Họ và tên: Nguyễn Hồng Quân

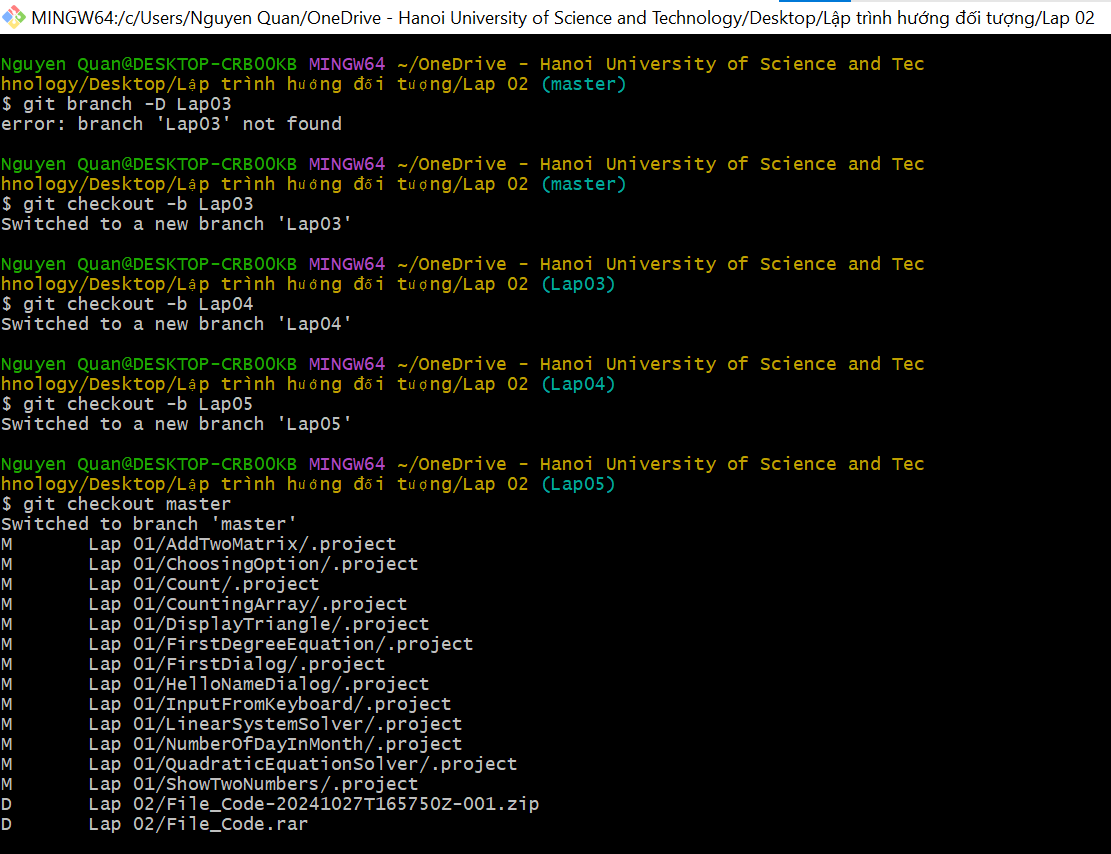
Mã số sinh viên: 20225908

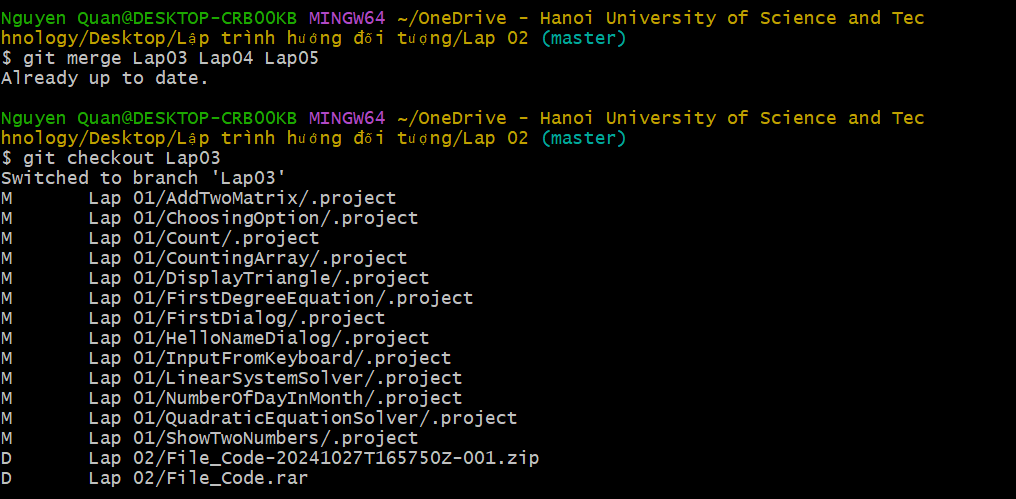
Lớp: 744521 – Lập trình hướng đối tượng – IT3103

