

Báo cáo bài thực hành số 3

Họ và tên: Đặng Kim Ngân

MSSV: 20225751

Mã lớp: 744520

1. New written code

1.1. Working with method overloading

Tại class Cart.java: nạp chồng phương thức addDigitalVideoDisc

```
public void addDigitalVideoDisc(DigitalVideoDisc[] dvdList) {
    if (itemsOrdered.size() + dvdList.length > MAX_NUMBERS_ORDERED) {
        System.out.println("The cart is almost full!");
    } else {
        for (DigitalVideoDisc dvd : dvdList) {
            itemsOrdered.add(dvd);
            System.out.println(dvd.getTitle() + " has been added!");
        }
    }
}

public void addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {
    addDigitalVideoDisc(new DigitalVideoDisc[]{dvd1, dvd2});
}
```

1.2. Passing parameter

- Thêm setter cho title của lớp DigitalVideoDisc.

```
public void setTitle(String title) {
    this.title = title;
}
```

- Tạo lớp TestPassingParameter như dưới đây:

```

public class TestPassingParameter {
    public static void main(String[] args) {
        DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
        DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderella");

        swap(jungleDVD, cinderellaDVD);
        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
        System.out.println("cinderella dvd title: " + cinderellaDVD.getTitle());

        changeTitle(jungleDVD, cinderellaDVD.getTitle());
        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
    }

    public static void swap(Object o1, Object o2) {
        Object tmp = o1;
        o1 = o2;
        o2 = tmp;
    }

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

```

1.3. Classifier Member and Instance Member

- Tại class DigitalVideoDisc thêm instance như sau:

```

private static int nbDigitalVideoDiscs = 0;
private int id;

```

```

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

public DigitalVideoDisc(String category, String title, float cost) {
    this.category = category;
    this.title = title;
    this.cost = cost;
    nbDigitalVideoDiscs++;
    this.id = nbDigitalVideoDiscs;
}

public DigitalVideoDisc(String director, String category, String title, float cost) {
    this.director = director;
    this.category = category;
    this.title = title;
    this.cost = cost;
    nbDigitalVideoDiscs++;
    this.id = nbDigitalVideoDiscs;
}

public DigitalVideoDisc(String title, String category, String director, int length, float cost) {
    this.title = title;
    this.category = category;
    this.director = director;
    this.length = length;
    this.cost = cost;
    nbDigitalVideoDiscs++;
    this.id = nbDigitalVideoDiscs;
}
}

```

1.4. Print and Search Cart

1.4.1. Print

- Viết lại hàm toString() để thay đổi kiểu trả về của string ở trong class DigitalVideoDisc

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

public boolean isMatch(String title) {
    return this.title.equalsIgnoreCase(title);
}
```

- Tạo phương thức print() in danh sách các mặt hàng trong class Cart

```
public void print() {
    System.out.println("*****CART*****");
    System.out.println("Ordered Items:");
    float totalCost = 0;
    for (int i = 0; i < itemsOrdered.size(); i++) {
        DigitalVideoDisc dvd = itemsOrdered.get(i);
        System.out.println((i + 1) + ". " + dvd);
        totalCost += dvd.getCost();
    }
    System.out.println("Total cost: " + totalCost + " $");
    System.out.println("*****");
}
```

1.4.2. Search

- Tìm kiếm theo ID và tìm kiếm theo Title ở trong class Cart

```

public void searchById(int id) {
    for (DigitalVideoDisc dvd : itemsOrdered) {
        if (dvd.getId() == id) {
            System.out.println("Found DVD: " + dvd);
            return;
        }
    }
    System.out.println("No DVD found with ID: " + id);
}

public void searchByTitle(String title) {
    boolean found = false;
    for (DigitalVideoDisc dvd : itemsOrdered) {
        if (dvd.isMatch(title)) {
            System.out.println("Found DVD: " + dvd);
            found = true;
        }
    }
    if (!found) {
        System.out.println("No DVD found with title: " + title);
    }
}
}

```

- Tạo class CartTest để kiểm tra các hàm trên

```

public class CartTest {
    public static void main(String[] args) {

        Cart cart = new 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);

        cart.print();
        cart.searchById(1);
        cart.searchByTitle("Star");
    }
}

```

1.5. Implement the Store class

- Tạo class Store, chứa thuộc tính itemsInStore[] là một mảng các DVD có sẵn trong cửa hàng. Thêm 2 phương thức addDVD và removeDVD

