BÁO CÁO THỰC HÀNH LAB 3

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Table of Contents

[2. Working with method overloading 1](#_Toc151160342)

[3. Passing parameter 2](#_Toc151160343)

[4. Use debug run 4](#_Toc151160344)

[5. Classifier member and instance member 6](#_Toc151160345)

[6.1. Print the cart 9](#_Toc151160346)

[6.2. Search for disc in the cart 10](#_Toc151160347)

[7. Implement the Store class 12](#_Toc151160348)

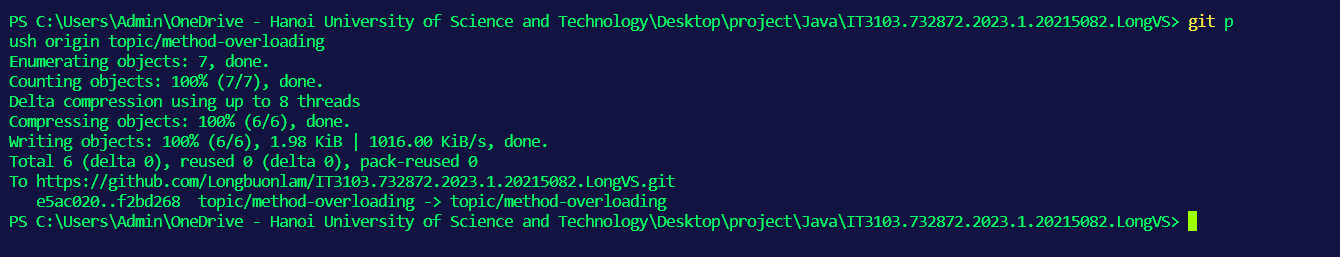
[8. Re-organize your projects 15](#_Toc151160349)

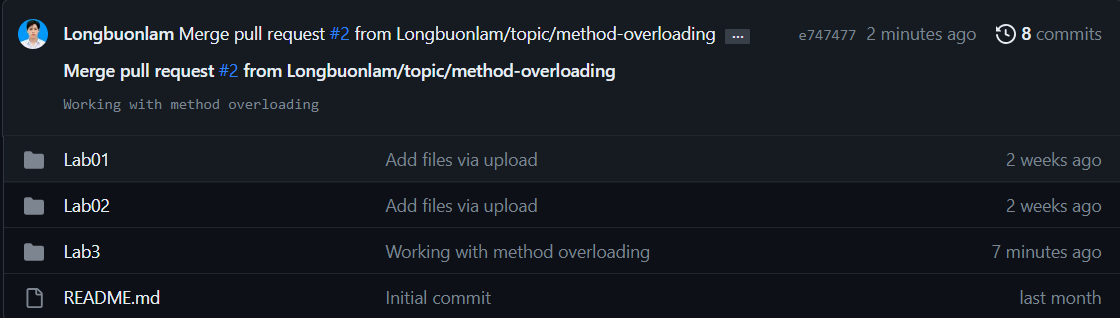
[9. String, StringBuilder and StringBuffer 17](#_Toc151160350)

# 

# 2. Working with method overloading

Create a branch topic/method-overloading for the exercise on method overloading





# 3. Passing parameter

Create a branch topic/passing-parameter for the exercise where you investigate on Java’s parameter passing

**Q1: Is JAVA a Pass by Value or a Pass by Reference programming language?**

Trong Java, khi truyền một kiểu dữ liệu nguyên thủy (ví dụ: int, float, char, v.v.) vào một phương thức, nó được truyền theo giá trị. Điều này có nghĩa là một bản sao của giá trị biến được truyền vào phương thức, và bất kỳ thay đổi nào được thực hiện trên tham số bên trong phương thức không ảnh hưởng đến biến gốc bên ngoài phương thức.

Khi truyền một đối tượng (bao gồm mảng) vào một phương thức, thì tham chiếu đến đối tượng được truyền theo giá trị. Điều này có nghĩa là tham chiếu bản sao được sao chép, nhưng cả tham chiếu gốc và tham chiếu sao chép vẫn trỏ đến cùng một đối tượng. Do đó, những thay đổi được thực hiện vào trạng thái đối tượng bên trong phương thức sẽ phản ánh ra ngoài phương thức.

Tóm lại, Java được gọi là "pass-by-value" đối với kiểu dữ liệu nguyên thủy và tham chiếu đến đối tượng, nhưng quan trọng là phải hiểu rằng khi truyền một đối tượng, bạn đang truyền một bản sao của tham chiếu, không phải là đối tượng thực sự.

**Q2: After the call of swap(jungleDVD, cinderellaDVD) why does the title of these two objects still remain?**

Trong phương thức swap, bạn đang cố gắng đổi chỗ các tham chiếu của hai đối tượng DigitalVideoDisc (jungleDVD và cinderellaDVD). Tuy nhiên, vì Java thực hiện truyền tham số theo giá trị, chỉ có giá trị của các tham chiếu (địa chỉ) được truyền vào phương thức swap. Do đó, việc đổi chỗ bên trong phương thức không ảnh hưởng đến các tham chiếu gốc bên ngoài phương thức. Phương thức swap không có ảnh hưởng đáng kể.

Sau lời gọi swap(jungleDVD, cinderellaDVD), các tiêu đề của jungleDVD vàcinderellaDVD vẫn giữ nguyên vì các đối tượng mà chúng tham chiếu không bị thay đổi.

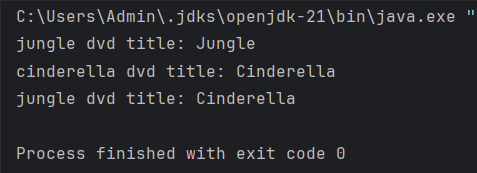
**Q3: After the call of changeTitle(jungleDVD, cinderellaDVD.getTitle()) why is the title of the JungleDVD changed?**

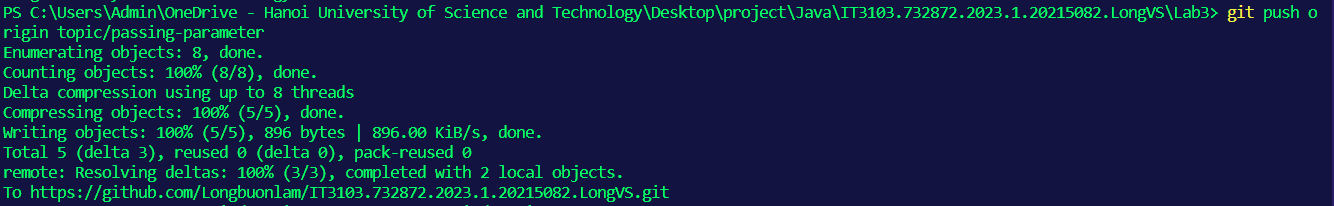
Trong phương thức changeTitle, ta đang sửa đổi tiêu đề của đối tượng DigitalVideoDisc được tham chiếu bởi tham số dvd. Tuy nhiên, khi bạn truyền jungleDVD vào phương thức này, một bản sao của tham chiếu đến đối tượng jungleDVD được nhận.

Bên trong phương thức changeTitle:

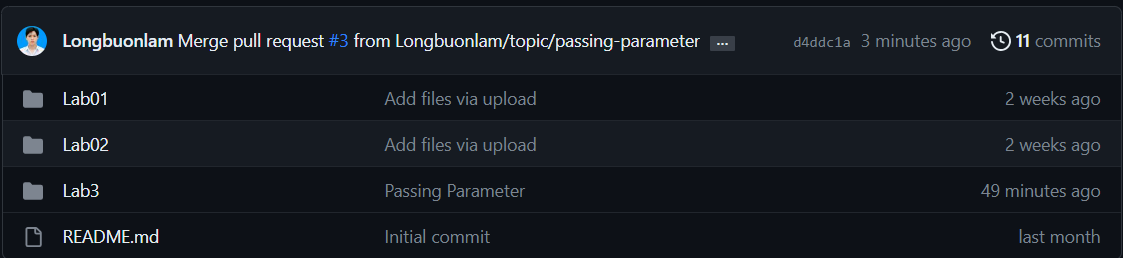
Phương thức thay đổi tiêu đề của đối tượng jungleDVD (vì đó là đối tượng được tham chiếu bởi tham số dvd), nhưng sau đó một đối tượng mới của DigitalVideoDisc được tạo ra với tiêu đề cũ, và tham số dvd được đặt để tham chiếu đến đối tượng mới này.Tuy nhiên, sự thay đổi này đối với tham số dvd không ảnh hưởng đến tham chiếu jungleDVD gốc bên ngoài phương thức.

