

Report Lab04

Name: Nguyễn Đức Mạnh

ID: 20235525

1. Source code:

1.1. AIMS class:

Eclipse - AIMSProject/src/hust.soict.dsai.aims - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

AIMS.java x Cart.java Disc.java DigitalVideoDisc.java Book.java

```
1 package hust.soict.dsai.aims;
2
3 import java.util.Scanner;
4
5 import hust.soict.dsai.aims.cart.Cart;
6 import hust.soict.dsai.aims.media.Media;
7 import hust.soict.dsai.aims.media.Playable;
8 import hust.soict.dsai.aims.store.Store;
9
10 public class Aims {
11     public static void showMenu() {
12         System.out.println("AIMS: ");
13         System.out.println("-----");
14         System.out.println("1. View store");
15         System.out.println("2. Update store");
16         System.out.println("3. See current cart");
17         System.out.println("0. Exit");
18         System.out.println("-----");
19         System.out.println("Please choose a number: 0-1-2-3");
20     }
21
22     public static void storeMenu() {
23         System.out.println("Options: ");
24         System.out.println("-----");
25         System.out.println("1. See a media's details");
26         System.out.println("2. Add a media to cart");
27         System.out.println("3. Play a media");
28         System.out.println("4. See current cart");
29         System.out.println("0. Back");
30         System.out.println("-----");
31         System.out.println("Please choose a number: 0-1-2-3-4");
32     }
33
34     public static void mediaDetailsMenu() {
35         System.out.println("Options: ");
36         System.out.println("-----");
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Aims.java x Cart.java Disc.java DigitalVideoDisc.java Book.java
34* public static void mediaDetailsMenu() {
35     System.out.println("Options: ");
36     System.out.println("-----");
37     System.out.println("1. Add to cart");
38     System.out.println("2. Play");
39     System.out.println("0. Back");
40     System.out.println("-----");
41     System.out.println("Please choose a number: 0-1-2");
42 }
43
44* public static void cartMenu() {
45     System.out.println("Options: ");
46     System.out.println("-----");
47     System.out.println("1. Filter medias in cart");
48     System.out.println("2. Sort medias in cart");
49     System.out.println("3. Remove media from cart");
50     System.out.println("4. Play a media");
51     System.out.println("5. Place order");
52     System.out.println("0. Back");
53     System.out.println("-----");
54     System.out.println("Please choose a number: 0-1-2-3-4-5");
55 }
56
57
58* public static void main(String[] args) {
59     Cart cart = new Cart();
60     Store store = new Store();
61     Scanner sc = new Scanner(System.in);
62     int a;
63
64     do {
65         showMenu();
66         a = sc.nextInt();
67         sc.nextLine();
68         switch (a) {
69             case 0:
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
64 do {
65     showMenu();
66     a = sc.nextInt();
67     sc.nextLine();
68     switch (a) {
69         case 0:
70             System.out.println("Exiting");
71             break;
72         case 1:
73             store.displayStore();
74             int storechoice;
75             do {
76                 storeMenu();
77                 storechoice = sc.nextInt();
78                 sc.nextLine();
79                 switch (storechoice) {
80                     case 0:
81                         break;
82                     case 1:
83                         System.out.print("Enter the title of the media: ");
84                         String title = sc.nextLine();
85                         Media media = store.findMedia(title);
86                         if (media != null) {
87                             System.out.println(media.toString());
88                             int mediachoice;
89                             do {
90                                 mediaDetailsMenu();
91                                 mediachoice = sc.nextInt();
92                                 sc.nextLine();
93                                 switch (mediachoice) {
94                                     case 1:
95                                         cart.addMedia(media);
96                                         break;
97                                     case 2:
98                                         if (media instanceof Playable) {
99                                             //PlayMedia(media);
100                                         }
101                                     default:
102                                         break;
103                                 }
104                             } while (mediachoice != 0);
105                         } else {
106                             System.out.println("Media not found");
107                         }
108                     } else {
109                         break;
110                     }
111                 case 2:
112                     System.out.println("Enter the title of the media: ");
113                     String title1 = sc.nextLine();
114                     Media media1 = store.findMedia(title1);
115                     if (media1 != null) {
116                         System.out.println(media1.toString());
117                         cart.addMedia(media1);
118                     } else {
119                         System.out.println("Media not found");
120                     }
121                     break;
122                 case 3:
123                     System.out.println("Enter the title of the media");
124                     ...
125                     break;
126                 ...
127             }
128         }
129     }
130 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
92 sc.nextLine();
93 switch (mediachoice) {
94     case 1:
95         cart.addMedia(media);
96         break;
97     case 2:
98         if (media instanceof Playable) {
99             ((Playable) media).play();
100         } else {
101             System.out.println("This media cannot be played");
102         }
103         break;
104     case 0:
105         break;
106     default:
107         System.out.println("Invalid choice.");
108         break;
109 }
110 } while (mediachoice != 0);
111 } else {
112     System.out.println("Media not found");
113 }
114 break;
115 case 2:
116     System.out.println("Enter the title of the media: ");
117     String title1 = sc.nextLine();
118     Media media1 = store.findMedia(title1);
119     if (media1 != null) {
120         System.out.println(media1.toString());
121         cart.addMedia(media1);
122     } else {
123         System.out.println("Media not found");
124     }
125     break;
126 case 3:
127     System.out.println("Enter the title of the media");
128     ...
129     break;
130 ...
131 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
125 break;
126 case 3:
127     System.out.println("Enter the title of the media");
128     String title2 = sc.nextLine();
129     Media media2 = store.findMedia(title2);
130     if (media2 != null) {
131         if (media2 instanceof Playable) {
132             ((Playable) media2).play();
133         } else {
134             System.out.println("This media cannot be played");
135         }
136     } else {
137         System.out.println("Media not found");
138     }
139     break;
140 case 4:
141     cart.displayCart();
142     int cartchoice;
143     do {
144         cartMenu();
145         cartchoice = sc.nextInt();
146         sc.nextLine();
147         switch (cartchoice) {
148             case 0:
149                 break;
150             case 1:
151                 cart.filterCart(sc);
152                 break;
153             case 2:
154                 cart.sortCart(sc);
155                 break;
156             case 3:
157                 cart.removeCart(sc);
158                 break;
159             case 4:
160                 cart.playMedia(sc);
161         }
162     }
163 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
...
        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");
    ...
```