LAP 03: Basic Object – Oriented Techniques

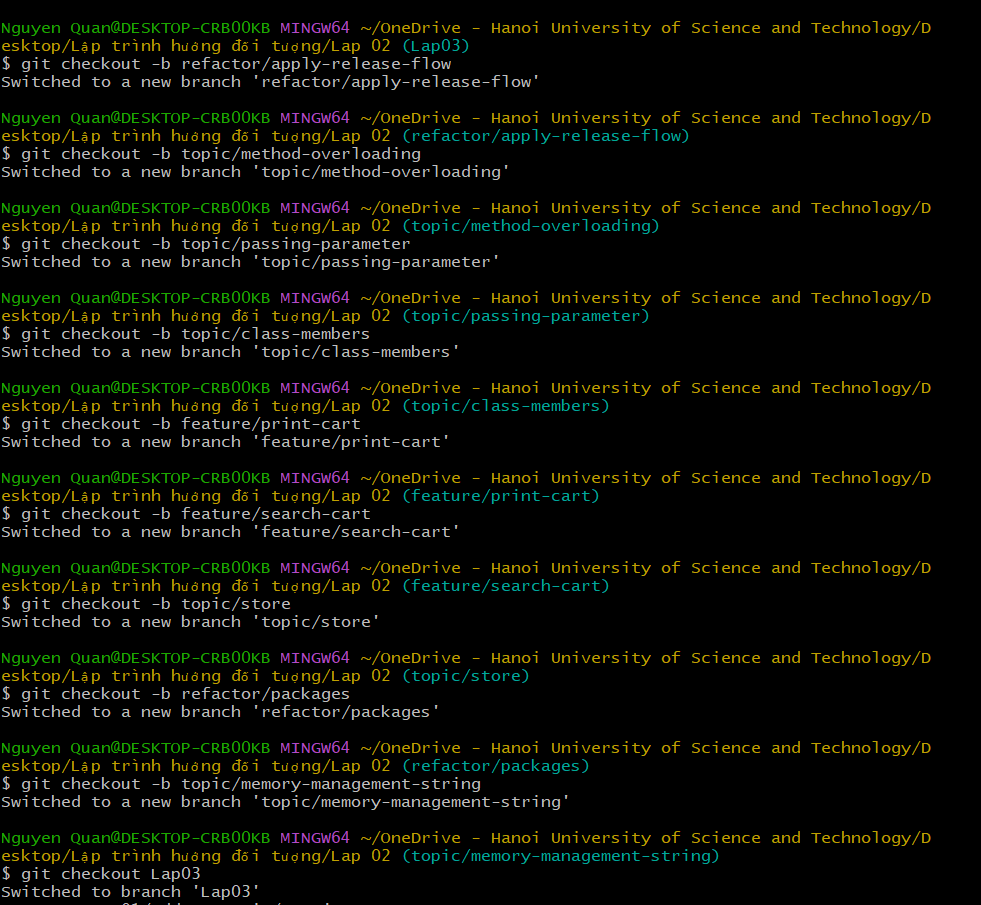
I. Tạo các branch trong project

1. Tạp branch con Lap03, Lap04, Lap05 trong branch cha trong master

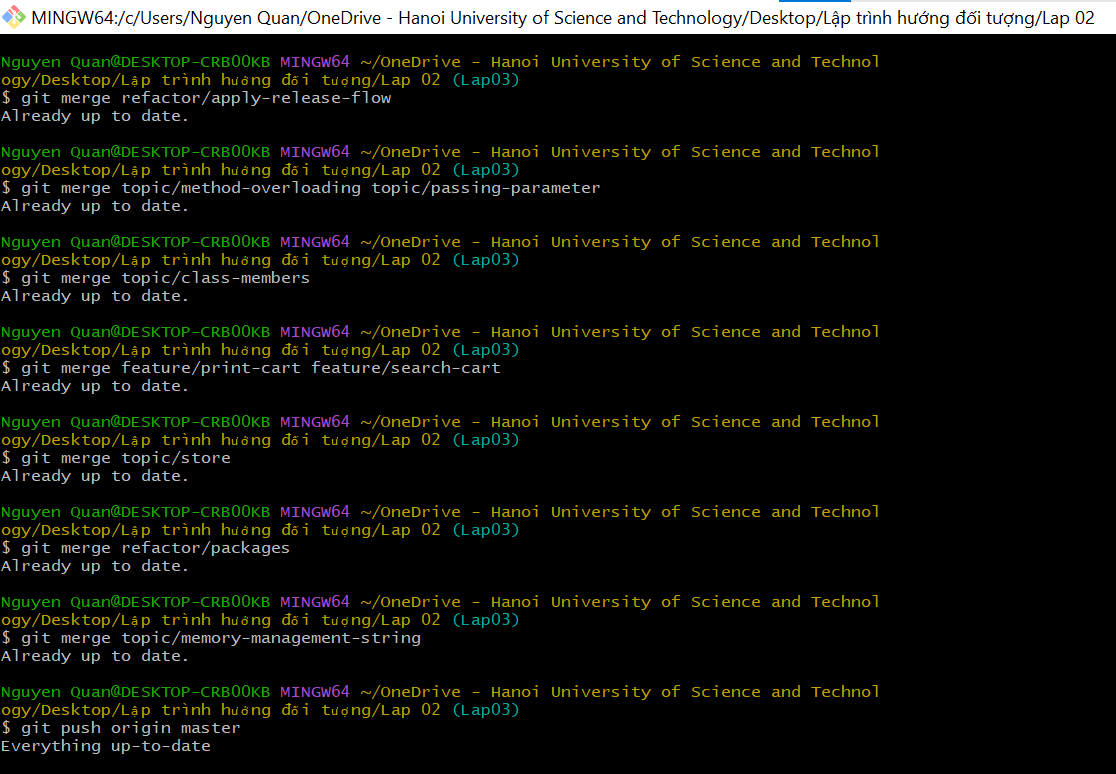




2. Tạo 9 branch trong Lap03 theo yêu cầu



3. Trộn branch



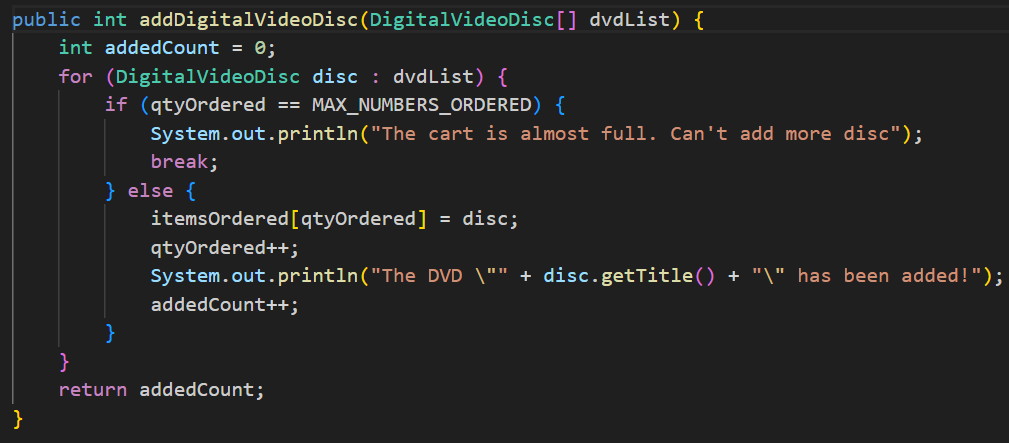