Do đó, sau lời gọi changeTitle(jungleDVD, cinderellaDVD.getTitle()), tiêu đề của jungleDVD chỉ thay đổi trong phạm vi của phương thức, và tham chiếu jungleDVD gốc bên ngoài phương thức vẫn trỏ đến đối tượng gốc với tiêu đề ban đầu.









# 4. Use debug run

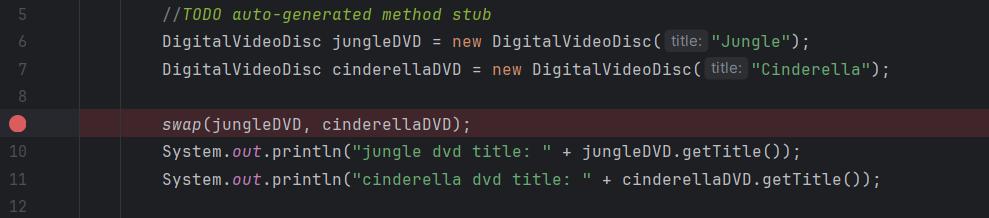
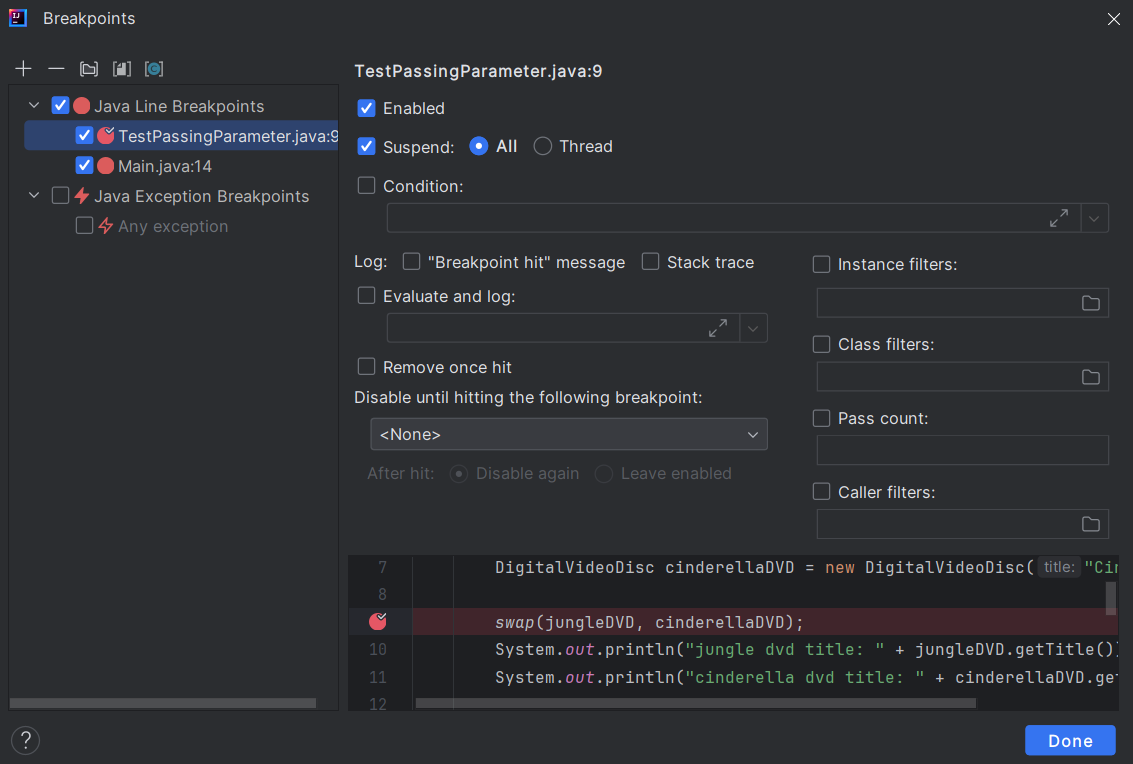


Figure : Breakpoint



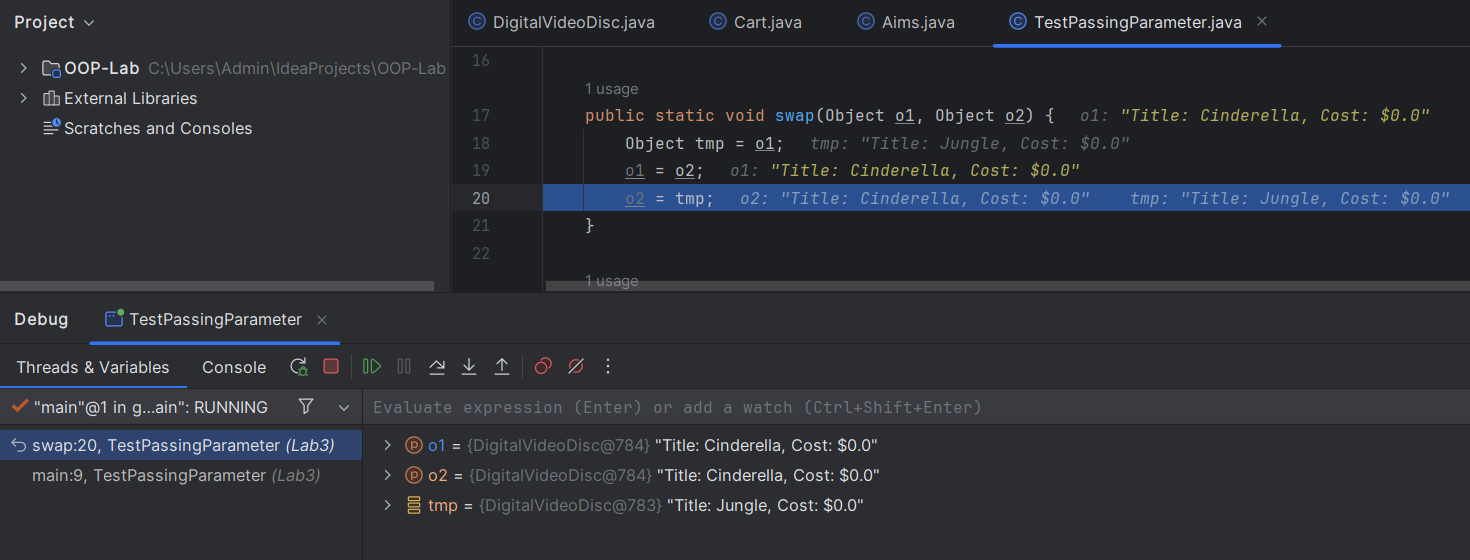
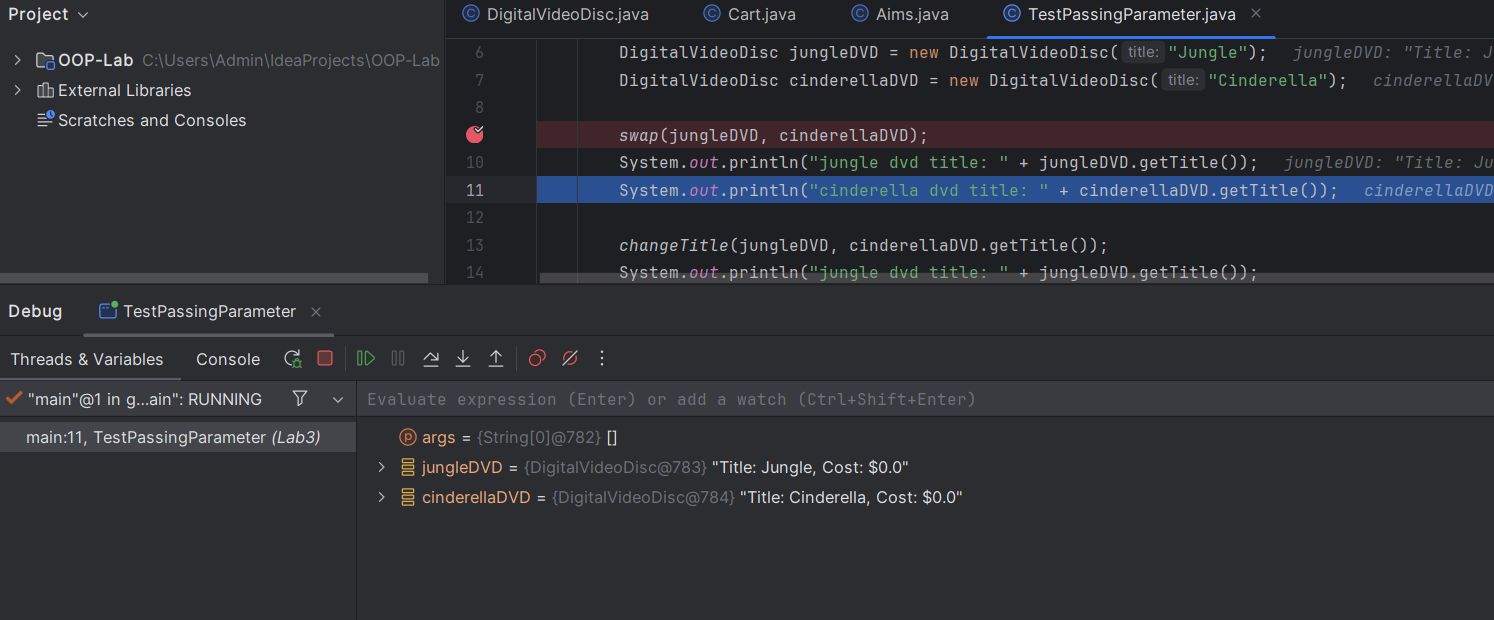