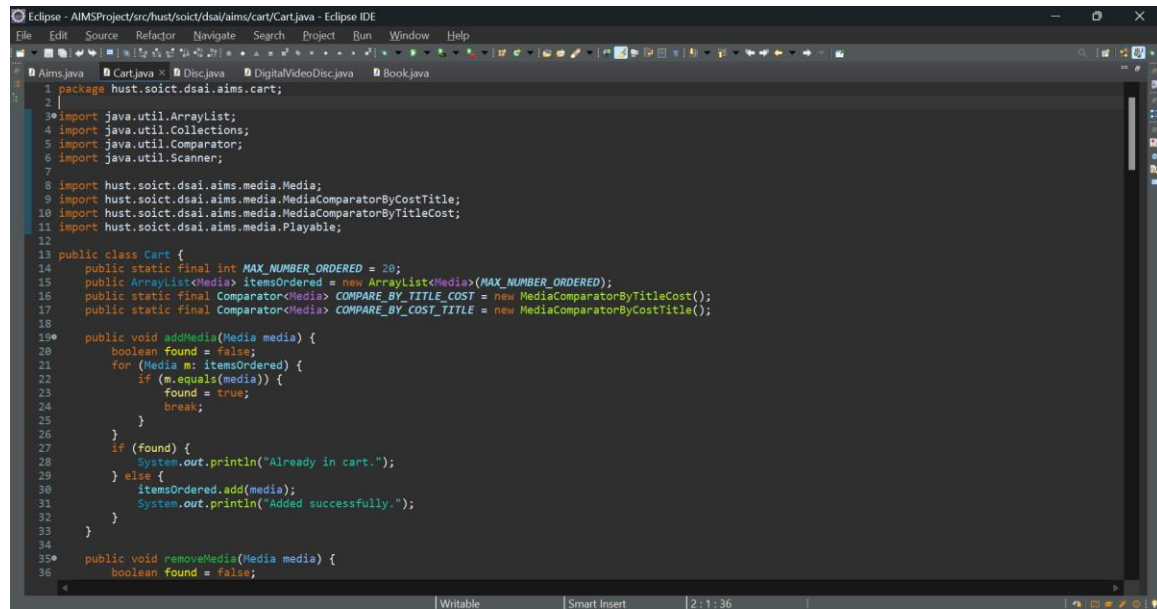
```
Eclipse - AIMSProject/src/hust/soict/dsa/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
...
        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;
        }
    }
}
```

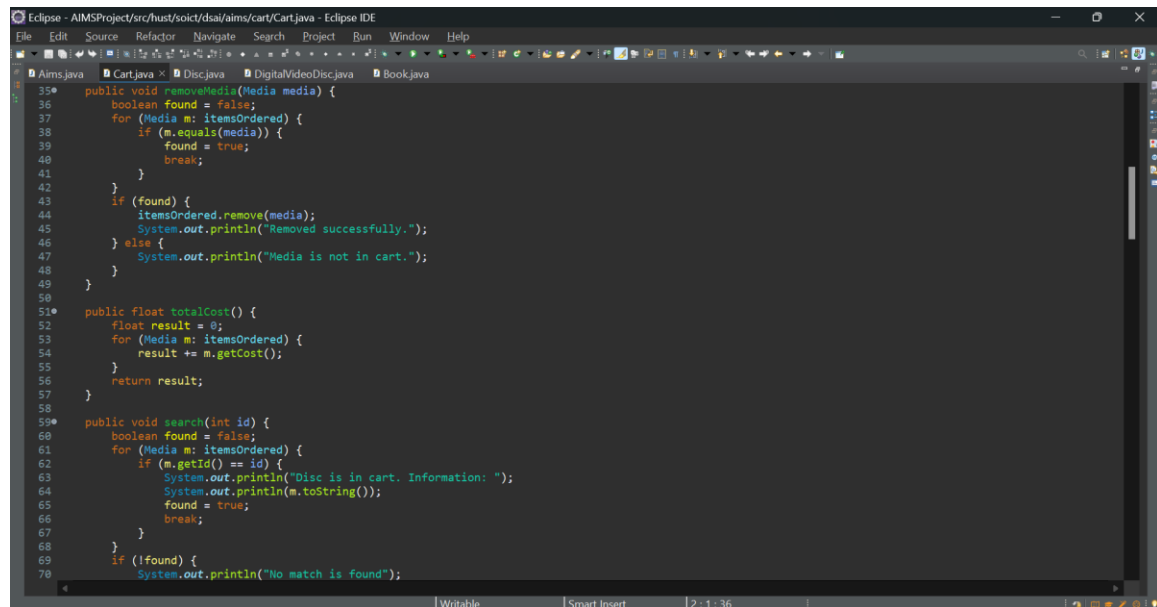
```
Eclipse - AIMSProject/src/hust/soict/dsa/aims/Aims.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Aims.java Cart.java Disc.java DigitalVideoDisc.java Book.java
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
...
        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:



```
1 package hust.soict.dsai.aims.cart;
2
3 import java.util.ArrayList;
4 import java.util.Collections;
5 import java.util.Comparator;
6 import java.util.Scanner;
7
8 import hust.soict.dsai.aims.media.Media;
9 import hust.soict.dsai.aims.media.MediaComparatorByCostTitle;
10 import hust.soict.dsai.aims.media.MediaComparatorByTitleCost;
11 import hust.soict.dsai.aims.media.Playable;
12
13 public class Cart {
14     public static final int MAX_NUMBER_ORDERED = 20;
15     public ArrayList<Media> itemsOrdered = new ArrayList<Media>(MAX_NUMBER_ORDERED);
16     public static final Comparator<Media> COMPARE_BY_TITLE_COST = new MediaComparatorByTitleCost();
17     public static final Comparator<Media> COMPARE_BY_COST_TITLE = new MediaComparatorByCostTitle();
18
19     public void addMedia(Media media) {
20         boolean found = false;
21         for (Media m: itemsOrdered) {
22             if (m.equals(media)) {
23                 found = true;
24                 break;
25             }
26         }
27         if (found) {
28             System.out.println("Already in cart.");
29         } else {
30             itemsOrdered.add(media);
31             System.out.println("Added successfully.");
32         }
33     }
34
35     public void removeMedia(Media media) {
36         boolean found = false;
```