4. Push lên github

Cách push: Mở Git Gui, checkout all branch, sau đó chọn các branch rồi up.

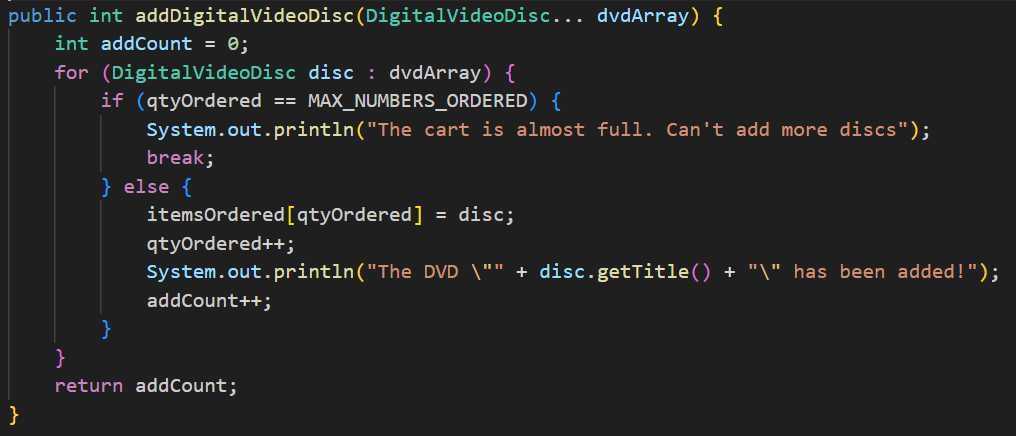
II. Working with method overloading

1. Overloading by differing types of parameter

Phương thức mới đầu tiên addDigitalVideoDisc (DigitalVideoDisc[] dvdList) cho phép thêm một danh sách các đĩa DVD vào giỏ hàng.