Figure : Step into



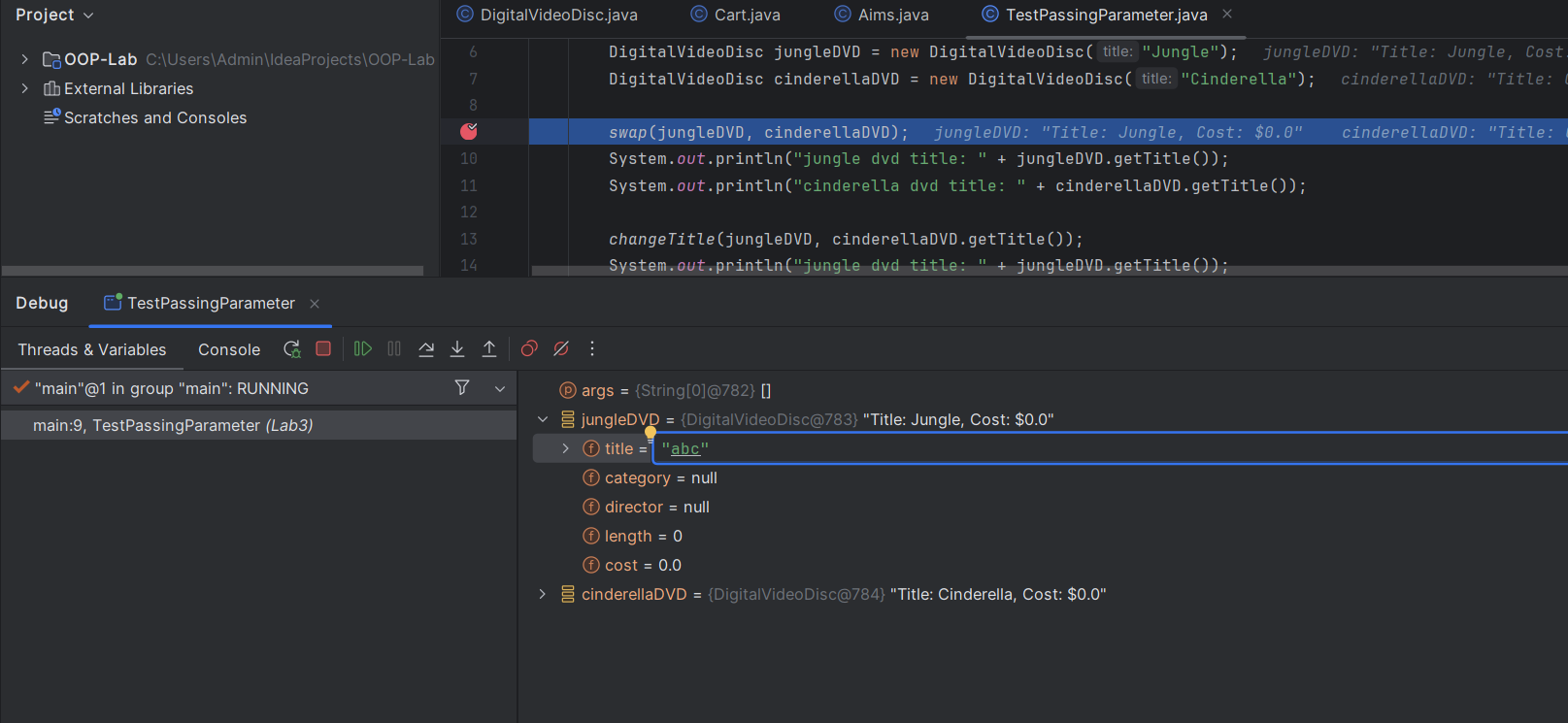


Figure : Change variable value



Figure : result

# 5. Classifier member and instance member

Create a branch topic/class-members for the exercise where you practice with classifier member and instance member