```
37         for (Media m: itemsOrdered) {
38             if (m.equals(media)) {
39                 found = true;
40                 break;
41             }
42         }
43         if (found) {
44             itemsOrdered.remove(media);
45             System.out.println("Removed successfully.");
46         } else {
47             System.out.println("Media is not in cart.");
48         }
49     }
50
51     public float totalCost() {
52         float result = 0;
53         for (Media m: itemsOrdered) {
54             result += m.getCost();
55         }
56         return result;
57     }
58
59     public void search(int id) {
60         boolean found = false;
61         for (Media m: itemsOrdered) {
62             if (m.getId() == id) {
63                 System.out.println("Disc is in cart. Information: ");
64                 System.out.println(m.toString());
65                 found = true;
66                 break;
67             }
68         }
69         if (!found) {
70             System.out.println("No match is found");
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
66 break;
67 }
68 }
69 if (!found) {
70     System.out.println("No match is found");
71 }
72 }
73
74 public void search(String title) {
75     boolean found = false;
76     for (Media m: itemsOrdered) {
77         if (m.getTitle().equalsIgnoreCase(title)) {
78             found = true;
79             System.out.println("Disc is in cart. Information: ");
80             System.out.println(m.toString());
81             break;
82         }
83     }
84     if (!found) {
85         System.out.println("No match is found");
86     }
87 }
88
89 public void displayCart() {
90     if (itemsOrdered.size() == 0) {
91         System.out.println("No item in cart.");
92     } else {
93         for (Media m: itemsOrdered) {
94             System.out.println(m.toString());
95         }
96     }
97 }
98
99 public void sortByTitleCost() {
100     Collections.sort(itemsOrdered, COMPARE_BY_TITLE_COST);
101 }
102 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
98
99 public void sortByTitleCost() {
100     Collections.sort(itemsOrdered, COMPARE_BY_TITLE_COST);
101 }
102
103 public void sortByCostTitle() {
104     Collections.sort(itemsOrdered, COMPARE_BY_COST_TITLE);
105 }
106
107 public boolean Contains(Object obj) {
108     boolean found = false;
109     for (Media m: itemsOrdered) {
110         if (m.equals(obj)) {
111             found = true;
112             break;
113         }
114     }
115     return found;
116 }
117
118 public Media findMedia(String title) {
119     for (Media m: itemsOrdered) {
120         if (m.getTitle().equalsIgnoreCase(title)) {
121             return m;
122         }
123     }
124     return null;
125 }
126
127 public boolean isEmpty() {
128     if (itemsOrdered.size() == 0) {
129         return true;
130     }
131     return false;
132 }
133 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/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
126
127 public boolean isEmpty() {
128     if (itemsOrdered.size() == 0) {
129         return true;
130     }
131     return false;
132 }
133
134 public void emptyCart() {
135     for (Media m: itemsOrdered) {
136         itemsOrdered.remove(m);
137     }
138 }
139
140 public void filterCart(Scanner sc) {
141     System.out.println("1. Filter by id");
142     System.out.println("2. Filter by title");
143     System.out.println("3. Back");
144     int filter = sc.nextInt();
145     switch (filter) {
146         case 1:
147             int id = sc.nextInt();
148             Cart cart1 = new Cart();
149             for (Media m: itemsOrdered) {
150                 if (m.getId() == id) {
151                     cart1.addMedia(m);
152                 }
153             }
154
155             if (cart1.isEmpty()) {
156                 System.out.println("No media is in cart.");
157             } else {
158                 cart1.displayCart();
159             }
160             break;
161         case 2:
162             // ...
163     }
164 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

160 break;
161 case 2:
162     String title = sc.nextLine();
163     Cart cart2 = new Cart();
164     for (Media m : itemsOrdered) {
165         if (m.getTitle().equalsIgnoreCase(title)) {
166             cart2.addMedia(m);
167         }
168     }
169     if (cart2.isEmpty()) {
170         System.out.println("No media is in cart.");
171     } else {
172         cart2.displayCart();
173     }
174     break;
175 case 3:
176     break;
177 default:
178     System.out.println("Invalid choice.");
179     break;
180 }
181 }
182
183 public void sortCart(Scanner sc) {
184     System.out.println("1. Sort by title-cost.");
185     System.out.println("2. Sort by cost-title.");
186     System.out.println("3. Back.");
187     int sort = sc.nextInt();
188     switch (sort) {
189         case 1:
190             this.sortByTitleCost();
191             this.displayCart();
192             break;
193         case 2:
194             this.sortByCostTitle();
195             this.displayCart();
196     }
197 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsa/aims/cart/Cart.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

191 this.displayCart();
192 break;
193 case 2:
194     this.sortByCostTitle();
195     this.displayCart();
196     break;
197 case 3:
198     break;
199 default:
200     System.out.println("Invalid choice.");
201     break;
202 }
203 }
204
205 public void removeCart(Scanner sc) {
206     this.displayCart();
207     System.out.print("Enter cart title to be removed: ");
208     String title = sc.nextLine();
209     Media media = this.findMedia(title);
210     if (media != null) {
211         this.removeMedia(media);
212     } else {
213         System.out.println("Not in cart.");
214     }
215 }
216
217 public void playMedia(Scanner sc) {
218     System.out.print("Enter the title of the media: ");
219     String title = sc.nextLine();
220     Media media = this.findMedia(title);
221     if (media != null) {
222         if (media instanceof Playable) {
223             ((Playable) media).play();
224         } else {
225             System.out.println("This media cannot be played");
226         }
227     } else {
228         System.out.println("Media not found");
229     }
230 }
231 }
232 }
```

