BÁO CÁO THỰC HÀNH LAB 3
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Contents

Table of Figure

1. Branch your repository

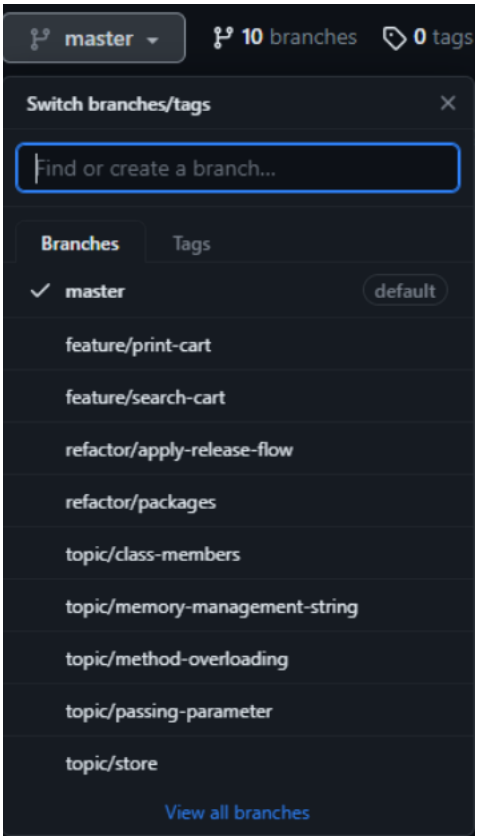

Figure 1 Branch

2. Working with method overloading

2.1 Overloading by differing types of parameters

```
1 usage  new *
public void addDigitalVideoDisc(DigitalVideoDisc[] dvdList) {
    for(DigitalVideoDisc disc : dvdList) {
        if(qtyOrdered < MAX_NUMBERS_ORDERED) {
            itemsOrdered[qtyOrdered] = disc;
            qtyOrdered++;
            System.out.println("The disc " + disc.getTitle() + " has been added");
        } else {
            System.out.println("The cart is almost full. Cannot add " + disc.getTitle());
            break;
        }
    }
}
```

Figure 2 Overloading add a list

2.2 Overloading

```
public void addDigitalVideoDisc(DigitalVideoDisc... dvds) {
  for (DigitalVideoDisc disc : dvds) {
      if (qtyOrdered < MAX_NUMBERS_ORDERED) {
          itemsOrdered[qtyOrdered] = disc;
          qtyOrdered++;
          System.out.println("The disc " + disc.getTitle() + " has been added");
      } else {
          System.out.println("The cart is almost full. Cannot add " + disc.getTitle());
          break;
      }
  }
```

Figure 3 Overloading with number of arguments

```
//Create new dvd objects and add them to the cart
DigitalVideoDisc dvd1 = new DigitalVideoDisc( title: "The Lion King", category: "Animation", director: "Roger Allers", length: 87,
DigitalVideoDisc dvd2 = new DigitalVideoDisc( title: "Star Wars", category: "Science Fiction", director: "George Lucas", length: 87
DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin", category: "Animation", cost: 18.99F);
anOrder.addDigitalVideoDisc(dvd1); // Single DVD

anOrder.addDigitalVideoDisc(new DigitalVideoDisc[]{dvd2, dvd3}); // Array of DVDs
// anOrder.addDigitalVideoDisc(dvd1, dvd2, dvd3); // Varargs (arbitrary number of DVDs)
```

Figure 4 Code test

```
Pham Duc Dung 20215265
The disc has been added
The disc Star Wars has been added
The disc Aladin has been added
Total Cost is: 63.89
The disc has been removed.
Total Cost is:
38.94
```

Figure 5 Ket qua

3. Passing parameter

```
public class TestPassingParameter {
    public static void main(String[] args) {
        System.out.println("Pham Duc Dung 20215265");
        DigitalVideoDisc jungleDVD = new DigitalVideoDisc( title: "Jungle");
        DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc( title: "Cinderella");
        swap(jungleDVD, cinderellaDVD);
        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
        System.out.println("cinderellaDVD title: "+ cinderellaDVD.getTitle());
        changeTitle(jungleDVD,cinderellaDVD.getTitle());
        System.out.println("jungle dvd title: " + jungleDVD.getTitle());

    }
    1 usage
    public static void swap(Object o1, Object o2) { // thay vì thay đổi references, chúng ta swap the contents(title..) của 2 đối tư
        Object tmp = o1;
        o1 = o2;
        o2 = tmp;
    }
    1 usage
```

Figure 6 Class testPassingParameter

```
    1 usage
    public static void changeTitle( DigitalVideoDisc dvd, String title) {
        String oldTitle = dvd.getTitle();
        dvd.setTitle(title);
        dvd = new DigitalVideoDisc(oldTitle);
    }
}
```