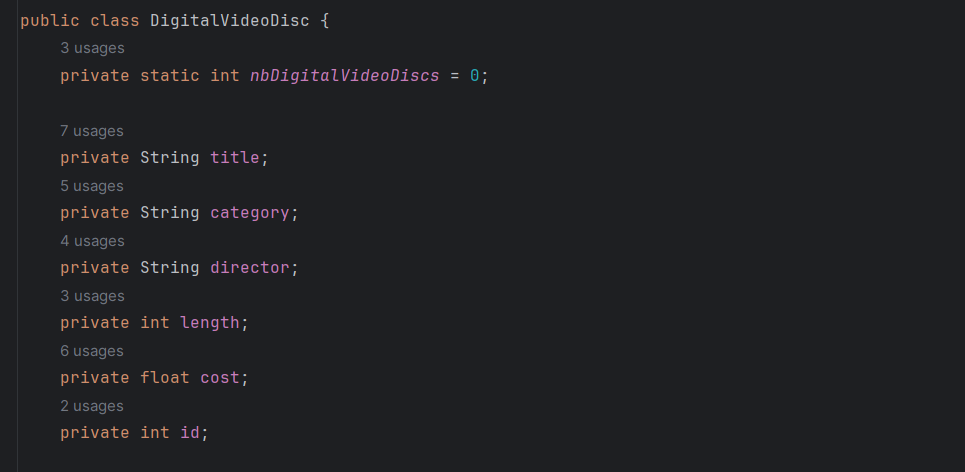


Figure : Add id attribute

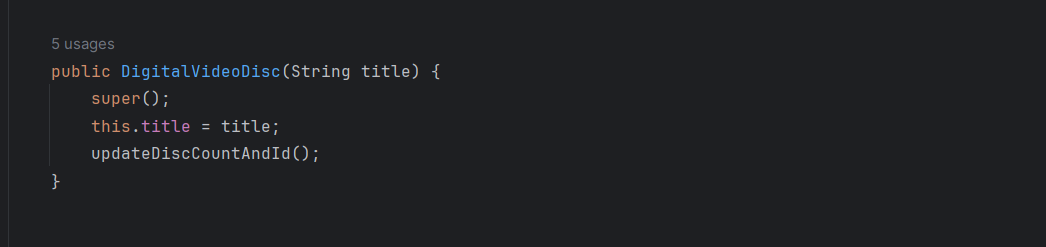


Figure :Modify class contructors

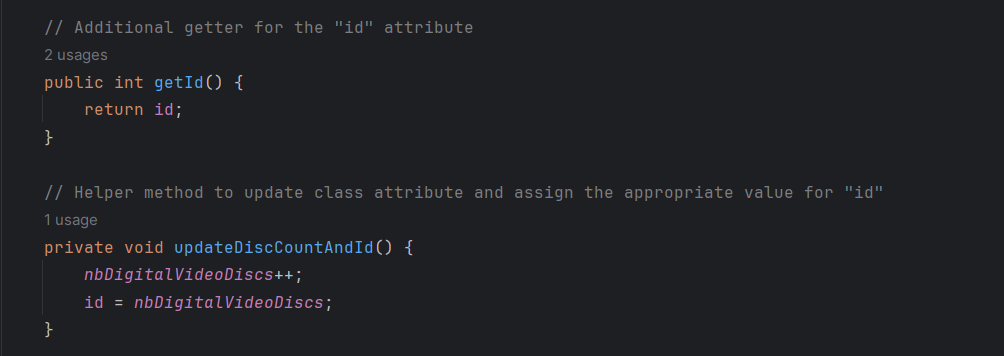


Figure : Increase id by 1 and getID function

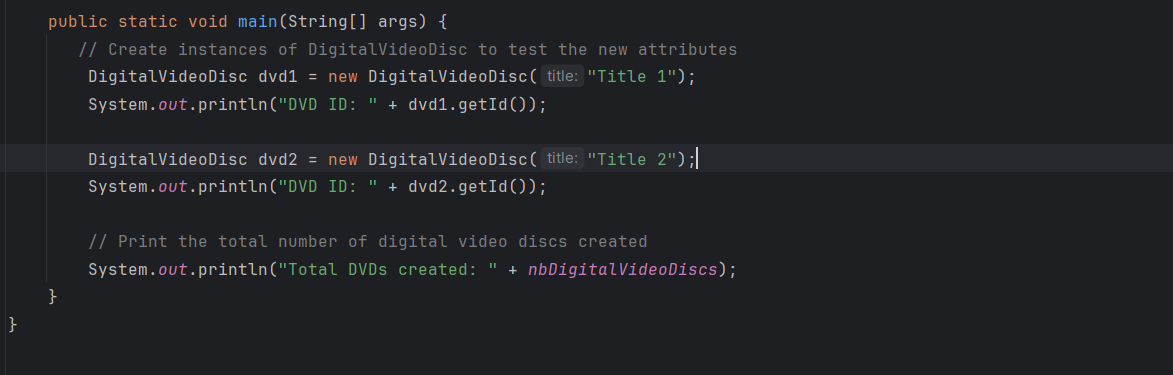


Figure : main method

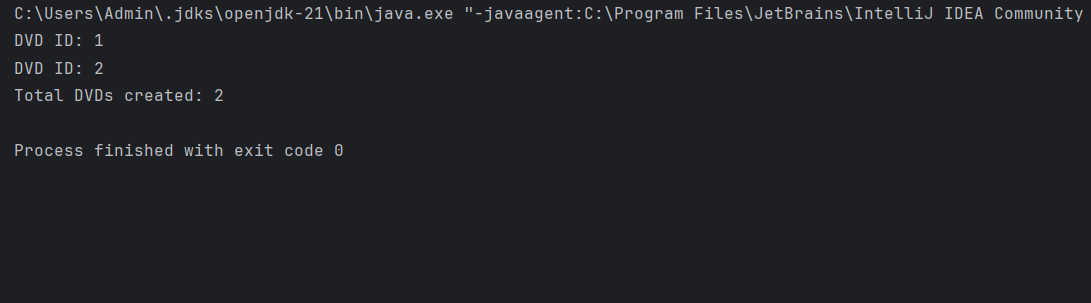
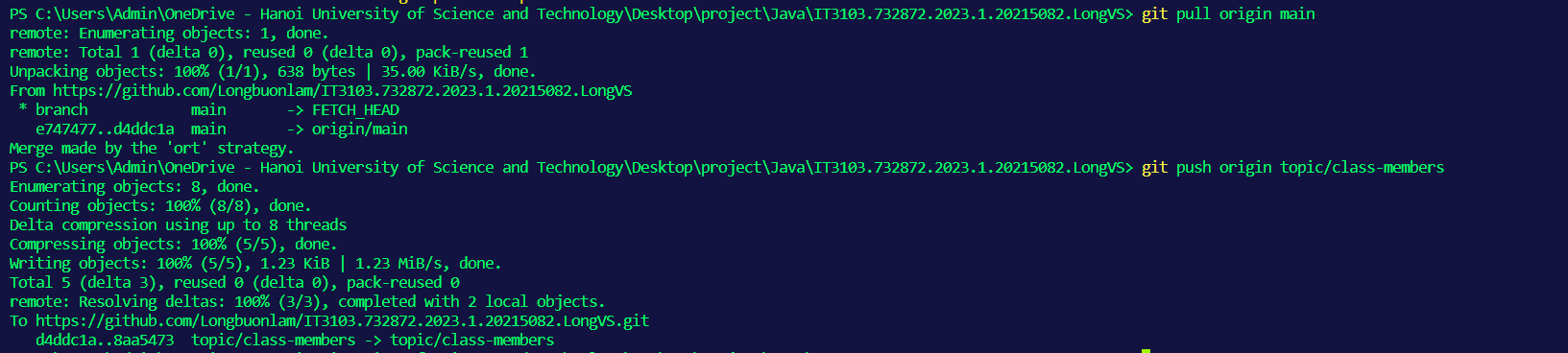
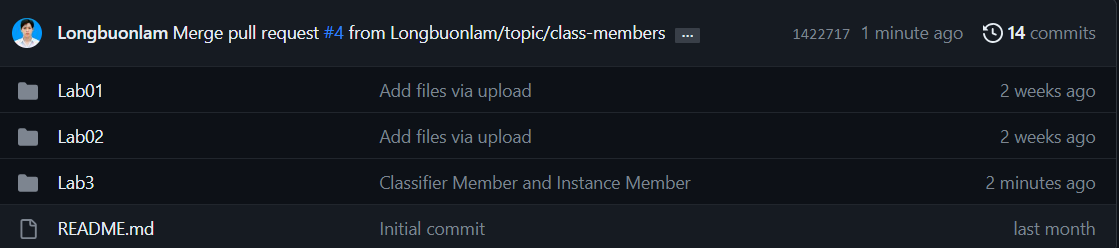


Figure : result







# 6.1. Print the cart

Create a branch feature/print-cart for the implementation of the print items in cart feature

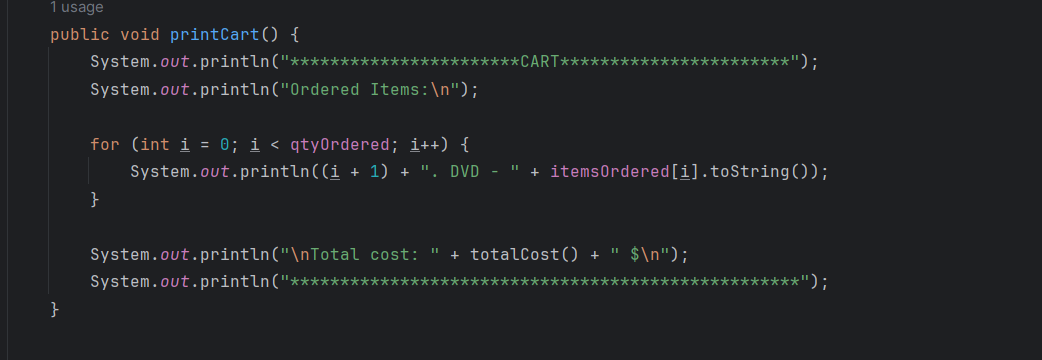


Figure : printCart()