```
219 String title = sc.nextLine();
220 Media media = this.findMedia(title);
221 if (media != null) {
222     if (media instanceof Playable) {
223         ((Playable) media).play();
224     } else {
225         System.out.println("This media cannot be played");
226     }
227 } else {
228     System.out.println("Media not found");
229 }
230 }
231 }
232 }
```

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

1 package hust.soict.dsai.aims.store;
2 import java.util.ArrayList;
3 import java.util.Scanner;
4
5 import hust.soict.dsai.aims.media.Book;
6 import hust.soict.dsai.aims.media.CompactDisc;
7 import hust.soict.dsai.aims.media.DigitalVideoDisc;
8 import hust.soict.dsai.aims.media.Media;
9
10 public class Store {
11     private final int storeCapacity = 1000;
12     public ArrayList<Media> itemsInStore = new ArrayList<>(storeCapacity);
13
14     public void addMedia(Media media) {
15         boolean found = false;
16         for (Media m: itemsInStore) {
17             if (m.equals(media)) {
18                 found = true;
19                 break;
20             }
21         }
22         if (found) {
23             System.out.println("The media has been added.");
24         } else {
25             itemsInStore.add(media);
26             System.out.println("Added successfully.");
27         }
28     }
29
30     public void removeMedia(Media media) {
31         boolean found = false;
32         for (Media m: itemsInStore) {
33             if (m.equals(media)) {
34                 found = true;
35                 break;
36             }
37         }
38         if (found) {
39             itemsInStore.remove(media);
40             System.out.println("Removed successfully.");
41         } else {
42             System.out.println("The media is not in store.");
43         }
44     }
45
46     public void displayStore() {
47         if (itemsInStore.size() == 0) {
48             System.out.println("No item in store.");
49         } else {
50             for (Media m: itemsInStore) {
51                 System.out.println(m.toString());
52             }
53         }
54     }
55
56     public void searchMedia(String title) {
57         boolean found = false;
58         for (Media m: itemsInStore) {
59             if (m.getTitle().equalsIgnoreCase(title)) {
60                 found = true;
61                 System.out.println("Disc in store. Disc's information: ");
62                 System.out.println(m.toString());
63                 break;
64             }
65         }
66         if (!found) {
67             System.out.println("Disc is not found.");
68         }
69     }
70
71     public Media findMedia(String title) {
72         for (Media m: itemsInStore) {
73             if (m.getTitle().equalsIgnoreCase(title)) {
74                 return m;
75             }
76         }
77         return null;
78     }
79
80     public void addMedia(Scanner sc) {
81         System.out.println("Choose type of media: ");
82         System.out.println("1. Book");
83         System.out.println("2. DVD");
84         System.out.println("3. CD");
85         int choice = sc.nextInt();
86         sc.nextLine();
87         if (choice == 1) {
88             System.out.println("Enter id: ");
89             int id = sc.nextInt();
90             sc.nextLine();
91             System.out.println("Enter title: ");
92             String title = sc.nextLine();
93             System.out.println("Enter category: ");
94             String category = sc.nextLine();
95             System.out.println("Enter cost: ");
96             float cost = sc.nextFloat();
97             sc.nextLine();
98             Book book = new Book(id, title, category, cost, null);
99             this.addMedia(book);
100         } else if (choice == 2) {
101             System.out.println("Enter title: ");
102             String title = sc.nextLine();
103             System.out.println("Enter category: ");
104             String category = sc.nextLine();
105             System.out.println("Enter cost: ");
106             float cost = sc.nextFloat();
107             sc.nextLine();
108             CompactDisc cd = new CompactDisc(id, title, category, cost, null);
109             this.addMedia(cd);
110         } else if (choice == 3) {
111             System.out.println("Enter title: ");
112             String title = sc.nextLine();
113             System.out.println("Enter category: ");
114             String category = sc.nextLine();
115             System.out.println("Enter cost: ");
116             float cost = sc.nextFloat();
117             sc.nextLine();
118             DigitalVideoDisc dvd = new DigitalVideoDisc(id, title, category, cost, null);
119             this.addMedia(dvd);
120         }
121     }
122 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsai/aims/store/Store.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

34         found = true;
35         break;
36     }
37
38     if (found) {
39         itemsInStore.remove(media);
40         System.out.println("Removed successfully.");
41     } else {
42         System.out.println("The media is not in store.");
43     }
44 }
45
46 public void displayStore() {
47     if (itemsInStore.size() == 0) {
48         System.out.println("No item in store.");
49     } else {
50         for (Media m: itemsInStore) {
51             System.out.println(m.toString());
52         }
53     }
54 }
55
56 public void searchMedia(String title) {
57     boolean found = false;
58     for (Media m: itemsInStore) {
59         if (m.getTitle().equalsIgnoreCase(title)) {
60             found = true;
61             System.out.println("Disc in store. Disc's information: ");
62             System.out.println(m.toString());
63             break;
64         }
65     }
66     if (!found) {
67         System.out.println("Disc is not found.");
68     }
69 }
```

```
Eclipse - AIMSProject/src/hust/soict/dsai/aims/store/Store.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

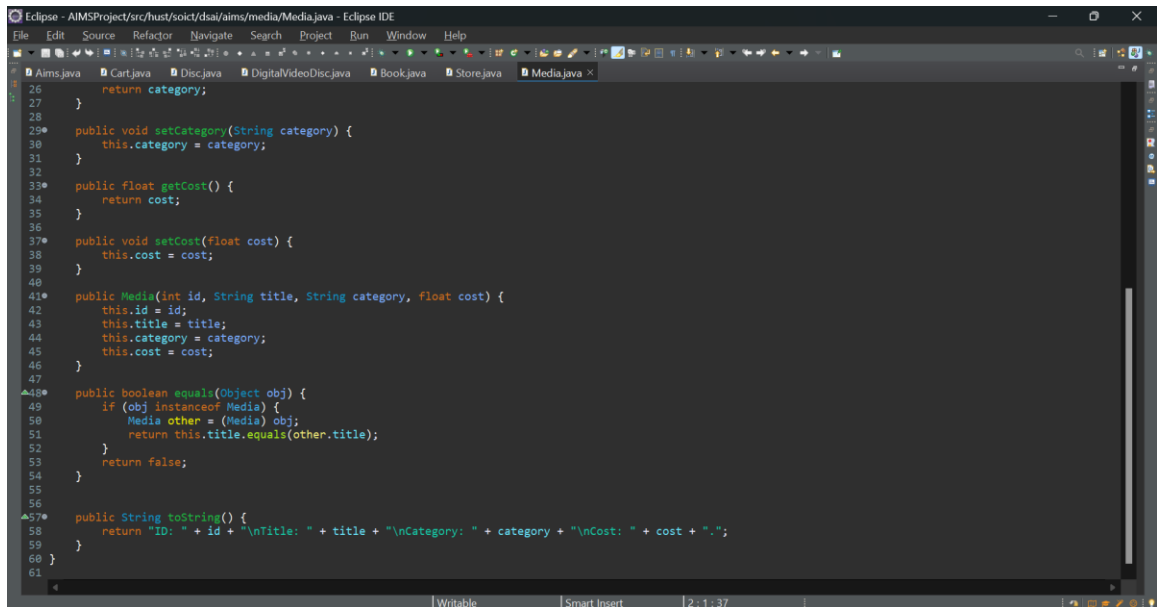
67         System.out.println("Disc is not found.");
68     }
69 }
70
71 public Media findMedia(String title) {
72     for (Media m: itemsInStore) {
73         if (m.getTitle().equalsIgnoreCase(title)) {
74             return m;
75         }
76     }
77     return null;
78 }
79
80 public void addMedia(Scanner sc) {
81     System.out.println("Choose type of media: ");
82     System.out.println("1. Book");
83     System.out.println("2. DVD");
84     System.out.println("3. CD");
85     int choice = sc.nextInt();
86     sc.nextLine();
87     if (choice == 1) {
88         System.out.println("Enter id: ");
89         int id = sc.nextInt();
90         sc.nextLine();
91         System.out.println("Enter title: ");
92         String title = sc.nextLine();
93         System.out.println("Enter category: ");
94         String category = sc.nextLine();
95         System.out.println("Enter cost: ");
96         float cost = sc.nextFloat();
97         sc.nextLine();
98         Book book = new Book(id, title, category, cost, null);
99         this.addMedia(book);
100     } else if (choice == 2) {
101         System.out.println("Enter title: ");
102         String title = sc.nextLine();
103         System.out.println("Enter category: ");
104         String category = sc.nextLine();
105         System.out.println("Enter cost: ");
106         float cost = sc.nextFloat();
107         sc.nextLine();
108         CompactDisc cd = new CompactDisc(id, title, category, cost, null);
109         this.addMedia(cd);
110     } else if (choice == 3) {
111         System.out.println("Enter title: ");
112         String title = sc.nextLine();
113         System.out.println("Enter category: ");
114         String category = sc.nextLine();
115         System.out.println("Enter cost: ");
116         float cost = sc.nextFloat();
117         sc.nextLine();
118         DigitalVideoDisc dvd = new DigitalVideoDisc(id, title, category, cost, null);
119         this.addMedia(dvd);
120     }
121 }
```

```
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 x
97 sc.nextLine();
98 Book book = new Book(id, title, category, cost, null);
99 this.addMedia(book);
100 } else if (choice == 2) {
101     System.out.println("Enter title");
102     String title = sc.nextLine();
103     System.out.println("Enter category");
104     String category = sc.nextLine();
105     System.out.println("Enter cost");
106     float cost = sc.nextFloat();
107     System.out.println("Enter length");
108     int length = sc.nextInt();
109     System.out.println("Enter director");
110     String director = sc.nextLine();
111     DigitalVideoDisc dvd = new DigitalVideoDisc(title, category, director, length, cost);
112     this.addMedia(dvd);
113 } else if (choice == 3) {
114     System.out.println("Enter title");
115     String title = sc.nextLine();
116     System.out.println("Enter category");
117     String category = sc.nextLine();
118     System.out.println("Enter cost");
119     float cost = sc.nextFloat();
120     System.out.println("Enter length");
121     int length = sc.nextInt();
122     System.out.println("Enter director");
123     String director = sc.nextLine();
124     System.out.println("Enter artist");
125     String artist = sc.nextLine();
126     CompactDisc cd = new CompactDisc(title, category, cost, length, director, artist);
127     this.addMedia(cd);
128 } else {
129     System.out.println("Invalid.");
130 }
131 }
132 }
```

```
129     System.out.println("Invalid.");
130 }
131 }
132 }
133 public void removeMedia(Scanner sc) {
134     System.out.println("Enter the title to be removed:");
135     String title = sc.nextLine();
136     boolean found = false;
137     for (Media m: itemsInStore) {
138         if (m.getTitle().equalsIgnoreCase(title)) {
139             found = true;
140             this.removeMedia(m);
141         }
142     }
143     if (found) {
144         System.out.println("Removed successfully.");
145     } else {
146         System.out.println("Media is not in store.");
147     }
148 }
149 }
```

