Họ và tên: Tạ Tùng Dương MSSV: 20215267

BÁO CÁO THỰC HÀNH LAB03
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Table of Contents

Table of Figures

1. Working with method overloading

1.1. Overloading by differing types of parameter

```
1.      // Method to add list new DVDs
2.      public void addDigitalVideoDisc(DigitalVideoDisc []dvdList) {
3.          // If cart is full
4.          if (qtyOrdered + dvdList.length > 20) {
5.              System.out.println("The cart is almost full");
6.              return;
7.          }
8.
9.          // Add to cart
10.         System.arraycopy(dvdList, 0, itemsOrdered, qtyOrdered, dvdList.length);
11.
12.         // Increase the qtyOrdered
13.         qtyOrdered += dvdList.length;
14.
15.         // Notify
16.         System.out.println("The list has been added");
17.     }
```

1.2. Overloading by differing the number of parameters

```
1. // Method to add two new DVD
2.      public void addDigitalVideoDisc(DigitalVideoDisc dvd1,DigitalVideoDisc dvd2) {
3.          // If cart is full
4.          if (qtyOrdered >= 19) {
5.              System.out.println("The cart is almost full");
6.              return;
7.          }
8.
9.          // Increase the qtyOrdered
10.         qtyOrdered += 2;
11.
12.         // Add to cart
13.         itemsOrdered[qtyOrdered - 2] = dvd1;
14.         itemsOrdered[qtyOrdered - 1] = dvd1;
15.
16.         // Notify
17.         System.out.println("The disc has been added");
```

```
18.     }
```

2. Passing parameter

2.1. Script

```
 1. package LAB03.AimsProject;
 2.
 3. public class TestPassingParameter {
 4.     static class DVDWrapper {
 5.         DigitalVideoDisc disc;
 6.
 7.         DVDWrapper(DigitalVideoDisc disc) {
 8.             this.disc = disc;
 9.         }
10.     }
11.
12.     public static void main(String[] args) {
13.         // TODO Auto-generated method stub
14.         DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
15.         DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderella");
16.         DVDWrapper jungleDVDWrapper = new DVDWrapper(jungleDVD);
17.         DVDWrapper cinderellaDVDWrapper = new DVDWrapper(cinderellaDVD);
18.
19.         swap(jungleDVDWrapper, cinderellaDVDWrapper);
20.         System.out.println("jungle dvd title: " + jungleDVDWrapper.disc.getTitle());
21.         System.out.println("cinderella dvd title: " + cinderellaDVDWrapper.disc.getTitle());
22.
23.         changeTitle(jungleDVD, cinderellaDVD.getTitle());
24.         System.out.println("jungle dvd title: " + jungleDVD.getTitle());
25.     }
26.     public static void swap(DVDWrapper dvd1, DVDWrapper dvd2){
27.         DigitalVideoDisc tmp = dvd1.disc;
28.         dvd1.disc = dvd2.disc;
29.         dvd2.disc = tmp;
30.     }
31.
32.     public static void changeTitle(DigitalVideoDisc dvd, String title) {
33.         String oldTitle = dvd.getTitle();
34.         dvd.setTitle(title);
35.         dvd = new DigitalVideoDisc(oldTitle);
36.     }
37. }
```

2.2. Result

```
--- exec:3.1.0:exec (default-cli) @ OOP ---
jungle dvd title: Cinderella
cinderella dvd title: Jungle
jungle dvd title: Cinderella
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.895 s
Finished at: 2023-11-15T15:58:36+07:00
------------------------------------------------------------------------
```

Figure 1: Passing Paramenter Result

3. Classifier Member and Instance Member

3.1. Script

Classifier Member and Instance Member:

```
// Attribute
    private int id;
    private String title;
    private String category;
    private String director;
    private int length;
    private double cost;

    private static int nbDigitalVideoDiscs = 0;

    // Constructor
    public DigitalVideoDisc(String title) {
        this.title = title;
        nbDigitalVideoDiscs++;
        id = nbDigitalVideoDiscs;
    }