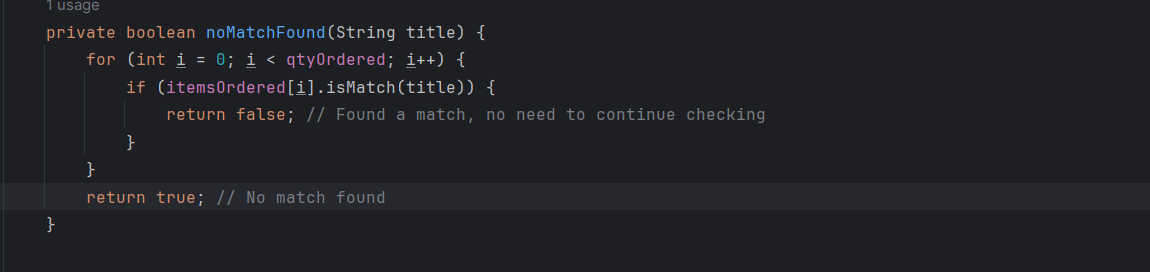


Figure : Check if there is no match

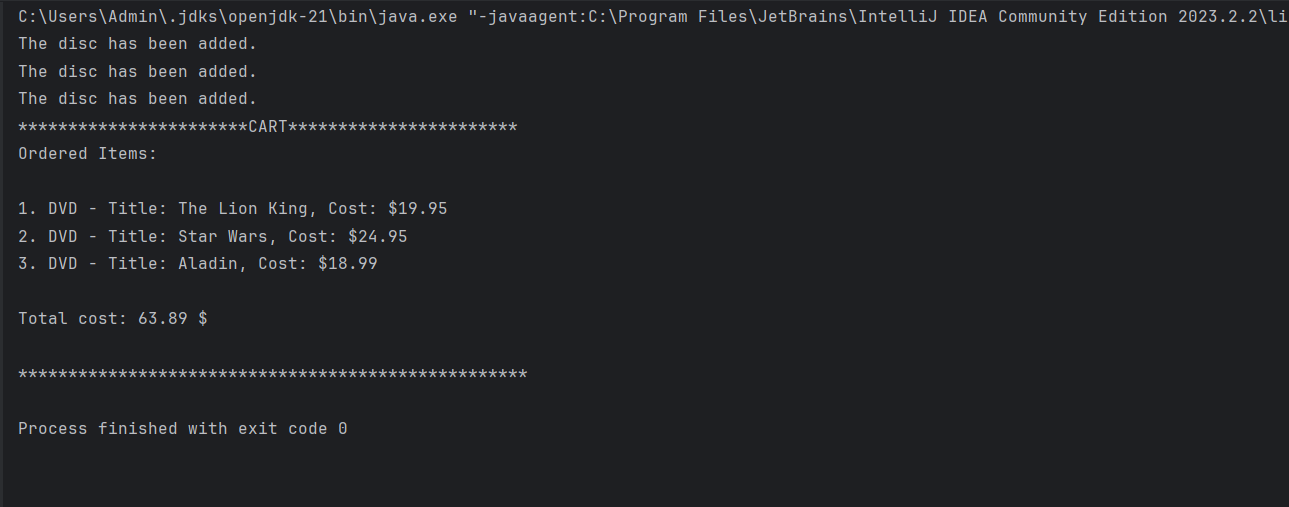
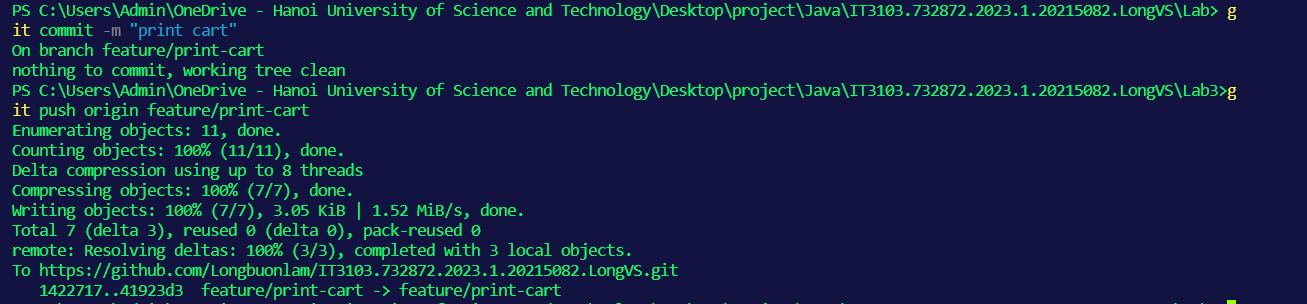
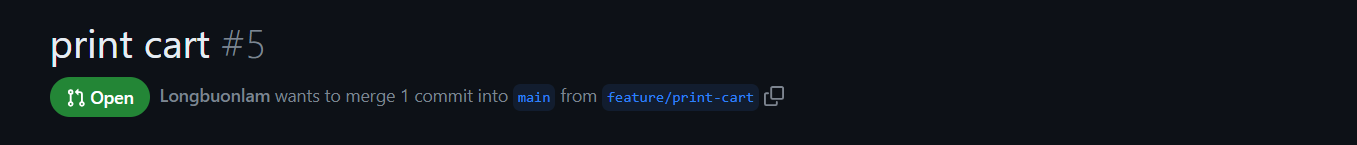
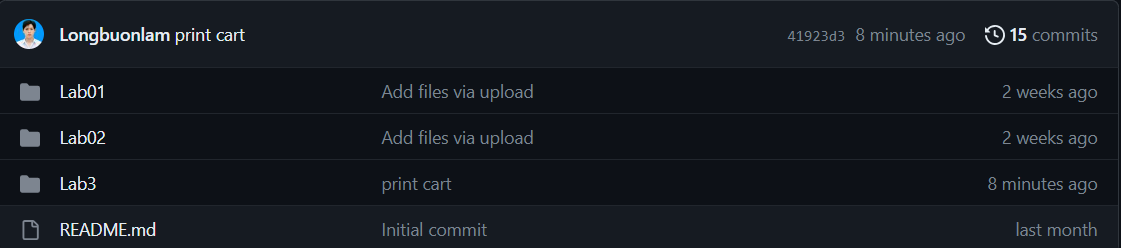


Figure : result







# 6.2. Search for disc in the cart

Create a branch feature/search-cart for the implementation of the search items in cart feature

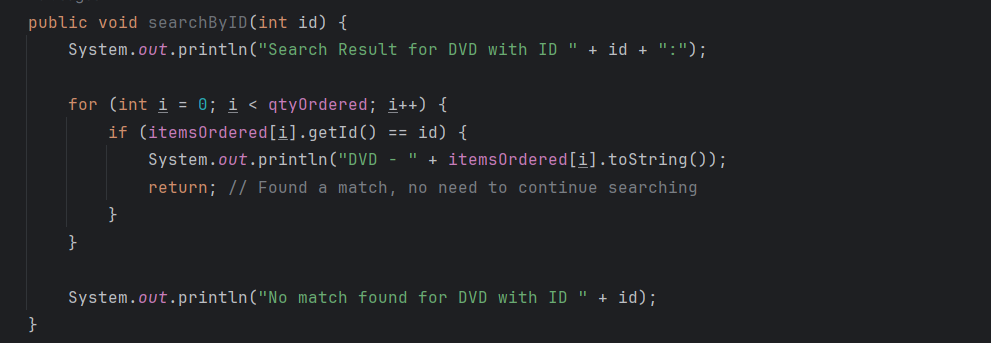


Figure : searchByID()

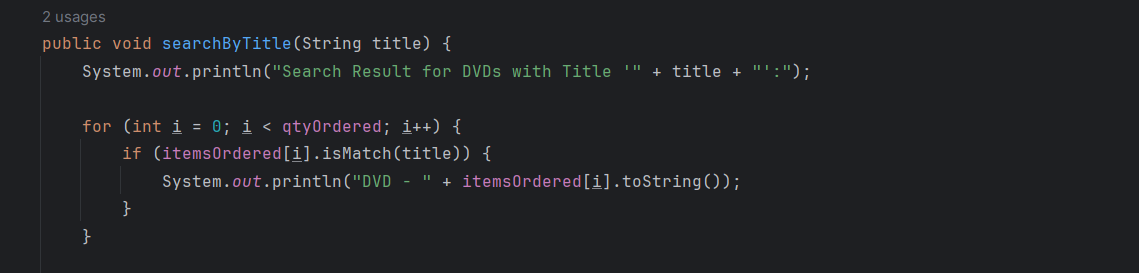
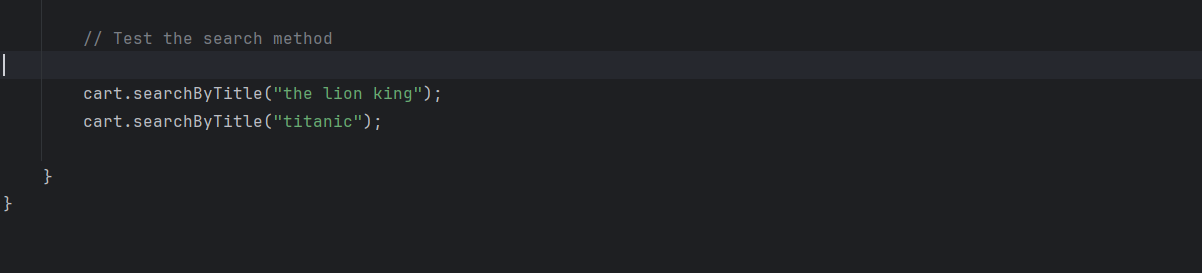


Figure : searchByTitle()