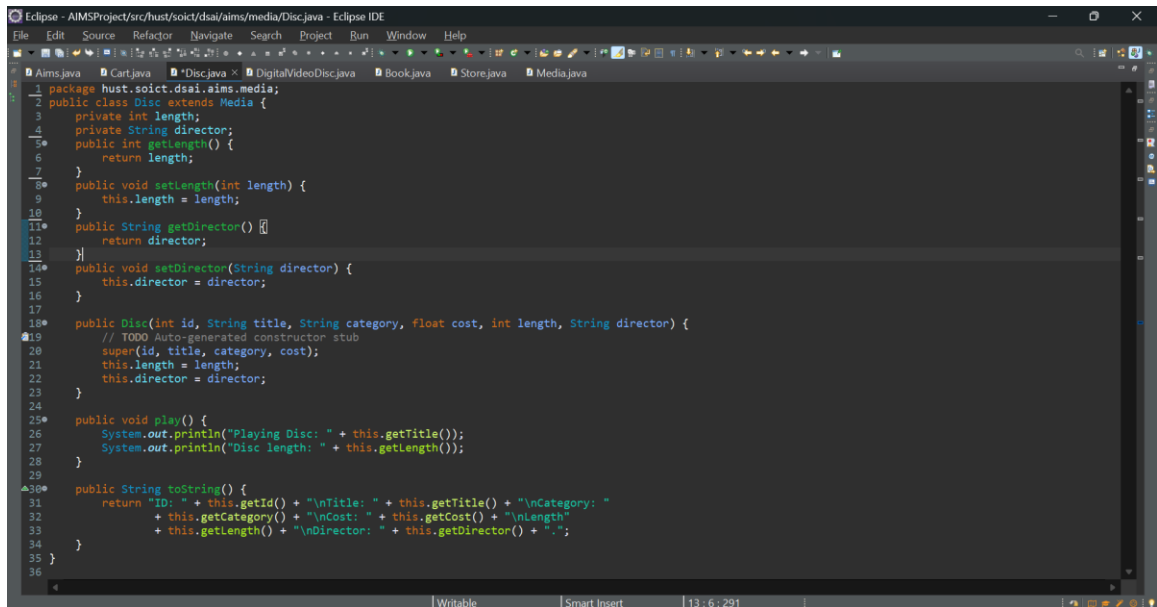
1.4. Media class:

```
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 x
1 package hust.soict.dsai.aims.media;
2
3 public abstract class Media {
4     private int id;
5     private String title;
6     private String category;
7     private float cost;
8
9     public int getId() {
10         return id;
11     }
12
13     public void setId(int id) {
14         this.id = id;
15     }
16
17     public String getTitle() {
18         return title;
19     }
20
21     public void setTitle(String title) {
22         this.title = title;
23     }
24
25     public String getCategory() {
26         return category;
27     }
28
29     public void setCategory(String category) {
30         this.category = category;
31     }
32
33     public float getCost() {
34         return cost;
35     }
36 }
```

