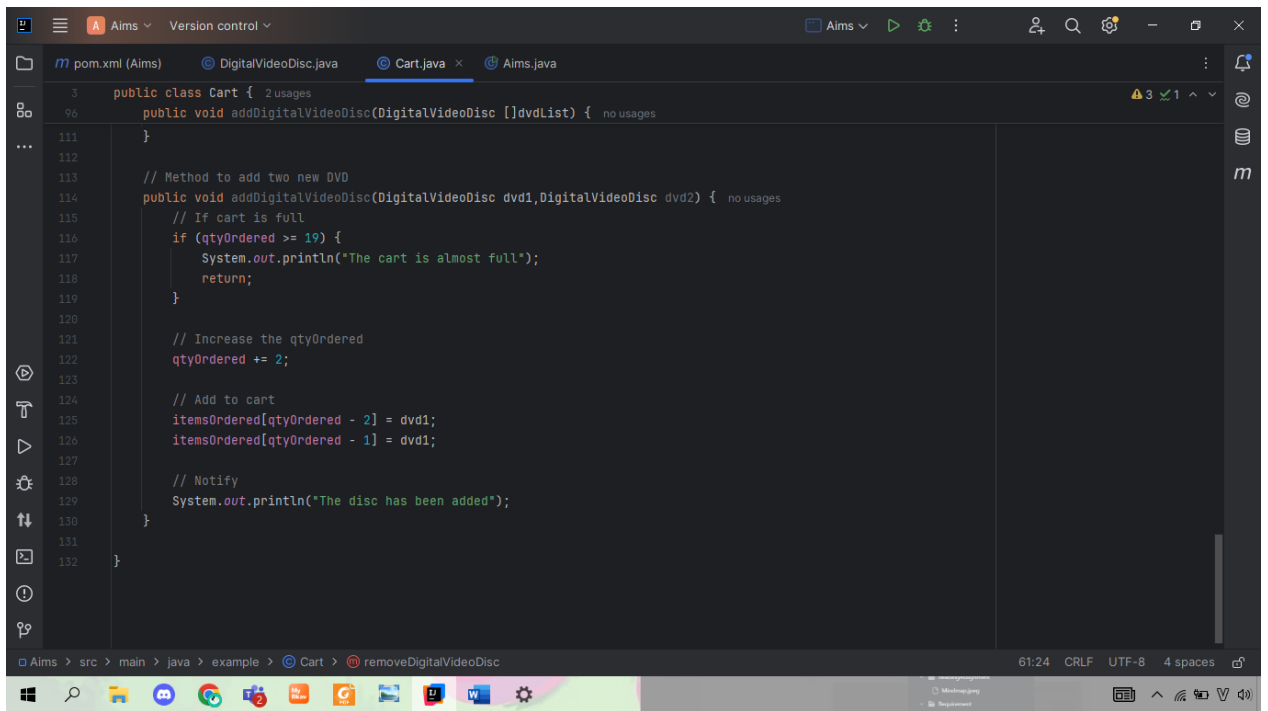
```
1 package example;
2
3 public class Cart { 2 usages
4     public static final int MAX_NUMBERS_ORDERED = 20; 1 usage
5     private DigitalVideoDisc []itemsOrdered = new DigitalVideoDisc[MAX_NUMBERS_ORDERED]; 10 usages
6
7     private int qtyOrdered; 14 usages
8
9     // Constructor
10    public Cart() { 1 usage
11        setQtyOrdered(0);
12    }
13
14    // Setter and Getter
15    public DigitalVideoDisc[] getItemsOrdered() { 2 usages
16        return itemsOrdered;
17    }
18
19    public void setItemsOrdered(DigitalVideoDisc[] itemsOrdered) { 1 usage
20        this.itemsOrdered = itemsOrdered;
21    }
22
23    public int getQtyOrdered() { 3 usages
24        return qtyOrdered;
25    }
26
27    public void setQtyOrdered(int qtyOrdered) { 2 usages
28        this.qtyOrdered = qtyOrdered;
29    }
29 }
```



```
3 public class Cart { 2 usages
68 public double totalCost() { 3 usages
73     cost += disc.getCost();
74 }
75 }
76
77 return Math.round(cost * 100.0) / 100.0;
78 }
79
80 //print Cart
81 public void printCart() { 1 usage
82     System.out.println("*****CART*****");
83     System.out.println("Ordered Items:\n");
84
85     for (int i = 0; i < qtyOrdered; i++) {
86         DigitalVideoDisc dvd = itemsOrdered[i];
87         System.out.println((i + 1) + ". DVD - " + dvd.getTitle() + " - " + dvd.getCategory() + " - "
88             + dvd.getDirector() + " - " + dvd.getLength() + ": " + dvd.getCost() + " $");
89     }
90
91     System.out.println("\nTotal cost: " + totalCost() + " $\n");
92     System.out.println("*****");
93 }
94
95 // method that has the same name but with different type of parameter.