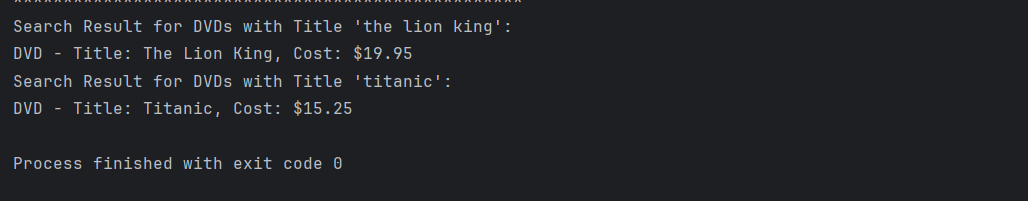
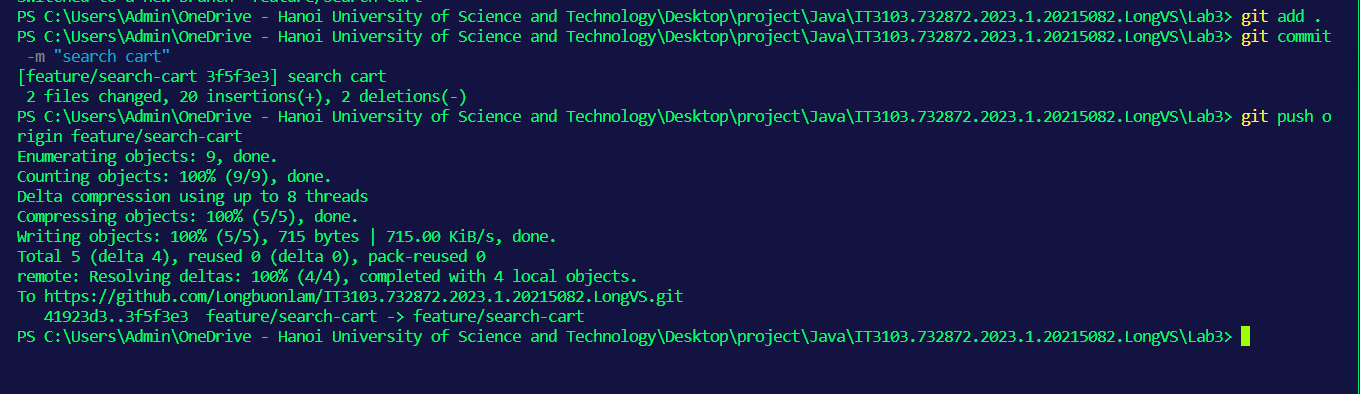
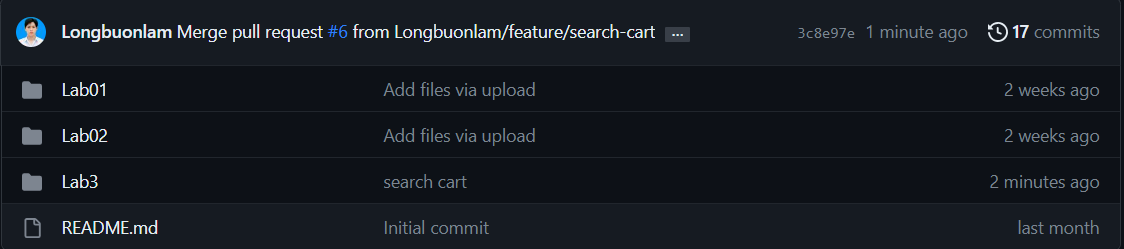