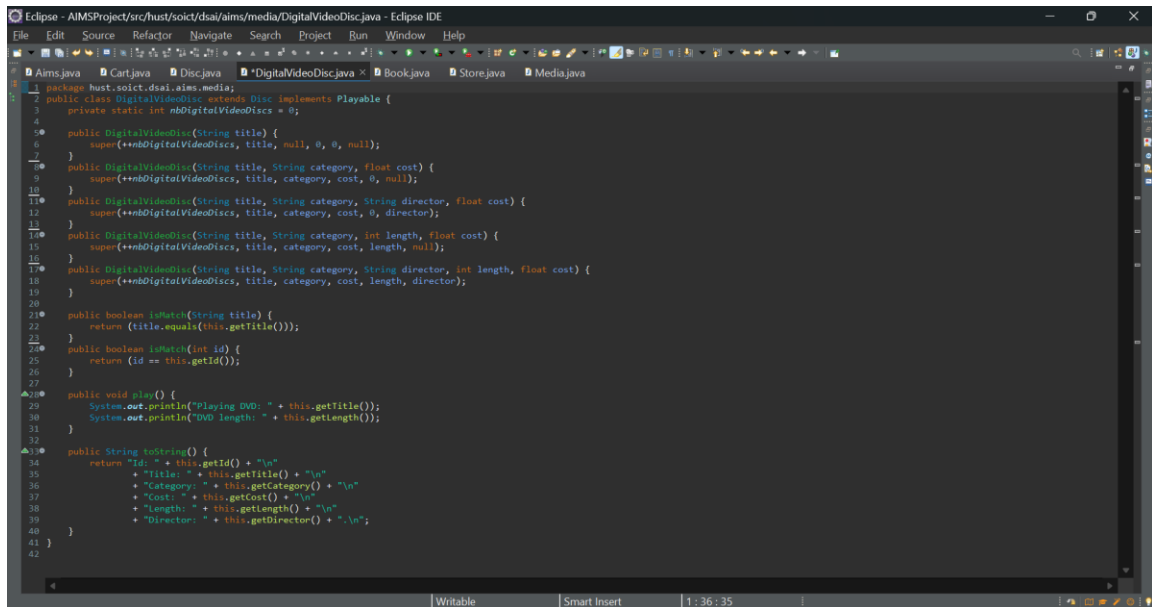
```
26     return category;
27 }
28
29 public void setCategory(String category) {
30     this.category = category;
31 }
32
33 public float getCost() {
34     return cost;
35 }
36
37 public void setCost(float cost) {
38     this.cost = cost;
39 }
40
41 public Media(int id, String title, String category, float cost) {
42     this.id = id;
43     this.title = title;
44     this.category = category;
45     this.cost = cost;
46 }
47
48 public boolean equals(Object obj) {
49     if (obj instanceof Media) {
50         Media other = (Media) obj;
51         return this.title.equals(other.title);
52     }
53     return false;
54 }
55
56
57 public String toString() {
58     return "ID: " + id + "\nTitle: " + title + "\nCategory: " + category + "\nCost: " + cost + ".";
59 }
60 }
61
```

1.5. Disc class:



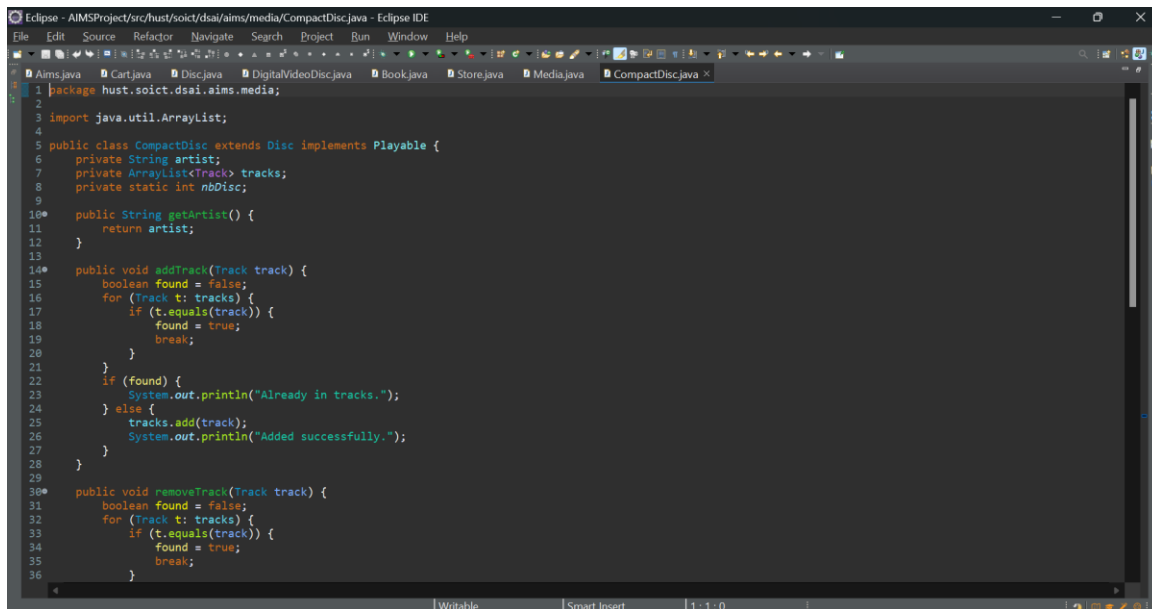
```
1 package hust.soict.dssl.aims.media;
2 public class Disc extends Media {
3     private int length;
4     private String director;
5     public int getLength() {
6         return length;
7     }
8     public void setLength(int length) {
9         this.length = length;
10    }
11    public String getDirector() {
12        return director;
13    }
14    public void setDirector(String director) {
15        this.director = director;
16    }
17
18    public Disc(int id, String title, String category, float cost, int length, String director) {
19        // TODO Auto-generated constructor stub
20        super(id, title, category, cost);
21        this.length = length;
22        this.director = director;
23    }
24
25    public void play() {
26        System.out.println("Playing Disc: " + this.getTitle());
27        System.out.println("Disc length: " + this.getLength());
28    }
29
30    public String toString() {
31        return "ID: " + this.getId() + "\nTitle: " + this.getTitle() + "\nCategory: "
32            + this.getCategory() + "\nCost: " + this.getCost() + "\nLength"
33            + this.getLength() + "\nDirector: " + this.getDirector() + ".";
34    }
35 }
36
```