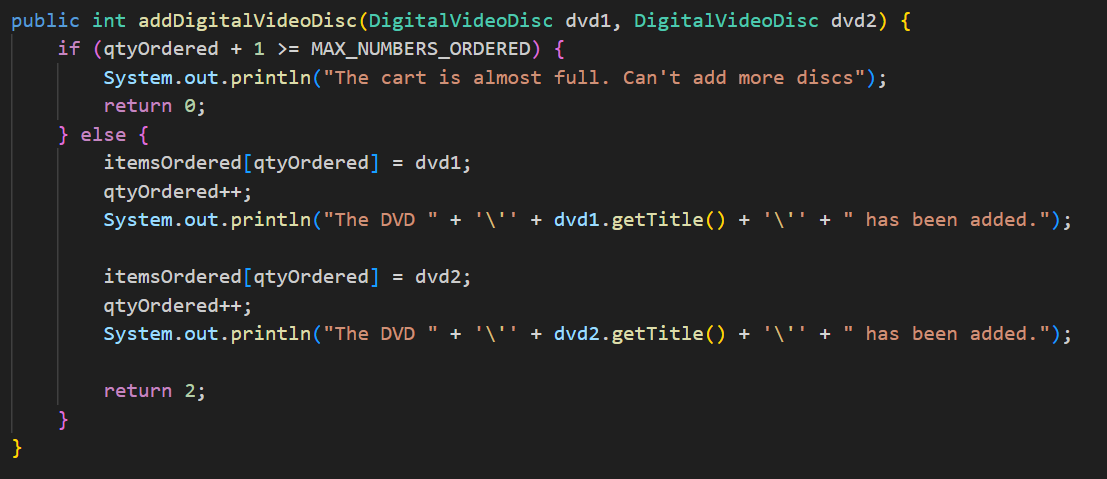


Phương thức mới thứ hai addDigitalVideoDisc (DigitalVideoDisc... dvdArray) sử dụng varargs để cho phép thêm một số lượng tùy ý các đĩa DVD vào giỏ hàng.



Cách sử dụng varargs linh hoạt hơn vì nó cho phép thêm bất kì số lượng đĩa DVD mà không cần chỉ định mảng một cách rõ ràng. Điều này làm cho mã nguồn sạch sẽ hơn và thuận tiện hơn khi gọi phương thức.