96 @ public void addDigitalVideoDisc(DigitalVideoDisc []dvdList) { no usages
97     // If cart is full
98     if (qtyOrdered + dvdList.length > 20) {
99         System.out.println("The cart is almost full");
100         return;
101     }
102
103     // Add to cart
104     System.arraycopy(dvdList, srcPos: 0, itemsOrdered, qtyOrdered, dvdList.length);
105
106     // Increase the qtyOrdered
107     qtyOrdered += dvdList.length;
108
109     // Notify
110     System.out.println("The list has been added");
111 }
112
113 // Method to add two new DVD
114 public void addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) { no usages
115     // If cart is full
116     if (qtyOrdered >= 19) {
117         System.out.println("The cart is almost full");
118         return;
119     }
120 }
```

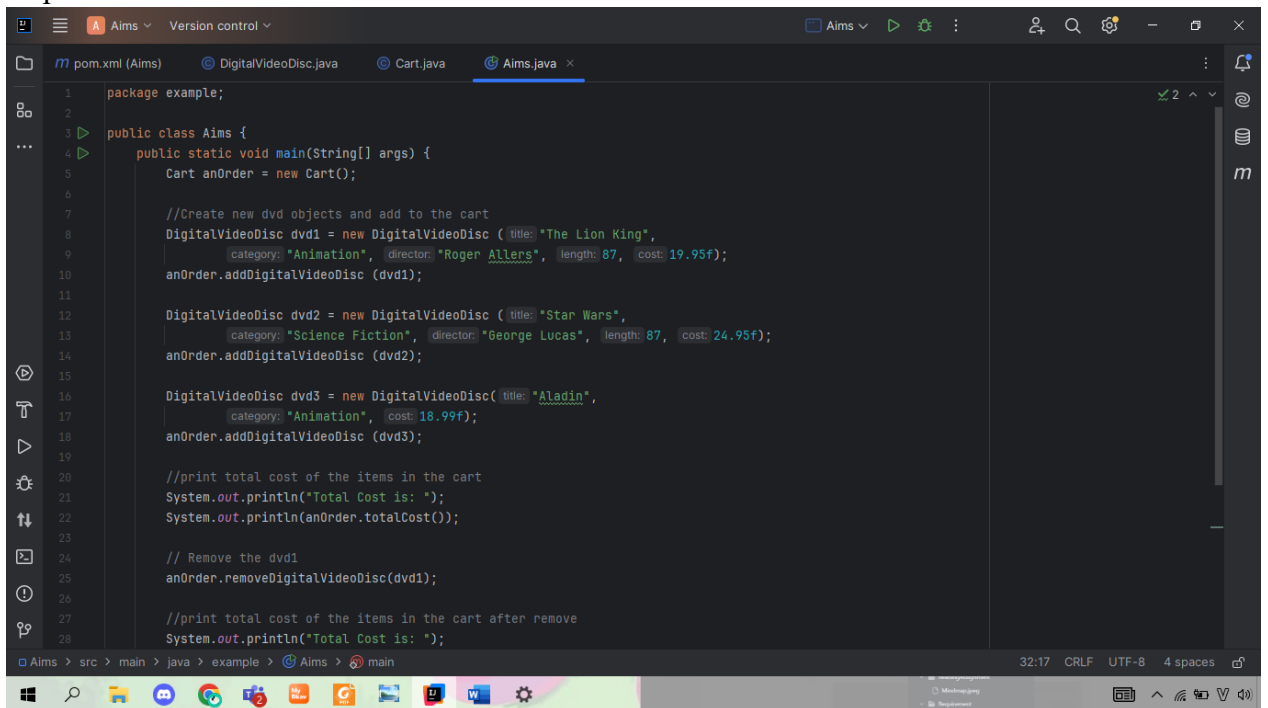
Các lớp overloading

```
94
95 // method that has the same name but with different type of parameter.