1.6. DigitalVideoDisc class:



```
1 package hust.soict.dsa1.aims.media;
2 public class DigitalVideoDisc extends Disc implements Playable {
3     private static int nbDigitalVideoDiscs = 0;
4
5     public DigitalVideoDisc(String title) {
6         super(++nbDigitalVideoDiscs, title, null, 0, 0, null);
7     }
8     public DigitalVideoDisc(String title, String category, float cost) {
9         super(++nbDigitalVideoDiscs, title, category, cost, 0, null);
10    }
11    public DigitalVideoDisc(String title, String category, String director, float cost) {
12        super(++nbDigitalVideoDiscs, title, category, cost, 0, director);
13    }
14    public DigitalVideoDisc(String title, String category, int length, float cost) {
15        super(++nbDigitalVideoDiscs, title, category, cost, length, null);
16    }
17    public DigitalVideoDisc(String title, String category, String director, int length, float cost) {
18        super(++nbDigitalVideoDiscs, title, category, cost, length, director);
19    }
20
21    public boolean isMatch(String title) {
22        return (title.equals(this.getTitle()));
23    }
24    public boolean isMatch(int id) {
25        return (id == this.getId());
26    }
27
28    public void play() {
29        System.out.println("Playing DVD: " + this.getTitle());
30        System.out.println("DVD length: " + this.getLength());
31    }
32
33    public String toString() {
34        return "Id: " + this.getId() + "\n"
35            + "Title: " + this.getTitle() + "\n"
36            + "Category: " + this.getCategory() + "\n"
37            + "Cost: " + this.getCost() + "\n"
38            + "Length: " + this.getLength() + "\n"
39            + "Director: " + this.getDirector() + "\n";
40    }
41 }
42
```

1.7. CompactDisc class:



```
1 package hust.soict.dsa1.aims.media;
2
3 import java.util.ArrayList;
4
5 public class CompactDisc extends Disc implements Playable {
6     private String artist;
7     private ArrayList<Track> tracks;
8     private static int nbDisc;
9
10    public String getArtist() {
11        return artist;
12    }
13
14    public void addTrack(Track track) {
15        boolean found = false;
16        for (Track t: tracks) {
17            if (t.equals(track)) {
18                found = true;
19                break;
20            }
21        }
22        if (found) {
23            System.out.println("Already in tracks.");
24        } else {
25            tracks.add(track);
26            System.out.println("Added successfully.");
27        }
28    }
29
30    public void removeTrack(Track track) {
31        boolean found = false;
32        for (Track t: tracks) {
33            if (t.equals(track)) {
34                found = true;
35                break;
36            }
37        }
38    }
39 }

```

```
31 boolean found = false;
32 for (Track t: tracks) {
33     if (t.equals(track)) {
34         found = true;
35         break;
36     }
37 }
38 if (found) {
39     tracks.remove(track);
40     System.out.println("Removed successfully.");
41 } else {
42     System.out.println("Track is not in list of tracks");
43 }
44 }
45
46 public int getLength() {
47     int length = 0;
48     for (Track t: tracks) {
49         length += t.getLength();
50     }
51     return length;
52 }
53
54 public CompactDisc(String title, String category, float cost, int length, String director, String artist) {
55     // TODO Auto-generated constructor stub
56     super(title, category, cost, length, director);
57     this.artist = artist;
58 }
59
60 public void play() {
61     for (Track t: tracks) {
62         t.play();
63     }
64 }
65
66 public String toString() {
```

```
56     super(title, category, cost, length, director);
57     this.artist = artist;
58 }
59
60 public void play() {
61     for (Track t: tracks) {
62         t.play();
63     }
64 }
65
66 public String toString() {
67     StringBuilder sb = new StringBuilder();
68     sb.append("Id: ").append(nbDisc).append("\n")
69     .append("Title: ").append(this.getTitle()).append("\n")
70     .append("Category: ").append(this.getCategory()).append("\n")
71     .append("Cost: ").append(this.getCost()).append("\n")
72     .append("Length: ").append(this.getLength()).append("\n")
73     .append("Director: ").append(this.getDirector()).append("\n")
74     .append("Artist: ").append(this.artist).append("\n");
75     for (Track t: tracks) {
76         sb.append(t.toString());
77     }
78     return sb.toString();
79 }
80 }
81
```

1.8. Book class:

```
1 package hust.soict.dsai.aims.media;
2
3 import java.util.List;
4
5
6 public class Book extends Media {
7     private List<String> authors = new ArrayList<String>();
8
9     public Book(int id, String title, String category, float cost, List<String> authors) {
10         // TODO Auto-generated constructor stub
11         super(id, title, category, cost);
12         this.authors = authors;
13     }
14
15     public List<String> getAuthors() {
16         return authors;
17     }
18
19     public void setAuthors(List<String> authors) {
20         this.authors = authors;
21     }
22
23     public void addAuthor(String authorName) {
24         boolean found = false;
25         for (String name: authors) {
26             if (name.equals(authorName)) {
27                 found = true;
28                 break;
29             }
30         }
31         if (found) {
32             System.out.println("The author is already in list.");
33         } else {
34             authors.add(authorName);
35             System.out.println("Added successfully.");
36         }
37     }
38 }
```

```
1 package hust.soict.dsai.aims.media;
2
3 import java.util.ArrayList;
4
5 public class Book implements Playable {
6     private String title;
7     private int length;
8     private ArrayList<String> authors;
9
10    public String getTitle() {
11        return title;
12    }
13
14    public int getLength() {
15        return length;
16    }
17
18    public Book(String title, int length) {
19        this.title = title;
20        this.length = length;
21    }
22
23    public void play() {
24        System.out.println("Playing DVD: " + this.getTitle());
25        System.out.println("DVD length: " + this.getLength());
26    }
27
28    public boolean equals(Object obj) {
29        if (obj instanceof Book) {
30            Book other = (Book) obj;
31            return this.title.equals(other.title) && this.length == other.length;
32        }
33        return false;
34    }
35
36    public void addAuthor(String authorName) {
37        boolean found = false;
38        for (String name : authors) {
39            if (name.equals(authorName)) {
40                found = true;
41                break;
42            }
43        }
44        if (!found) {
45            authors.add(authorName);
46            System.out.println("Added successfully.");
47        }
48    }
49
50    public void removeAuthor(String authorName) {
51        boolean found = false;
52        for (String name : authors) {
53            if (name.equals(authorName)) {
54                found = true;
55                break;
56            }
57        }
58        if (!found) {
59            System.out.println("The author is not in the list.");
60        } else {
61            authors.remove(authorName);
62            System.out.println("Remove successfully.");
63        }
64    }
65}
```