```

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

    public void addDVD(DigitalVideoDisc dvd) {
        itemsInStore.add(dvd);
        System.out.println("DVD \"" + dvd.getTitle() + "\" has been added to the store.");
    }

    public void removeDVD(DigitalVideoDisc dvd) {
        if (itemsInStore.remove(dvd)) {
            System.out.println("DVD \"" + dvd.getTitle() + "\" has been removed from the store.");
        } else {
            System.out.println("DVD \"" + dvd.getTitle() + "\" is not found in the store.");
        }
    }

    public void printStore() {
        System.out.println("*****STORE*****");
        System.out.println("Available DVDs in the Store:");
        for (int i = 0; i < itemsInStore.size(); i++) {
            DigitalVideoDisc dvd = itemsInStore.get(i);
            System.out.println((i + 1) + ". " + dvd);
        }
        System.out.println("*****");
    }

    public void searchByTitle(String keyword) {
        boolean matchFound = false;
        System.out.println("Searching for DVDs with \"" + keyword + "\" in the title:");
        for (DigitalVideoDisc dvd : itemsInStore) {
            if (dvd.isMatch(keyword)) {
                System.out.println("Found: " + dvd);
                matchFound = true;
            }
        }
        if (!matchFound) {
            System.out.println("No DVDs found with \"" + keyword + "\" in the title.");
        }
    }
}

```

- Tạo class StoreTest để kiểm tra 2 phương thức trên

```

public class StoreTest {
    public static void main(String[] args) {

        Store store = new Store();

        DigitalVideoDisc dvd1 = new DigitalVideoDisc("Jungle", "Adventure", "John Doe", 120, 19.99f);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Cinderella", "Animation", "Jane Smith", 90, 14.99f);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Mission Impossible", "Action", "Tom Cruise", 110, 24.99f);
        DigitalVideoDisc dvd4 = new DigitalVideoDisc("Avatar", "Sci-Fi", "James Cameron", 150, 29.99f);

        store.addDVD(dvd1);
        store.addDVD(dvd2);
        store.addDVD(dvd3);

        store.printStore();

        store.removeDVD(dvd2);

        store.printStore();

        store.removeDVD(dvd4);

        store.addDVD(dvd4);

        store.printStore();

        store.searchByTitle("Mission");

        store.searchByTitle("Star Wars");

    }
}

```


1.6. String, StringBuilder and StringBuffer

- Tạo một lớp mới tên là ConcatenationInLoops để kiểm tra thời gian xử lý khi xây dựng chuỗi String bằng toán tử +, StringBuilder và StringBuffer.

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

- Tạo class GarbageCreator

```
public class GarbageCreator {  
    public static void readLargeFile() {  
        String filename = "very_large_file.txt";  
        try {  
            byte[] inputBytes = Files.readAllBytes(Paths.get(filename));  
            String content = new String(inputBytes);  
        } catch (IOException e) {  
            System.out.println("Error reading file: " + e.getMessage());  
        }  
    }  
}
```

- Tạo class NoGarbage

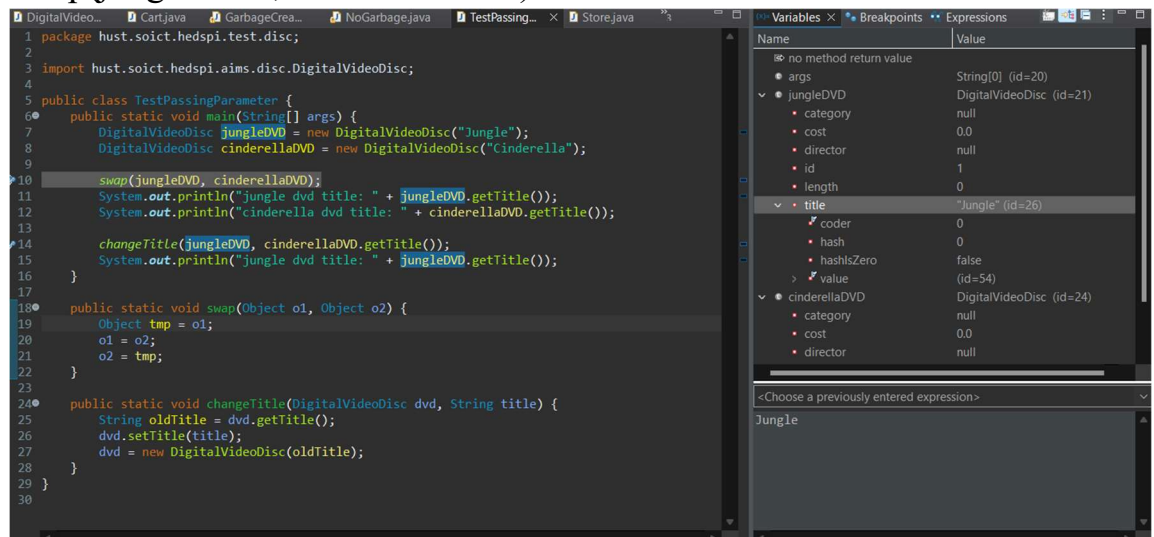
```