Figure : Result







# 7. Implement the Store class

Create a branch topic/store for the implementation of the class Store



Figure : Store class

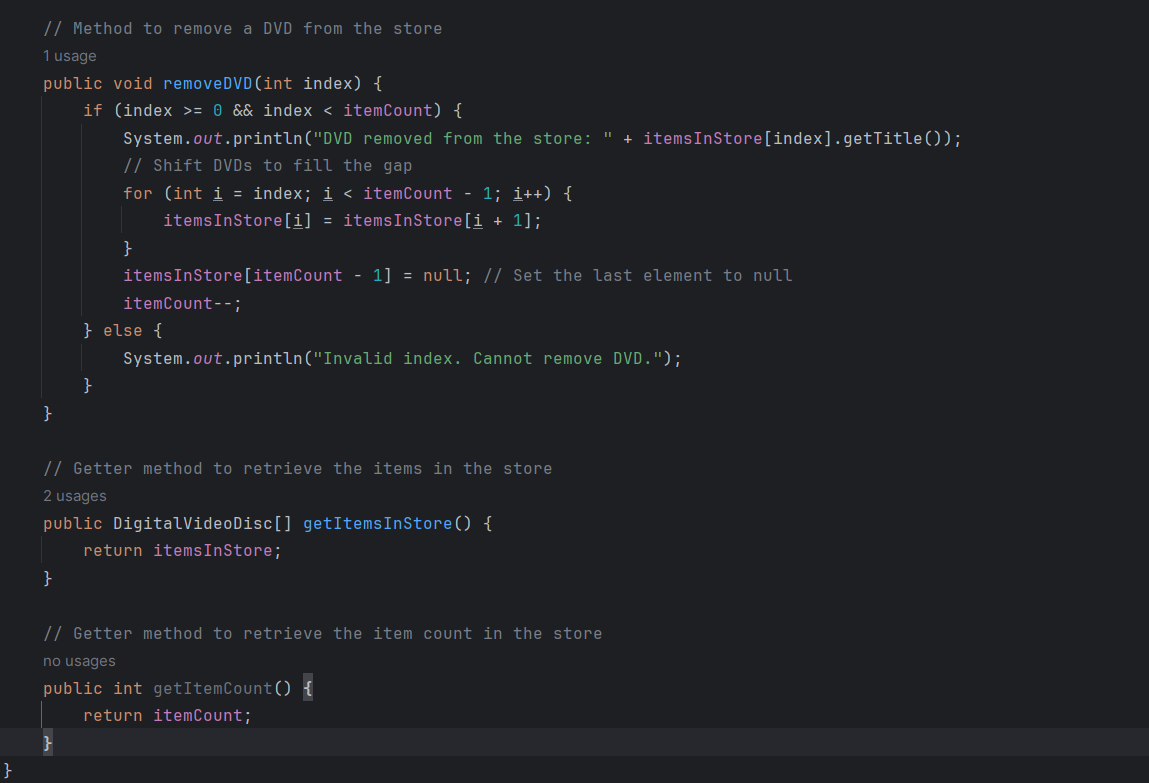




Figure : StoreTest class



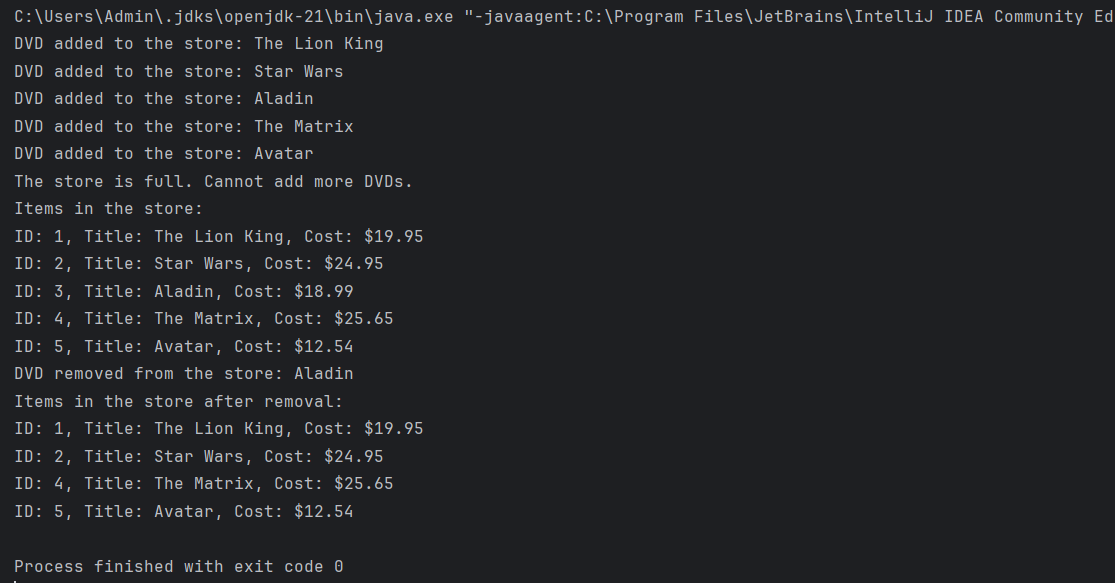
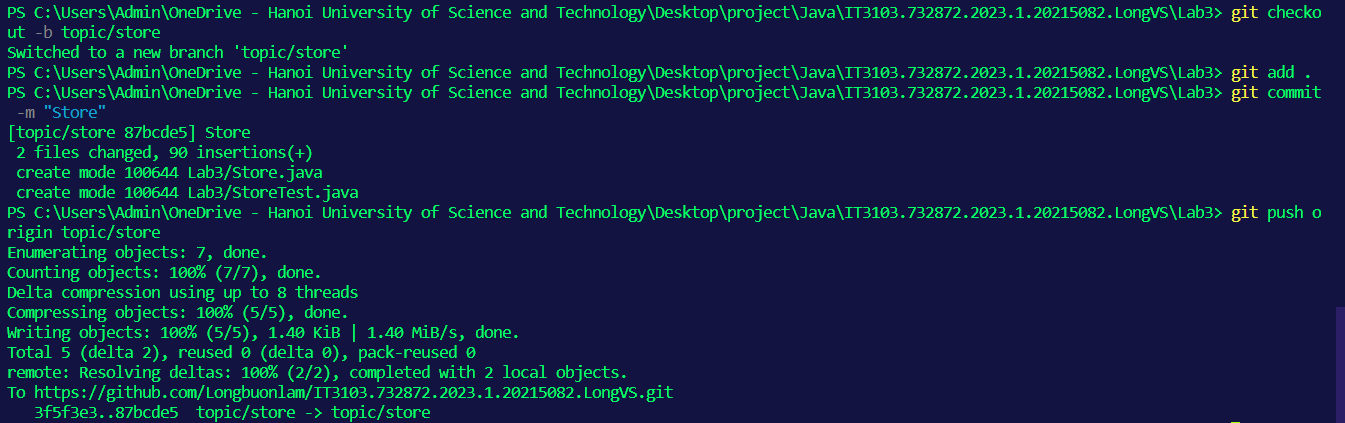
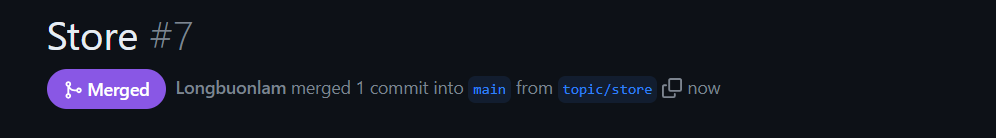
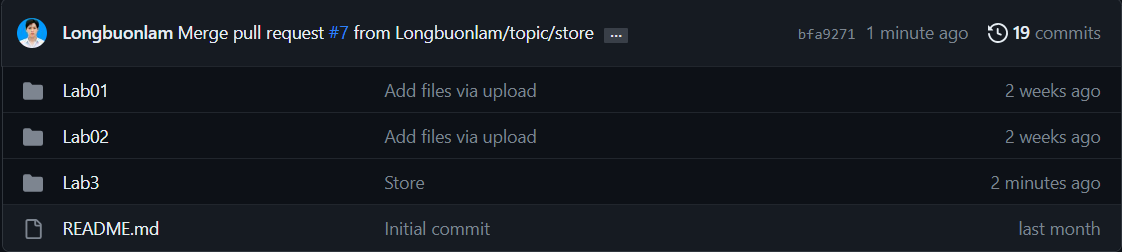


Figure : result

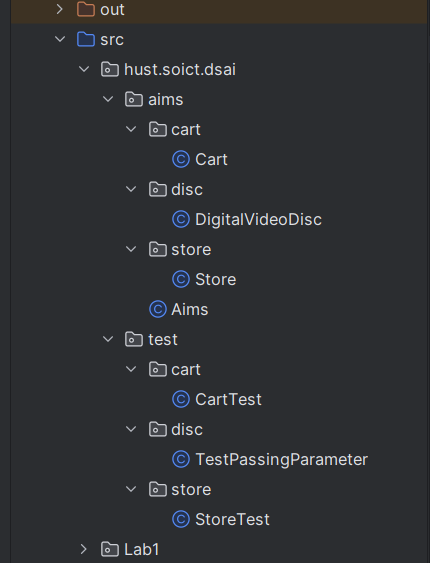


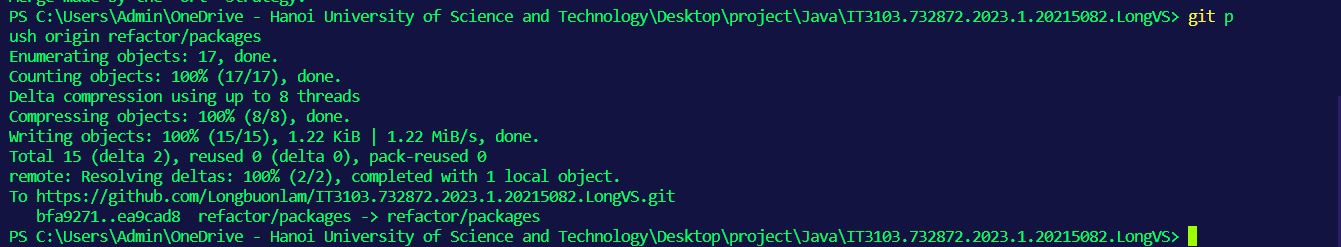


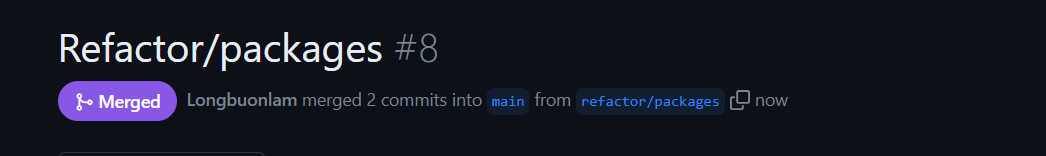


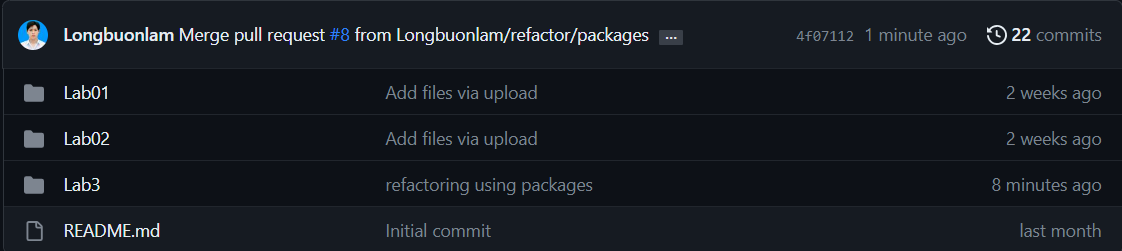
# 8. Re-organize your projects

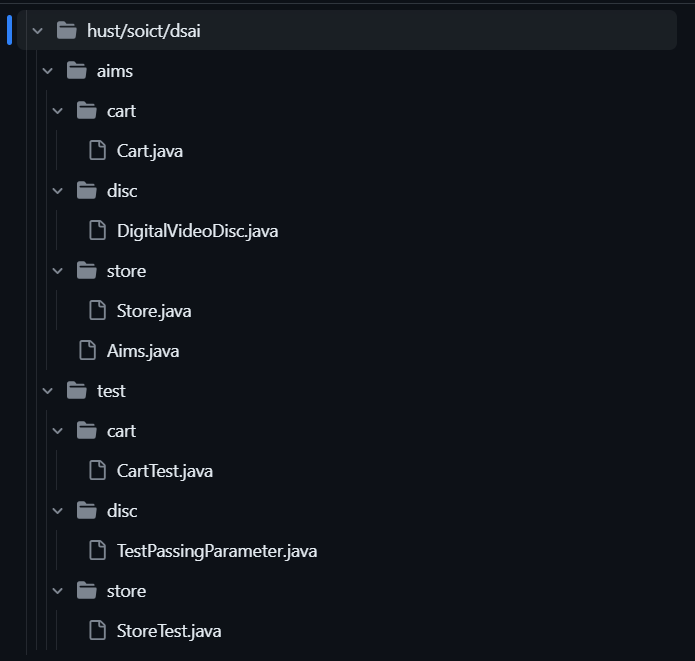
Create a branch refactor/packages for refactoring the projects in your repository using packages











# 9. String, StringBuilder and StringBuffer

Create a branch topic/memory-management-string for the String, StringBufer & StringBuilder exercise