```

Aims class:

```
package LAB03.AimsProject;

public class Aims {
    public static void main(String[] args) {
        //Create a new cart
        Cart anOrder = new Cart();

        //Create new dvd objects and add them to the cart
        DigitalVideoDisc dvd1 = new DigitalVideoDisc ("The Lion King",
                "Animation", "Roger Allers", 87, 19.95f);
        anOrder.addDigitalVideoDisc (dvd1);

        DigitalVideoDisc dvd2 = new DigitalVideoDisc ("Star Wars",
                "Science Fiction", "George Lucas", 87, 24.95f);
        anOrder.addDigitalVideoDisc (dvd2);

        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladin",
                "Animation", 18.99f);
        anOrder.addDigitalVideoDisc (dvd3);

        //print number of dvd in cart
        System.out.println("Number of dvd is: ");
        System.out.println(DigitalVideoDisc.getNbDigitalVideoDiscs());

        //print id of dvd3
        System.out.println("Id of dvd3 is: ");
        System.out.println(dvd3.getId());
    }
}
```

3.2. Result

```
--- exec:3.1.0:exec (default-cli) @ OOP ---
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:
43.94
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  0.800 s
Finished at: 2023-11-15T16:01:57+07:00
------------------------------------------------------------------------
```

Figure 2: Classifier Member and Instance Member Result

4. Open the Cart class

4.1. Script

Method in Cart class to print the list of ordered items of a cart:

```
1.      // Method to print the list of ordered items of a cart,
2.      // the price of each item, and the total price
3.      public void printCart() {
4.          System.out.println("**********************CART**********************");
5.          System.out.println("Ordered Items:");
6.          for (DigitalVideoDisc dvd : itemsOrdered) {
7.              if (dvd != null)
8.                  dvd.printDVD();
9.          }
10.         System.out.println("Total cost: " + totalCost());
11.         System.out.println("************************************************");
12.     }
```

Method in Cart class to search for DVDs in the cart by ID:

```
1.      // Method to search for DVDs in the cart by ID and display the search results.
2.      public void searchByID(int id) {
3.          boolean found = false;
4.          for (DigitalVideoDisc dvd: itemsOrdered) {
5.              if (dvd != null && dvd.getId() == id) {
6.                  found = true;
7.                  dvd.printDVD();
8.              }
9.          }
10.         if (!found) {
11.             System.out.println("Not found!");
12.         }
13.     }
```

Method in Cart class to search for DVDs in the cart by title:

```
1.      // Method to search for DVDs in the cart by title and print the results.
2.      public void searchByTitle(String title) {
3.          boolean found = false;
4.          for (DigitalVideoDisc dvd: itemsOrdered) {
```

```
            if (dvd != null && dvd.isMatch(title)) {
                found = true;
                dvd.printDVD();
            }
        }
        if (!found) {
            System.out.println("Not found!");
        }
    }
```

Method in DigitalVideoDisc class to check title and print a dvd:

```
// Method to print a dvd
    public void printDVD() {
        System.out.println(id + ". DVD - "
                + title + " - "
                + category + " - "
                + director + " - "
                + length + ": "
                + cost + "$");
    }

    // Method to check if the corresponding disk is a match given the title.
    public boolean isMatch(String title) {
        return title.equals(this.title);
    }
```

Cart Test:

```
package LAB03.AimsProject;

public class CartTest {
    public static void main(String[] args) {
        //Create a new cart
        Cart cart = new Cart();
        //Create new dvd objects and add them to the cart
        DigitalVideoDisc dvd1 = new DigitalVideoDisc ("The Lion King",
                "Animation", "Roger Allers", 87, 19.95f);
        cart.addDigitalVideoDisc (dvd1);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc ("Star Wars",
                "Science Fiction", "George Lucas", 87, 24.95f);
        cart.addDigitalVideoDisc (dvd2);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc ("Aladin",
                "Animation", 18.99f);
        cart.addDigitalVideoDisc (dvd3);
        //Test the print method
        cart.printCart();
        //To-do: Test the search methods here
        cart.searchByID(1);
        cart.searchByID(4);
        cart.searchByTitle("Star Wars");
        cart.searchByTitle("Harry Potter");
    }
}
```

4.2. Result

```
--- exec:3.1.0:exec (default-cli) @ OOP ---
The disc has been added
The disc has been added
The disc has been added
**********CART**********
Ordered Items:
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.950000762939453$
2. DVD - Star Wars - Science Fiction - George Lucas - 87: 24.950000762939453$
3. DVD - Aladin - Animation - null - 0: 18.9899997711181164$
Total cost: 63.89
**************************************************
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.950000762939453$
Not found!
2. DVD - Star Wars - Science Fiction - George Lucas - 87: 24.950000762939453$
Not found!
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.967 s
Finished at: 2023-11-15T16:09:55+07:00
------------------------------------------------------------------------
```