96 @ public void addDigitalVideoDisc(DigitalVideoDisc []dvdList) { no usages
97     // If cart is full
98     if (qtyOrdered + dvdList.length > 20) {
99         System.out.println("The cart is almost full");
100         return;
101     }
102
103     // Add to cart
104     System.arraycopy(dvdList, srcPos: 0, itemsOrdered, qtyOrdered, dvdList.length);
105
106     // Increase the qtyOrdered
107     qtyOrdered += dvdList.length;
108
109     // Notify
110     System.out.println("The list has been added");
111 }
112
113 // Method to add two new DVD
114 public void addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) { no usages
115     // If cart is full
116     if (qtyOrdered >= 19) {
117         System.out.println("The cart is almost full");
118         return;
119     }
120 }
```



```
3 public class Cart { 2 usages
96     public void addDigitalVideoDisc(DigitalVideoDisc []dvdList) { no usages
111     }
112
113     // Method to add two new DVD
114     public void addDigitalVideoDisc(DigitalVideoDisc dvd1,DigitalVideoDisc dvd2) { no usages
115         // If cart is full
116         if (qtyOrdered >= 19) {
117             System.out.println("The cart is almost full");
118             return;
119         }
120
121         // Increase the qtyOrdered
122         qtyOrdered += 2;
123
124         // Add to cart
125         itemsOrdered[qtyOrdered - 2] = dvd1;
126         itemsOrdered[qtyOrdered - 1] = dvd1;
127
128         // Notify
129         System.out.println("The disc has been added");
130     }
131 }
132 }
```

5. Lóp Aims



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

```
3 public class Aims {
4     public static void main(String[] args) {
5
6         DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin",
7             category: "Animation", cost: 18.99f);
8         anOrder.addDigitalVideoDisc (dvd3);
9
10        //print total cost of the items in the cart
11        System.out.println("Total Cost is: ");
12        System.out.println(anOrder.totalCost());
13
14        // Remove the dvd1
15        anOrder.removeDigitalVideoDisc(dvd1);
16
17        //print total cost of the items in the cart after remove
18        System.out.println("Total Cost is: ");
19        System.out.println(anOrder.totalCost());
20
21        // print items in cart
22        anOrder.printCart();
23    }
24 }
```

Kết quả:

```
Run
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:D:\Install\IntelliJ IDEA 2024.3.4\lib\idea_rt.jar=51443" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
The disc has been added
The disc has been added
The disc has been added
Total Cost is:
63.89
*****CART*****
Ordered Items:
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
2. DVD - Star Wars - Science Fiction - George Lucas - 87: 24.95 $
3. DVD - Aladin - Animation - null - 0: 18.99 $

Total cost: 63.89 $

*****
Removed.
Total Cost is:
43.94
*****CART*****
Ordered Items:
1. DVD - Star Wars - Science Fiction - George Lucas - 87: 24.95 $
2. DVD - Aladin - Animation - null - 0: 18.99 $
```