Figure 7 ChangeTitle

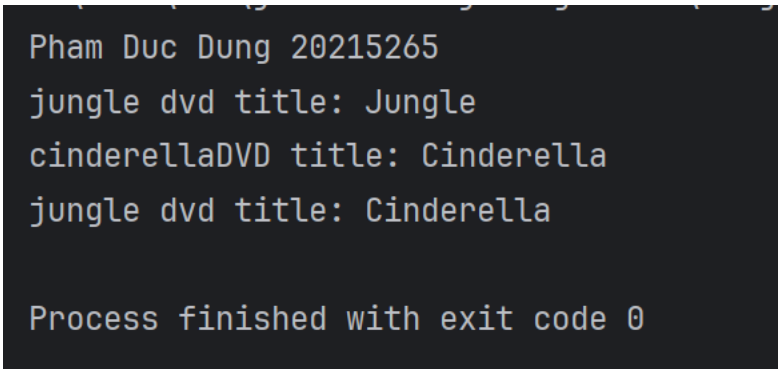

Figure 8 Ket qua

swap(jungleDVD, cinderellaDVD): After this call, the titles of the two DigitalVideoDisc objects remain the same because Java passes arguments by value. This means that swap receives copies of the references to jungleDVD and cinderellaDVD. Swapping these copies doesn't affect the original references in the main method.

changeTitle(jungleDVD, cinderellaDVD.getTitle()): The title of jungleDVD changes because the method changeTitle directly modifies the object that the reference jungleDVD points to. This is because jungleDVD is a reference to an object, and while the reference itself is passed by value, the object it points to can be modified within the method.

4. Use debug run

Làm lần lượt các bước :

- Setting, deleting & deactivate breakpoints
- Run in Debug mode
- tep Into, Step Over, Step Return, Resume
- Investigate value of variables
- Change value of variables

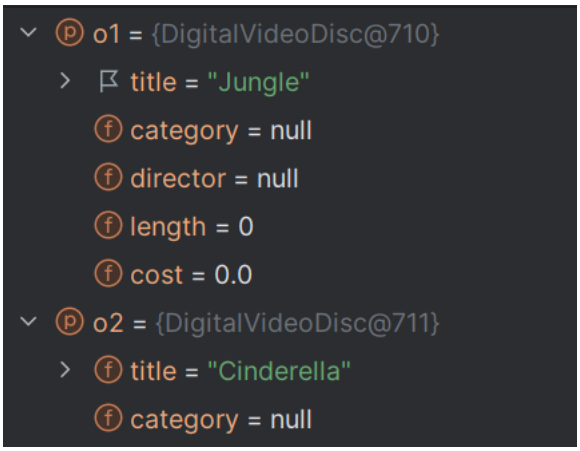

Figure 9 Gia tri cua cac thuoc tinh

```
jungle dvd title: Jungle PhamDucDung
cinderellaDVD title: Cinderella
jungle dvd title: Cinderella
```

Figure 10 Ket qua sau khi sua gia tri cua thuoc tinh

5. Classifier Member and Instance Member

```
7 usages
private float cost;
1 usage
private int id; // attribute for storing unique ID
2 usages
private static int nbDigitalVideoDiscs = 0; // class attribute to keep track the number of DVDs created
```

Figure 11: 2 attribute them vao

```
private DigitalVideoDisc() {
    nbDigitalVideoDiscs++; // Increment the count
    this.id = nbDigitalVideoDiscs; // Set id to the current count
}
new *
```

Figure 12 Use to keeptrach

6. Open the cart class

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

Figure 13 Method to string

```java
public void print() {
    System.out.println("**********CART**********");
    System.out.println("Ordered Items:");
    for (int i = 0; i < qtyOrdered; i++) {
        System.out.println(itemsOrdered[i].getTitle()) ;
    }
    System.out.println("Total cost: " + this.totalCost());
    System.out.println("**************************************************");
}
```

Figure 14 Method to print

```
**********CART**********
Ordered Items:
The Lion King
Star Wars
Movie 1
Movie 2
Movie 3
Movie A
Movie B
Total cost: 152.85
**************************************************
```

Figure 15 Ket qua

```java
public void search(String title) {
    int index = 0;
    for (int i = 0; i < qtyOrdered; i++) {
        if (itemsOrdered[i].getTitle() == title) {
            index++;
        }
    }
    if (index == 0 || qtyOrdered == 0) {
        System.out.println("Cannot find DVD!");
    } else {
        System.out.println("Find " + index + title + "DVD\n");
    }
}
```

