BÁO CÁO THỰC HÀNH LAB 3 LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

# Mục lục nội dung

1. [Working with method overloading 3](#_bookmark0)
   1. [Overloading by differing types of parameter 3](#_bookmark1)
   2. [Overloading by differing the number of parameters 3](#_bookmark3)
2. [Passing parameter 4](#_bookmark5)

[Code: 4](#_bookmark6)

[Result: 5](#_bookmark9)

1. [Classifier Member and Instance Member 5](#_bookmark11)

[Code: 6](#_bookmark12)

[Result: 8](#_bookmark16)

1. [Open the Cart class 8](#_bookmark18)

[Code: 8](#_bookmark19)

[Result: 10](#_bookmark24)

1. [Implement the Store class 11](#_bookmark26)

[Code: 11](#_bookmark27)

[Result: 14](#_bookmark31)

1. [String, StringBuilder and StringBuffer 15](#_bookmark33)

[Code: 15](#_bookmark34)

[Result: 15](#_bookmark36)

1. [Answer the Question 15](#_bookmark38)
2. [Class Diagram 16](#_bookmark39)

Mục lục hình ảnh

[Figure 1 Method addDigitalVideoDisc(DigitalVideoDisc [] dvdList) 3](#_bookmark2)

[Figure 2 Method addDigitalVideoDisc(DigitalVideoDisc dvd1,DigitalVideoDisc dvd2) 3](#_bookmark4)

[Figure 3 Passing parameter code 4](#_bookmark7)

[Figure 4 Passing parameter code 5](#_bookmark8)

[Figure 5 Passing parameter result 5](#_bookmark10)

[Figure 6 Classifier Member and Instance Member Code 6](#_bookmark13)

[Figure 7 Classifier Member and Instance Member Code 6](#_bookmark14)

[Figure 8 Classifier Member and Instance Member Code 7](#_bookmark15)

[Figure 9 Classifier Member and Instance Member Result 8](#_bookmark17)

[Figure 10 Method in class Cart to print the list 8](#_bookmark20)

[Figure 11 Method in class Cart to search 9](#_bookmark21)

[Figure 12 Method in class DigitalVideoDisc to check title and print a dvd 10](#_bookmark22)

[Figure 13 Test code 10](#_bookmark23)

[Figure 14 Result Open the Cart class 11](#_bookmark25)

[Figure 15 Code class Store 12](#_bookmark28)

[Figure 16 Code class Store 13](#_bookmark29)

[Figure 17 Code test class Store 14](#_bookmark30)

[Figure 18 Result Implement the Store class 14](#_bookmark32)

[Figure 19 Code ConcatenationInLoops 15](#_bookmark35)

[Figure 20 Result ConcatenationInLoops 15](#_bookmark37)

# Working with method overloading

### Overloading by differing types of parameter

### 

Figure 1 Method addDigitalVideoDisc(DigitalVideoDisc [] dvdList)

### Overloading by differing the number of parameters

### 

Figure 2 Method addDigitalVideoDisc(DigitalVideoDisc dvd1,DigitalVideoDisc dvd2)