Figure 3: Open The Cart Result

5. Implement the Store class

5.1. Script

Store class:

```java
package LAB03.AimsProject;

import java.util.ArrayList;

public class Store {
    // Attribute
    private ArrayList<DigitalVideoDisc> itemsInStore = new ArrayList<>();

    // Constructor
    public Store() {
    }

    // Method to add a dvd
    public void addDVD(DigitalVideoDisc disc) {
        // Add to store
        itemsInStore.add(disc);

        // Notify
        System.out.println("The disc has been added");
    }

    // Method to remove a dvd
    public void removeDVD(DigitalVideoDisc disc) {
        // Search for disc
        int indexOfRemoved = itemsInStore.indexOf(disc);

        // If not found
        if (indexOfRemoved == -1) {
```

```
29.             System.out.println("The disc is not found");
30.             return;
31.         }
32.
33.         // Remove
34.         itemsInStore.remove(indexOfRemoved);
35.
36.         // Notify
37.         System.out.println("The disc has been removed");
38.     }
39.
40.     // Getter and Setter
41.     public ArrayList<DigitalVideoDisc> getItemsInStore() {
42.         return itemsInStore;
43.     }
44.
45.     public void setItemsInStore(ArrayList<DigitalVideoDisc> itemsInStore) {
46.         this.itemsInStore = itemsInStore;
47.     }
48. }
```

Store Test:

```
 1. package LAB03.AimsProject;
 2.
 3. public class StoreTest {
 4.     public static void main(String[] args) {
 5.         //Create a new cart
 6.         Store store = new Store();
 7.
 8.         //Create new dvd objects and add them to the cart
 9.         DigitalVideoDisc dvd1 = new DigitalVideoDisc ("The Lion King",
10.                 "Animation", "Roger Allers", 87, 19.95f);
11.         store.addDVD(dvd1);
12.
13.         DigitalVideoDisc dvd2 = new DigitalVideoDisc ("Star Wars",
14.                 "Science Fiction", "George Lucas", 87, 24.95f);
15.         store.addDVD(dvd2);
16.
17.         DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladin",
18.                 "Animation", 18.99f);
19.         store.addDVD(dvd3);
20.
21.         // Remove success
22.         store.removeDVD(dvd1);
23.
24.         // Remove failed
25.         DigitalVideoDisc dvd4 = new DigitalVideoDisc("Harry Potter",
26.                 "Fiction", 18.99f);
27.         store.removeDVD(dvd4);
28.     }
29. }
```

5.2. Result

```
--- exec:3.1.0:exec (default-cli) @ OOP ---
The disc has been added
The disc has been added
The disc has been added
The disc has been removed
The disc is not found
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  0.838 s
Finished at: 2023-11-15T16:11:27+07:00
------------------------------------------------------------------------
```

Figure 4: Store Result

6. String, StringBuilder and StringBuffer

6.1. Script

```
1. package LAB03.OtherProjects.Garbage;
2.
3. import java.util.*;
4.
5. public class ConcatenationInLoops {
6.     public static void main(String[] args) {
7.         Random r = new Random(123);
8.         long start = System.currentTimeMillis();
9.         String s = "";
10.        for (int i = 0; i < 65536; i++)
11.            s += r.nextInt(2);
12.        System.out.println(System.currentTimeMillis() - start); // This prints roughly 4500.
13.        r = new Random (123);
14.        start = System.currentTimeMillis();
15.        StringBuilder sb = new StringBuilder();
16.        for (int i = 0; i < 65536; i++)
17.            sb.append(r.nextInt(2));
18.        s = sb.toString();
19.        System.out.println(System.currentTimeMillis() - start); // This prints 5.
20.    }
21. }
```

6.2. Result

```
--- exec:3.1.0:exec (default-cli) @ OOP ---
522
1
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  2.660 s
Finished at: 2023-11-15T16:13:25+07:00
------------------------------------------------------------------------
```

Figure 5: ConcatenationInLoops Result

7. Answer the Question

Question: Is JAVA a Pass by Value or a Pass by Reference programming language?

=> JAVA is a Pass by Value programming language.