Figure 16 Method to search by title

```java
public void search(int id) {
    int index = 0;
    for (int i = 0; i < qtyOrdered; i++) {
        if (itemsOrdered[i].getID() == id) {
            index++;
        }
    }
    if (index == 0 || qtyOrdered == 0) {
        System.out.println("Cannot find DVD!");
    } else {
        System.out.println("Find " + index + id + "DVD\n");
    }
}
```

Figure 17 Search by id

```
    public int getID() { return ID; }
    no usages  new *
    public void setID(int iD) { ID = iD; }
}
```

Figure 18 Getter va setter cua ID

```
**************************************************
Cannot find DVD!
Find 1The Lion KingDVD
```

Figure 19 Result for finding DVD

Aims.java | CartTest.java | Cart.java | DigitalVideoDisc.java

```
package soict;

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

        DigitalVideoDisc dvd1 = new DigitalVideoDisc( title: "The Lion king", category: "Animation", director: "Roger Allers", length: 87, cost: 19.95f);
        cart.addDigitalVideoDisc(dvd1);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc( title: "Star Wars", category: "Science Fiction", director: "George Lucas", length: 87, cost: 24.95f);
        cart.addDigitalVideoDisc(dvd2);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin", category: "Animatin", cost: 18.99f);
        cart.addDigitalVideoDisc(dvd3);
        cart.print();
    }
}
```

Figure 20 CartTest

7. Implement the **Store** class

```
package soict;

no usages  new *
public class Store {
    2 usages
    public static final int MAX_NUMBERS_ORDERED = 100;
    7 usages
    private DigitalVideoDisc itemsInStore[] = new DigitalVideoDisc[MAX_NUMBERS_ORDERED];
    7 usages
    private int qtyStore = 0;
```

Figure 21 Atribute store class

```java
public void addDigitalVideoDisc(DigitalVideoDisc disc) {
    if (qtyStore == MAX_NUMBERS_ORDERED) {
        System.out.println("The store is almost full");
        return;
    } else {
        itemsInStore[qtyStore] = disc;
        qtyStore++;
        System.out.println("The disc has been added");
    }
}
```

Figure 22 Method add

```java
public void removeDigitalVideoDisc(DigitalVideoDisc disc) {
    int index = 0;
    for (int i = 0; i < qtyStore; i++) {
        if (itemsInStore[i].equals(disc) ) {
            System.out.println("Remove " + itemsInStore[i].getTitle());
            System.arraycopy(itemsInStore, srcPos: i+1, itemsInStore, i, length: itemsInStore.length-i-1);
            i--;
            qtyStore --;
            index ++;
        } else if (i == qtyStore-1 && index == 0) {
            System.out.println("Not found!");
        }
    }
}
```

Figure 23 Method remove

```java
public float totalCost() {
    float total = 0;
    for (int i = 0; i < qtyStore; i++) {
        total += itemsInStore[i].getCost();
    }
    return total;
}
```

Figure 24 Calculate totalCost

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

        DigitalVideoDisc dvd1 = new DigitalVideoDisc( title: "The Lion King",  category: "Animation",  director: "Roger Allers",  length: 87,  cost: 19.95f);
        Store.addDigitalVideoDisc(dvd1);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc( title: "Star Wars",  category: "Science Fiction",  director: "George Lucas",  length: 87,  cost: 24.95f);
        Store.addDigitalVideoDisc(dvd2);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin",  category: "Animation",  cost: 18.99f);
        Store.addDigitalVideoDisc(dvd3);
        System.out.println("Total Cost is: ");
        System.out.println(Store.totalCost());