Figure : ConcatenationInLoops class

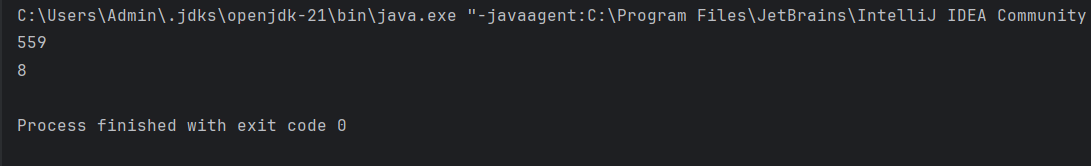


Figure : result

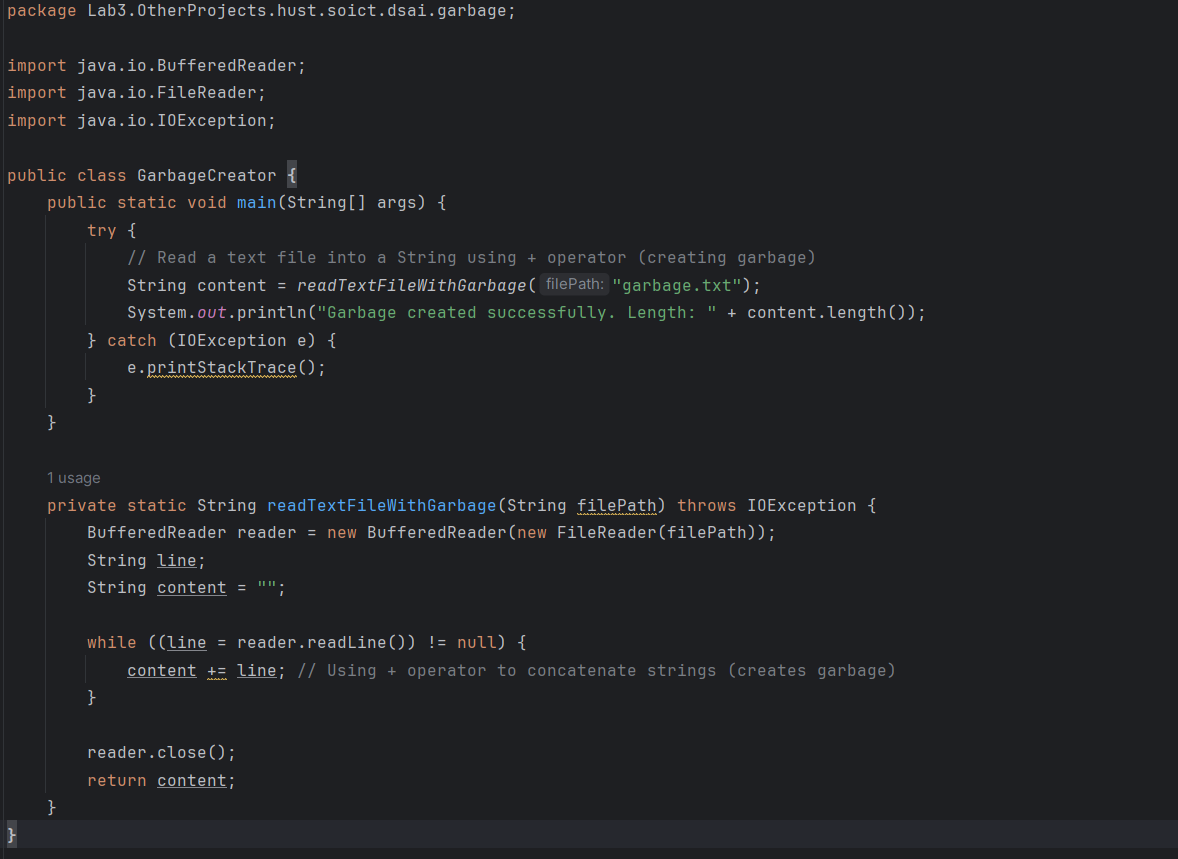


Figure : GarbageCreator class

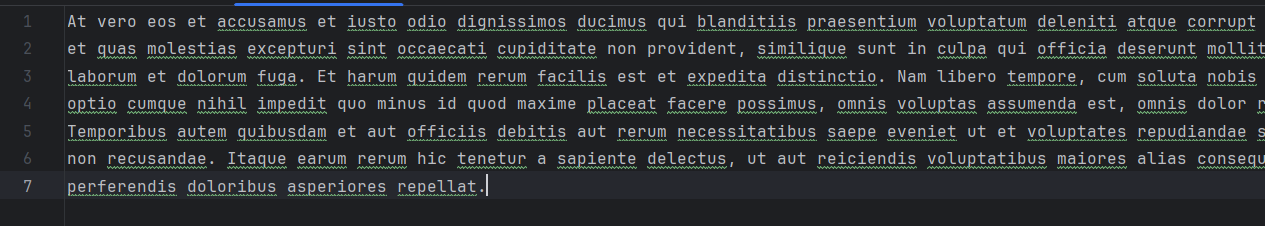


Figure : garbage.txt

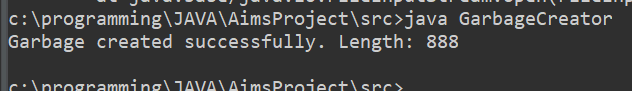


Figure : result

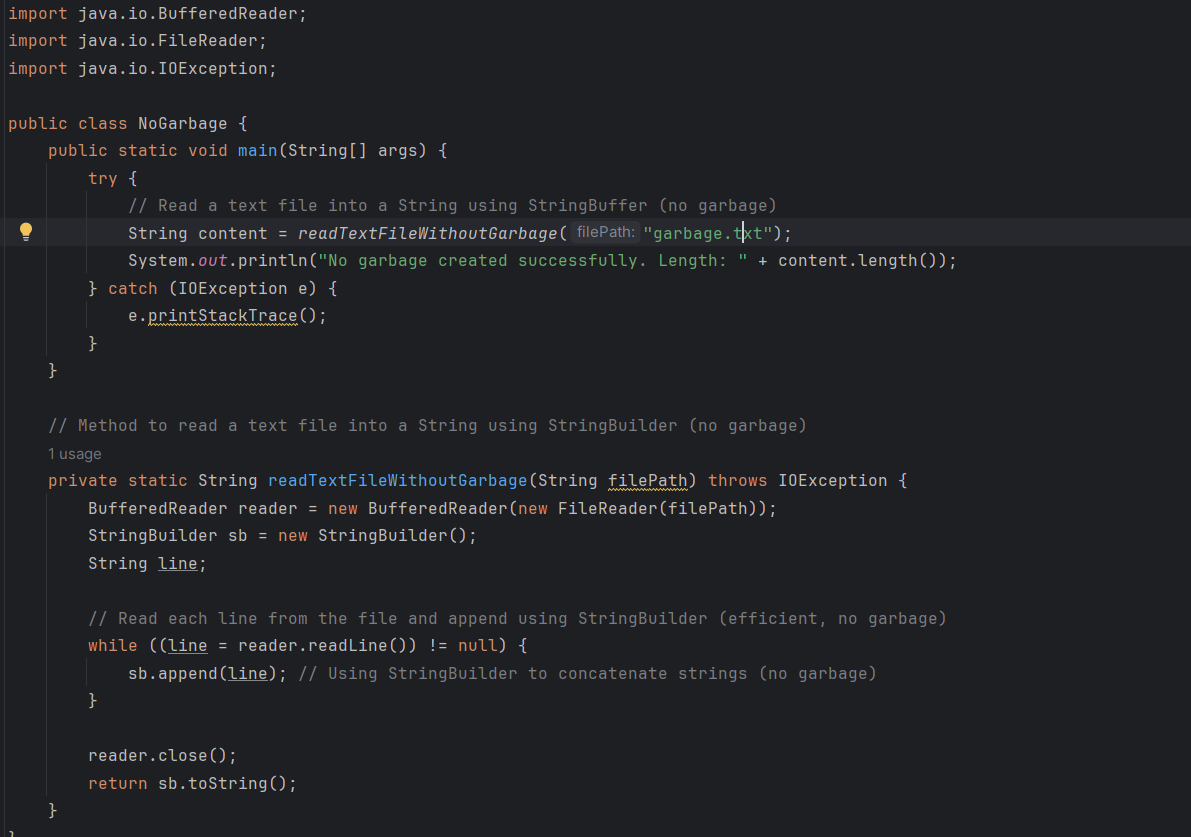


Figure : NoGarbage class

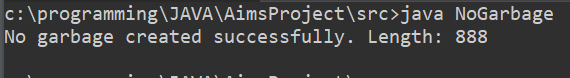


Figure : result