2. Overloading by differing the number of parameters

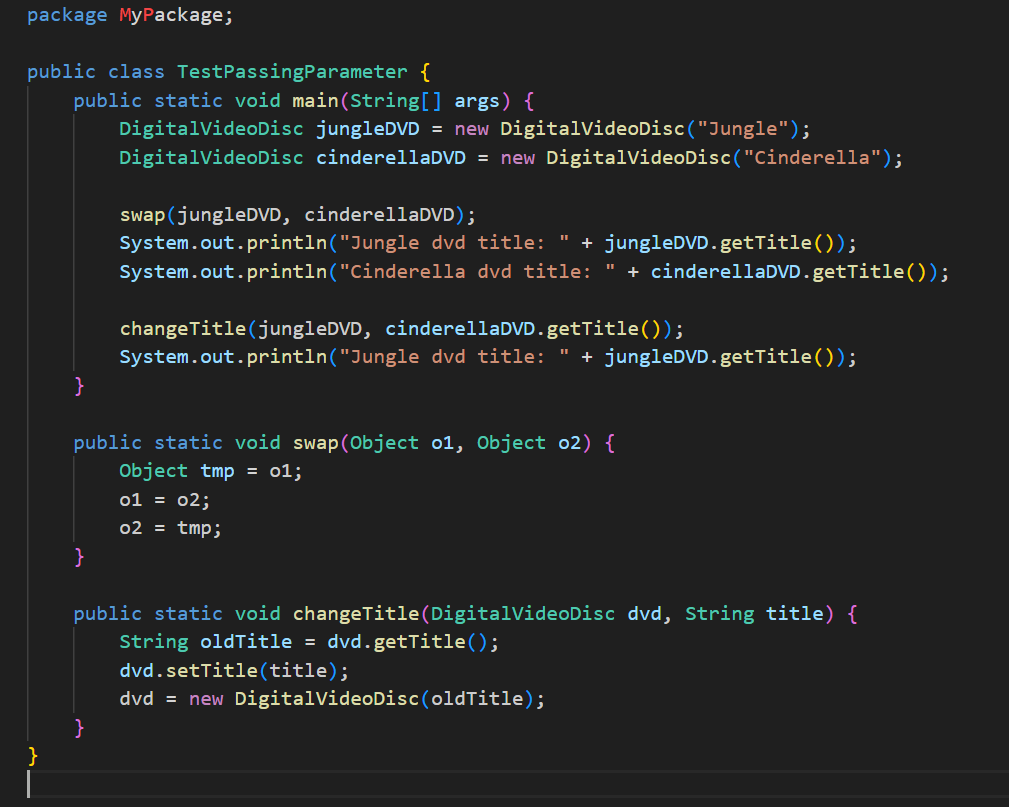


III. Passing parameter

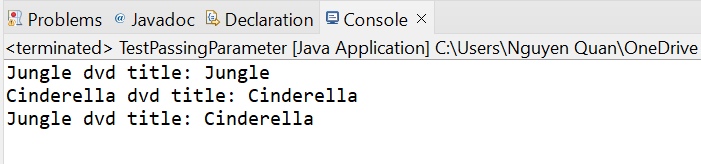
1. Java là ngôn ngữ lập trình “Pass by Value” hay “Pass by Reference”

Java là ngôn ngữ lập trình "Pass by Value". Trong Java, khi bạn truyền một tham số cho một phương thức, giá trị của tham số được sao chép và truyền vào phương thức. Điều này có nghĩa là nếu bạn thay đổi giá trị của tham số bên trong phương thức, giá trị của biến gọi phương thức không bị ảnh hưởng.

2. Tạo class TestPassingParameter



Kết quả khi chạy:



* Có vấn đề với swap.

3. Giải quyết vấn đề trong đoạn code này

3.1. Trả lời câu hỏi

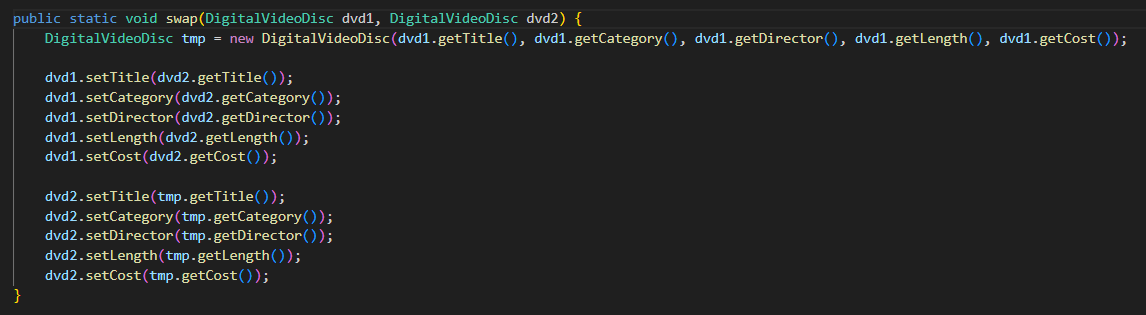
3.1.1 After the call of swap(jungleDVD, cinderellaDVD) why does the title of these two objects still remain?

*Trả lời:* Sau khi thực hiện phương thức swap(jungleDVD, cinderellaDVD), tiêu đề của hai đối tượng vẫn giữ nguyên vì trong Java, tham số được truyền vào phương thức là giá trị của đối tượng, không phải là tham chiếu đến đối tượng. Khi ta thay đổi giá trị của tham số bên trong phương thức (như việc đổi chỗ giữa o1 và o2), sự thay đổi này không ảnh hưởng đến giá trị của các đối tượng gốc.

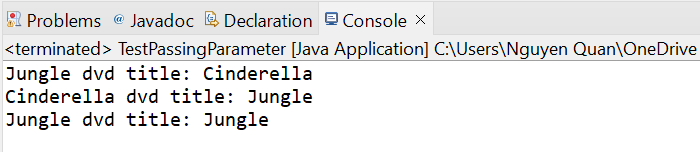
3.1.2 After the call of changeTitle(jungleDVD, cinderellaDVD.getTitle()) why is the title of the JungleDVD changed?\*

*Trả lời:* Sau khi gọi changeTitle(jungleDVD, cinderellaDVD.getTitle()), tiêu đề của jungleDVD bị thay đổi vì trong phương thức changeTitle, ta thực hiện thay đổi trực tiếp trên đối tượng dvd (được truyền vào phương thức) bằng cách gọi dvd.setTitle(title). Điều này ảnh hưởng trực tiếp đến đối tượng gốc được truyền vào phương thức.