        //Check remove
        Store.removeDigitalVideoDisc(dvd3);
        System.out.println("Total Cost is: ");
        System.out.println(Store.totalCost());
    }
}
```

Figure 25 StoreTest

8. Re-organize your projects

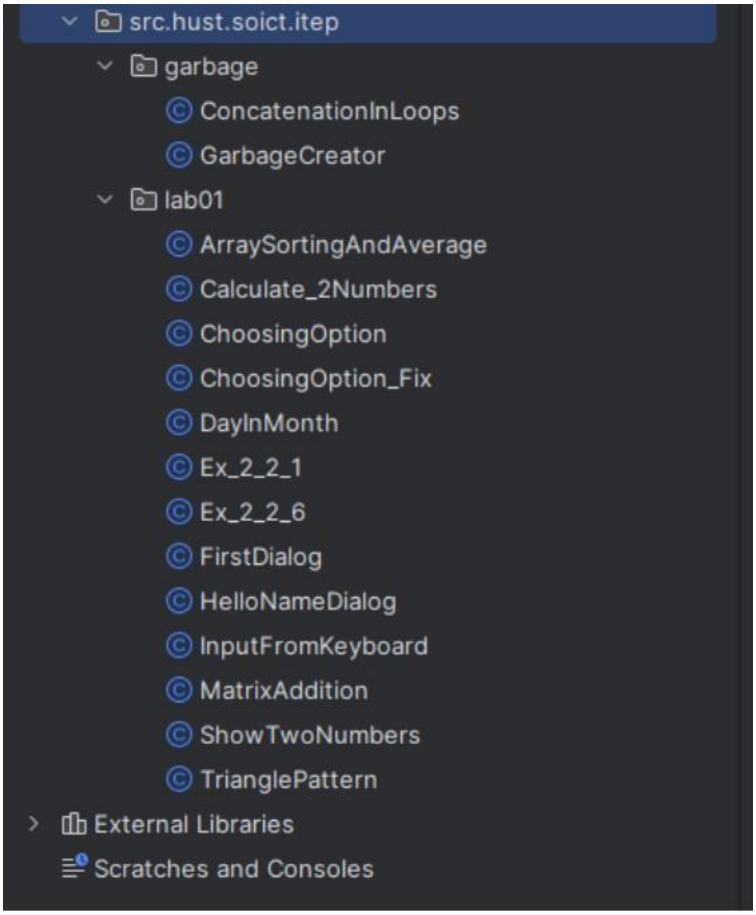

Figure 26 Otherprojects

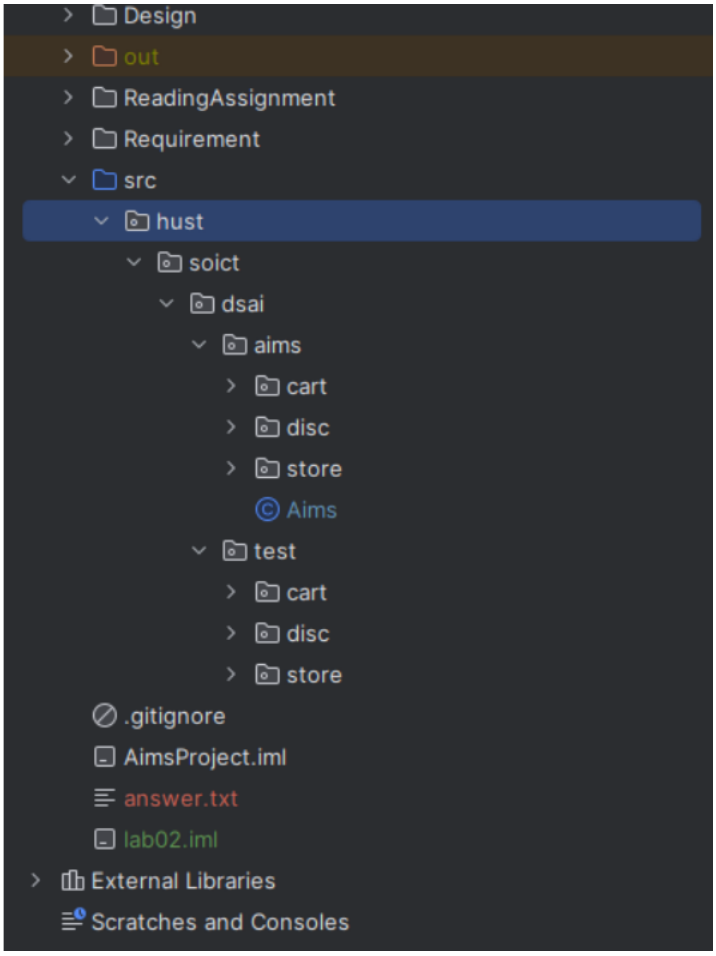

Figure 27 Aims Project

9. Create the Cart class to work with DigitalVideoDisc

```java
public class ConcatenationInLoops {
    public static void main(String[] args) {
        Random r = new Random( seed: 123);
        long start = System.currentTimeMillis();
        String s = "";
        for (int i = 0; i < 65536; i++)
            s += r.nextInt( bound: 2);
        System.out.println(System.currentTimeMillis() - start);


        r = new Random ( seed: 123);
        start = System.currentTimeMillis();
        StringBuilder sb = new StringBuilder();
        for (int i = 0; i < 65536; i++)
            sb.append(r.nextInt( bound: 2));
        s = sb.toString();
        System.out.println(System.currentTimeMillis() - start);
    }
}
```

Figure 28 Code ConcatenationInLoops

```
C:\Java\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.2\lib\idea_rt.jar=63275:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.2\bin" -Dfi
308
0

Process finished with exit code 0
```

Figure 29 Ket Qua