public class NoGarbage {
    public static void readLargeFile() {
        String filename = "very_large_file.txt";
        try {
            byte[] inputBytes = Files.readAllBytes(Paths.get(filename));
            StringBuilder sb = new StringBuilder();
            for (byte b : inputBytes) {
                sb.append((char)b);
            }
            String content = sb.toString();
        } catch (IOException e) {
            System.out.println("Error reading file: " + e.getMessage());
        }
    }
}

```

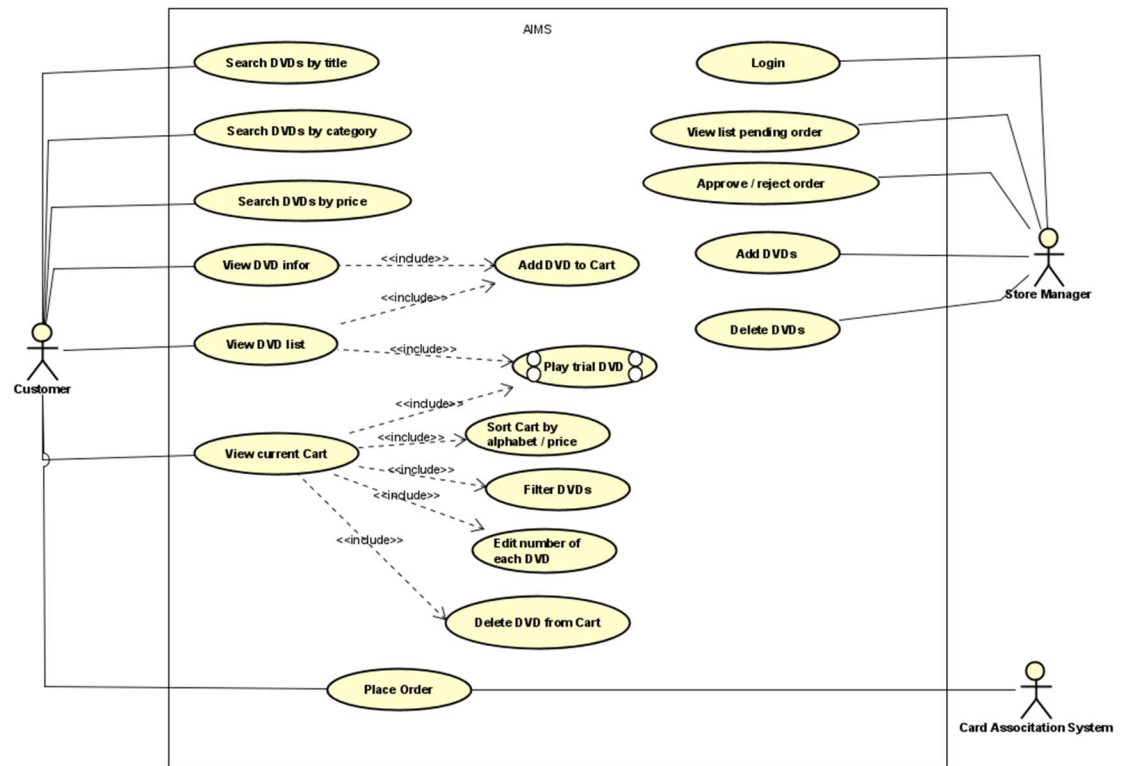
2. Debug

- Debug tại class TestPassingParameter. Đặt breakpoint tại swap(jungleDVD, cinderellaDVD)

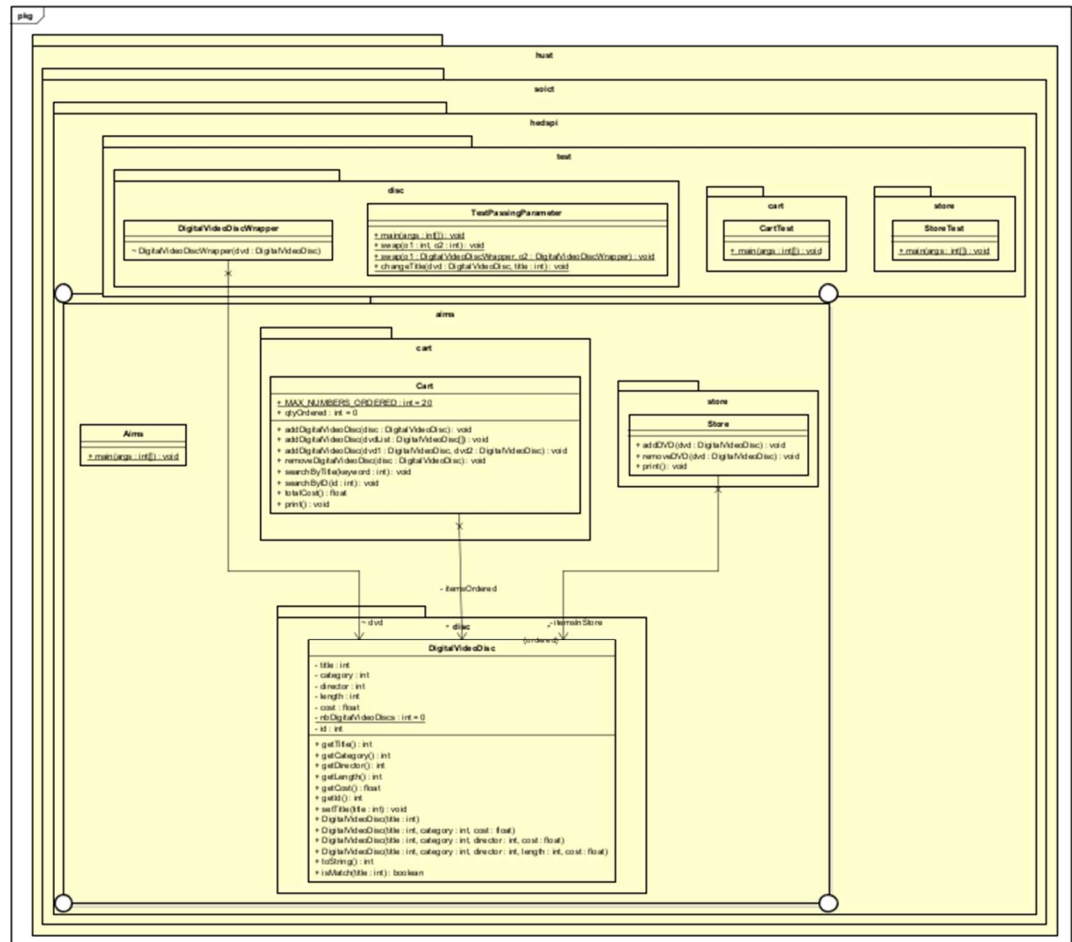


3. Diagram

3.1. Use case Diagram



3.2. Class Diagram



4. Answer Questions

- Is JAVA a Pass by Value or a Pass by Reference programming language?

Java is a pass-by-value programming language.

- After the call of swap(jungleDVD, cinderellaDVD), why do the titles of these two objects remain unchanged?

Bởi vì phương thức swap() chỉ hoán đổi giá trị của các trường title giữa hai đối tượng, chứ không thay đổi chính các tham chiếu đối tượng. Do đó, các tham chiếu đối tượng jungleDVD và cinderellaDVD vẫn trỏ đến cùng các đối tượng trong bộ nhớ như trước khi phương thức swap() được gọi.

- After the call of changeTitle(jungleDVD, cinderellaDVD.getTitle()), why is the title of jungleDVD changed?

Bởi vì phương thức `changeTitle()` trực tiếp sửa đổi trường `title` của đối tượng `jungleDVD` bằng cách sử dụng phương thức `setter`.

- Write a `toString()` method for the `DigitalVideoDisc` class. What should be the return type of this method?

Kiểu trả về của phương thức này là `String`.