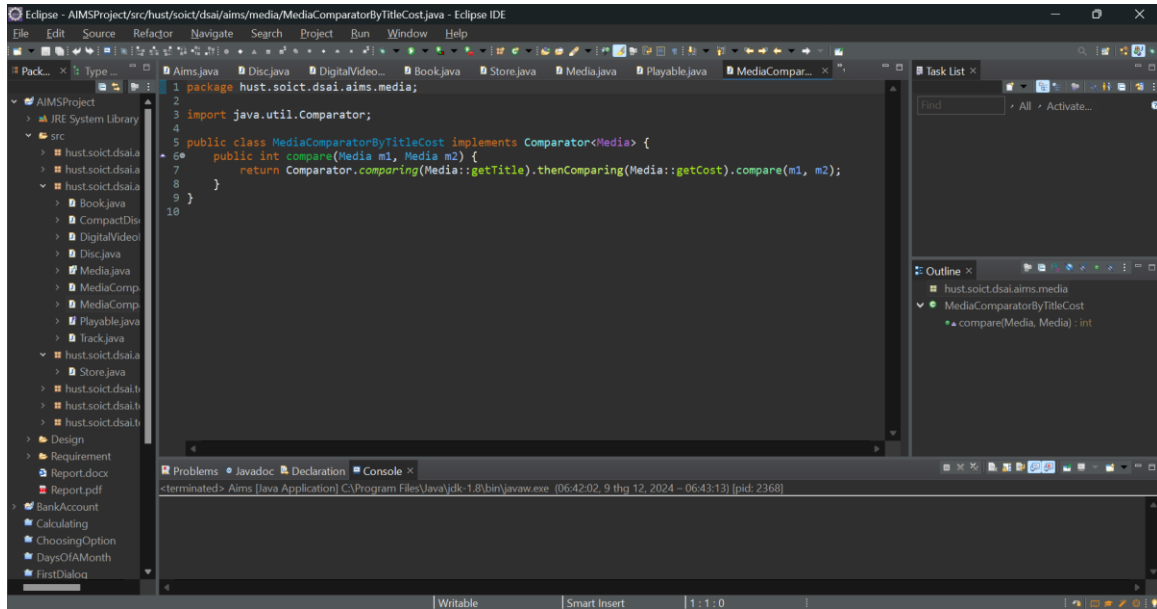
1.9. Track class:

```
1 package hust.soict.dsai.aims.media;
2
3 public class Track implements Playable {
4     private String title;
5     private int length;
6
7     public String getTitle() {
8         return title;
9     }
10
11    public int getLength() {
12        return length;
13    }
14
15    public Track(String title, int length) {
16        this.title = title;
17        this.length = length;
18    }
19
20    public void play() {
21        System.out.println("Playing DVD: " + this.getTitle());
22        System.out.println("DVD length: " + this.getLength());
23    }
24
25    public boolean equals(Object obj) {
26        if (obj instanceof Track) {
27            Track other = (Track) obj;
28            return this.title.equals(other.title) && this.length == other.length;
29        }
30        return false;
31    }
32}
```

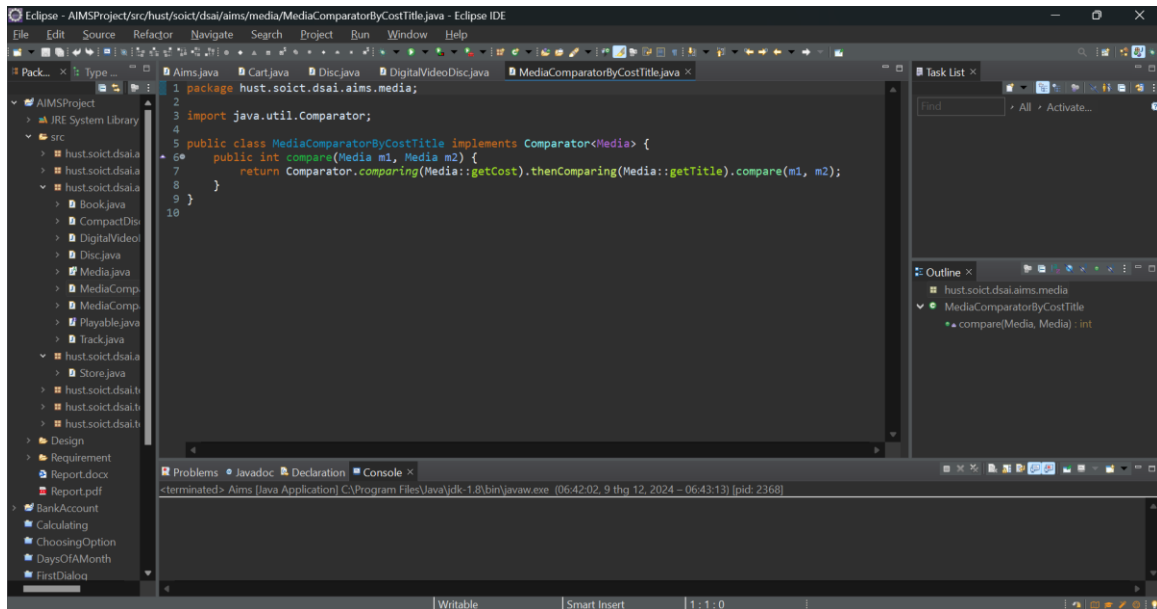
1.10. Interface Playable:

```
1 package hust.soict.dsai.aims.media;
2
3 public interface Playable {
4     public void play();
5 }
6
```

1.11. MediaComparatorByTitleCost:



1.12. MediaComparatorByCostTitle:



2. Diagram:

2.1. Class diagram:

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 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);  
}
```