3.2 Sửa lại hàm wap

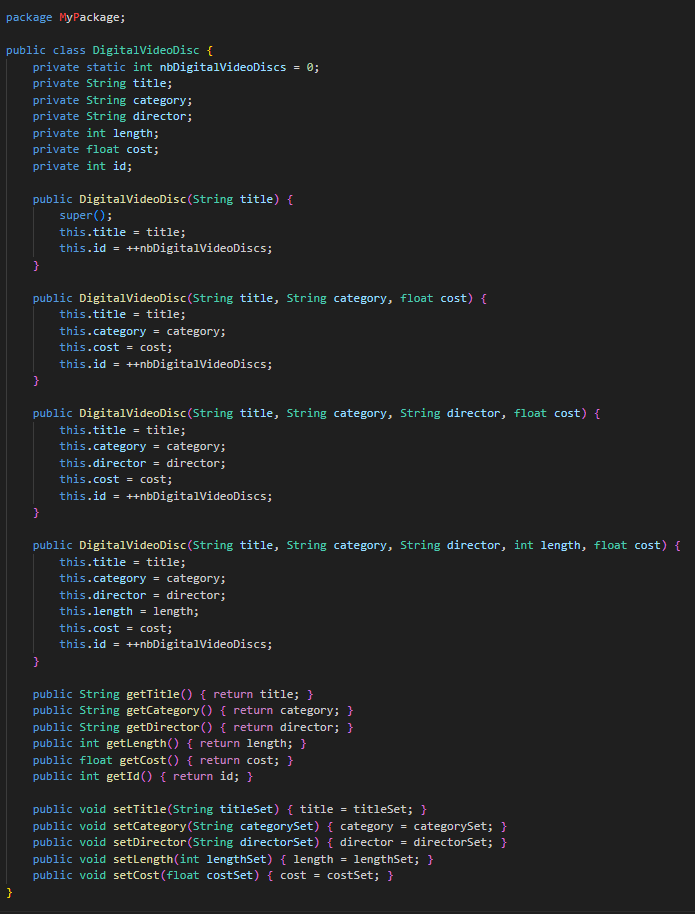


Kết quả sau khi sửa:



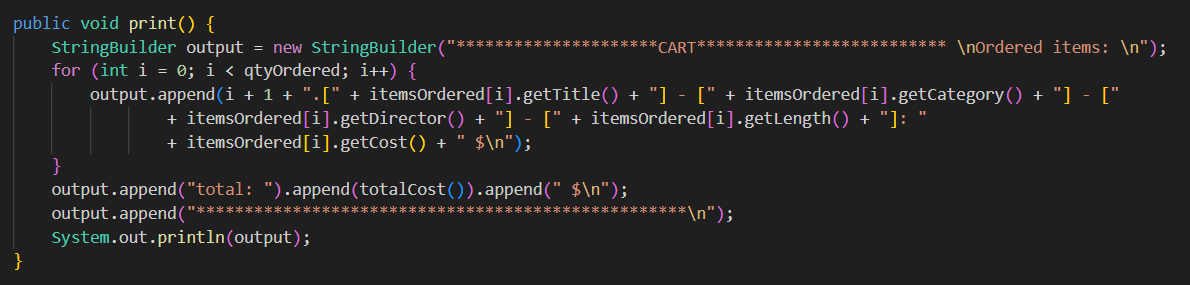
IV. Sử dụng Debug

V. Classifier Member and Instance Member



VI. Open the Cart class

1. Thêm hàm print vào Cart.java

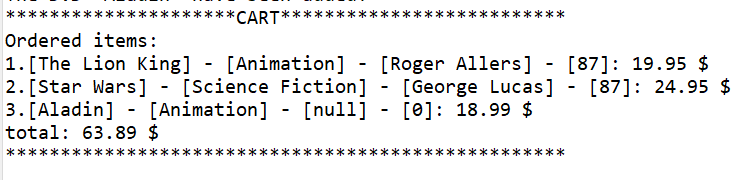


2. Kết quả

Phương thức print() được gọi ở TestCart.java

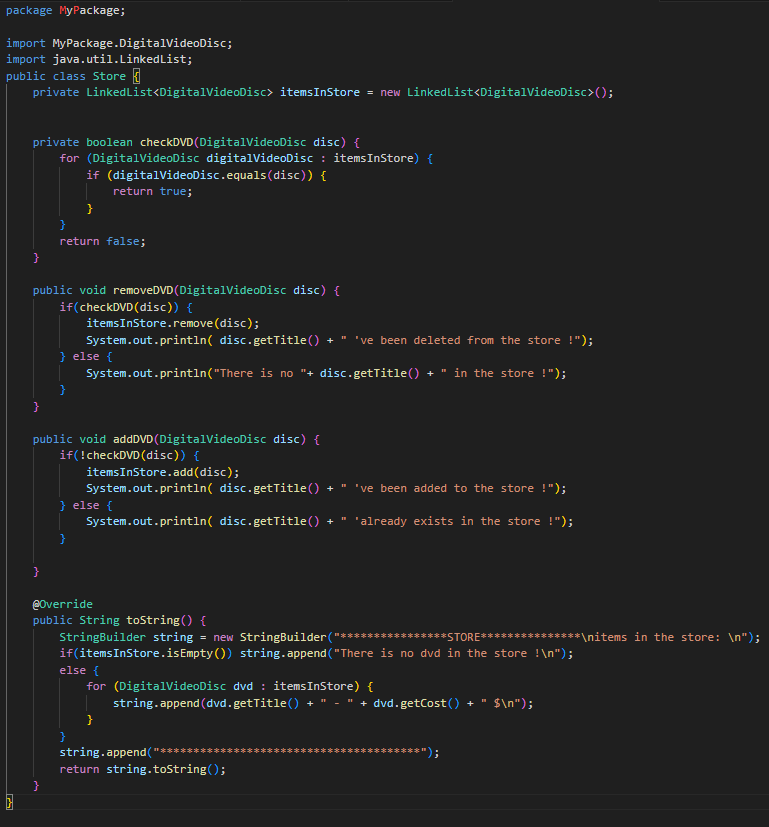


Kết quả:



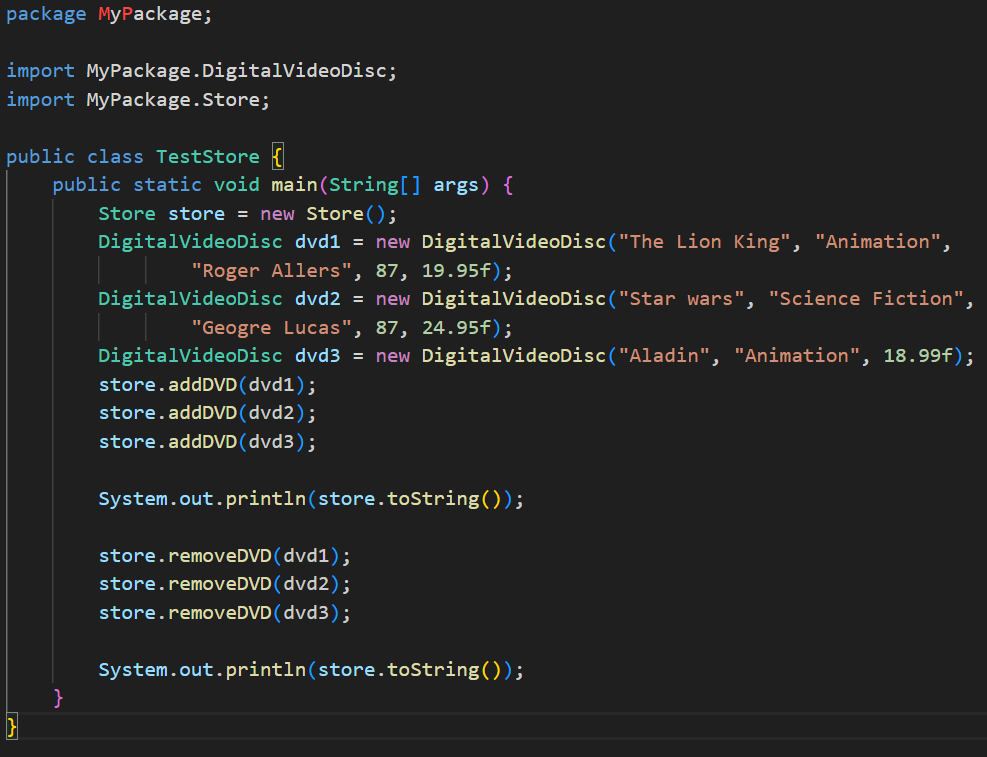
VII. Implement the Store class

1. Tạo lớp Store

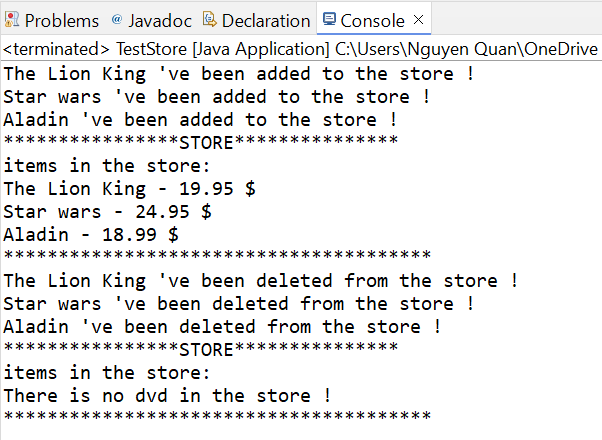


2. Tạo lớp TestStore và kiểm thử

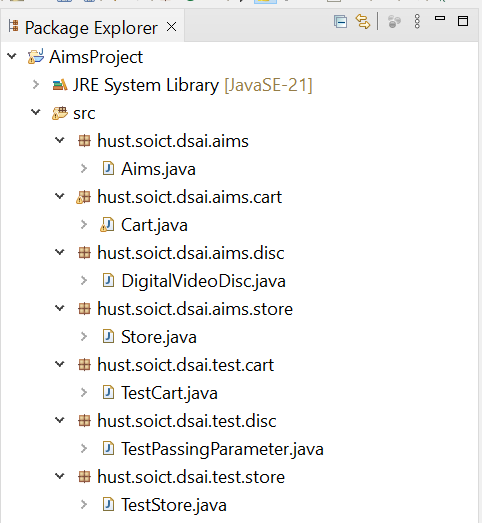
2.1 Tạo lớp TestStore



2.2 Kết quả

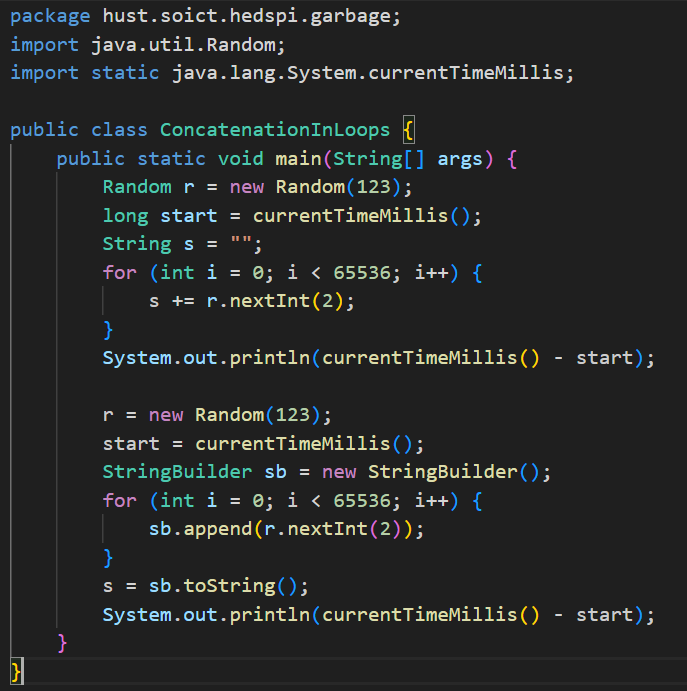


VIII. Re-organize your project

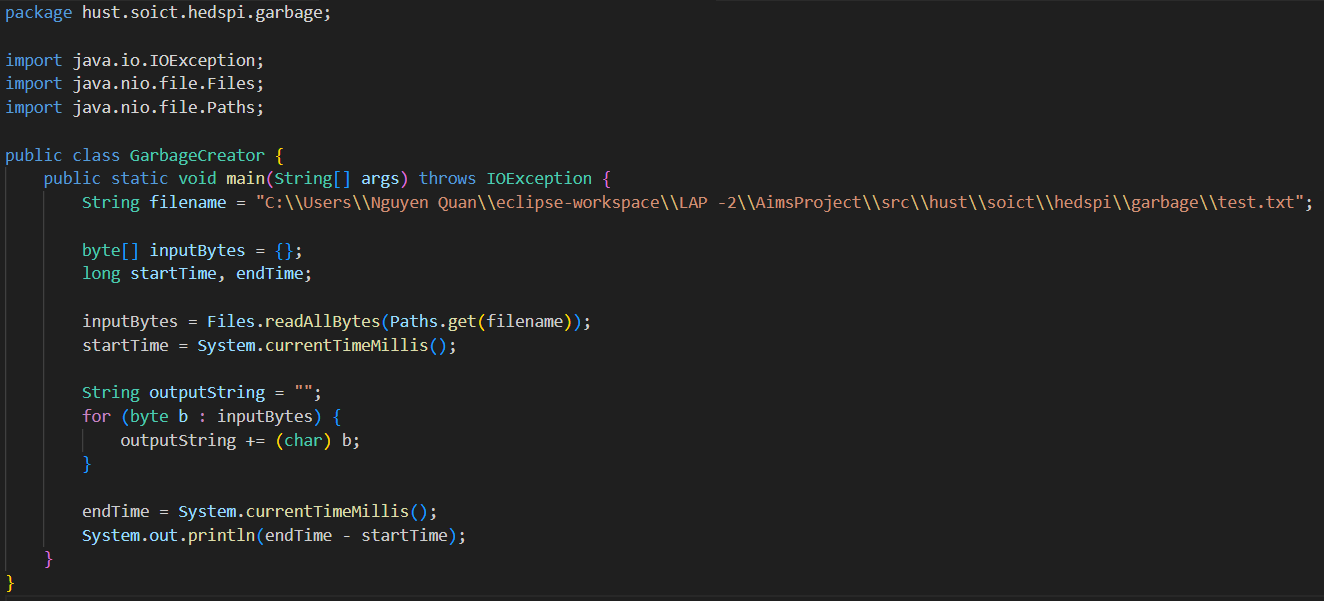


IX. String, StringBuilder and StringBuffer

1. Tạo lớp ConcatenationInLoops



2. Tạo lớp GarbageCreator



3. Tạo lớp NoGarbage



X. Release flow demonstration