# Passing parameter

## Code:

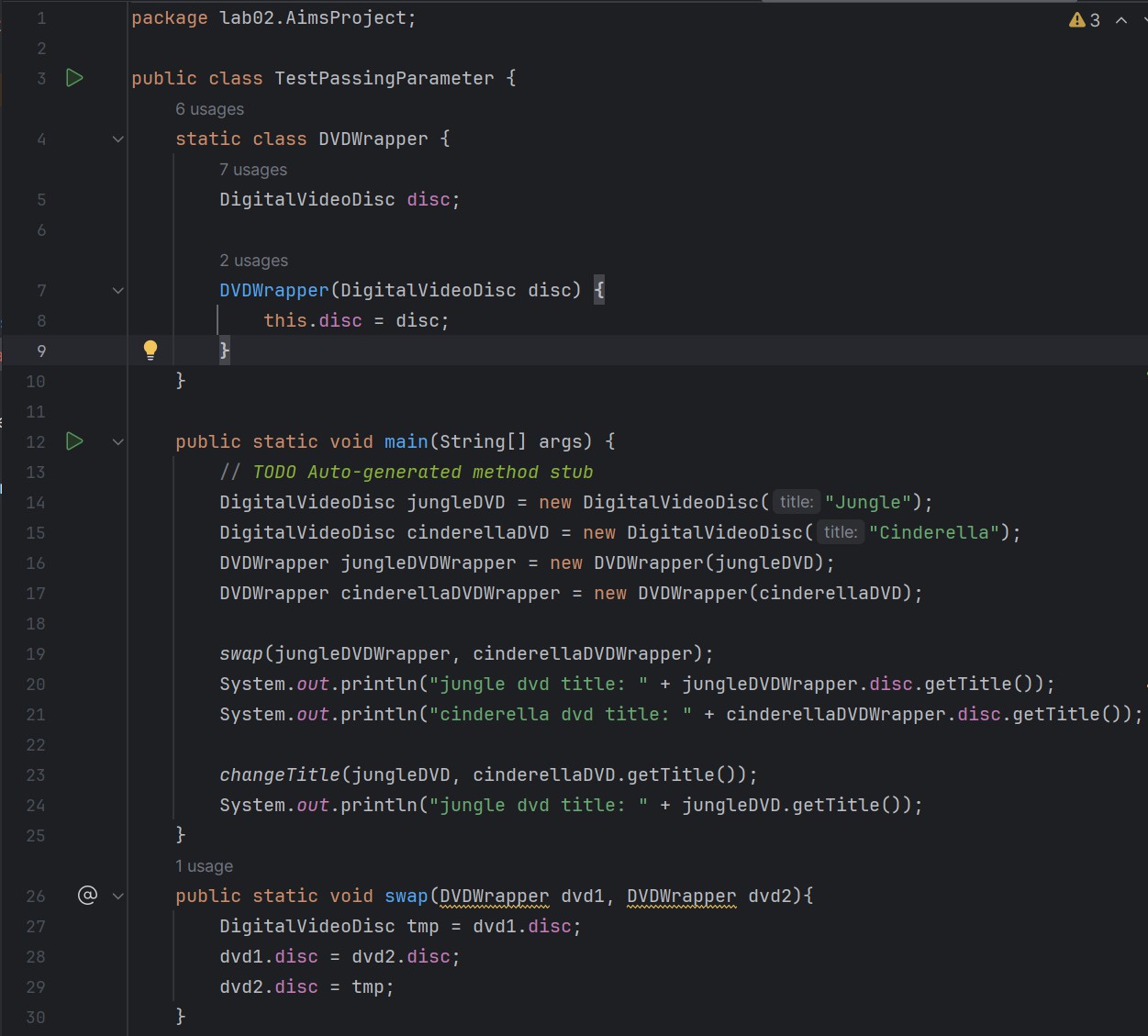


Figure 3 Passing parameter code

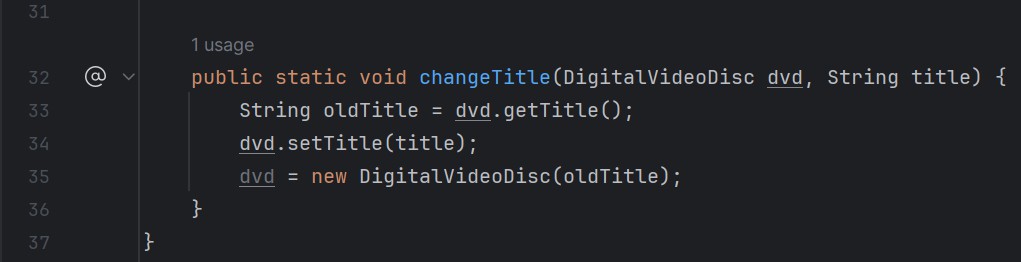


Figure 4 Passing parameter code

## Result:

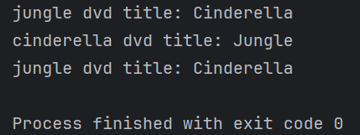
****

Figure 5 Passing parameter result

# Classifier Member and Instance Member

## Code:



Figure 6 Classifier Member and Instance Member Code

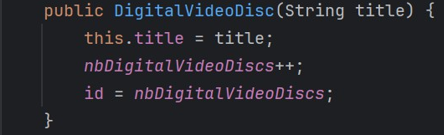


Figure 7 Classifier Member and Instance Member Code



Figure 8 Classifier Member and Instance Member Code

## Result:

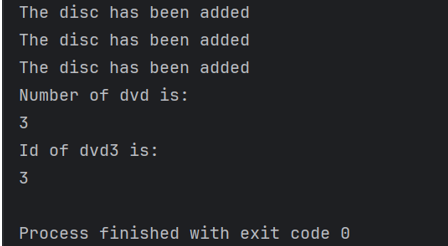
****

Figure 9 Classifier Member and Instance Member Result

# Open the Cart class

## Code:

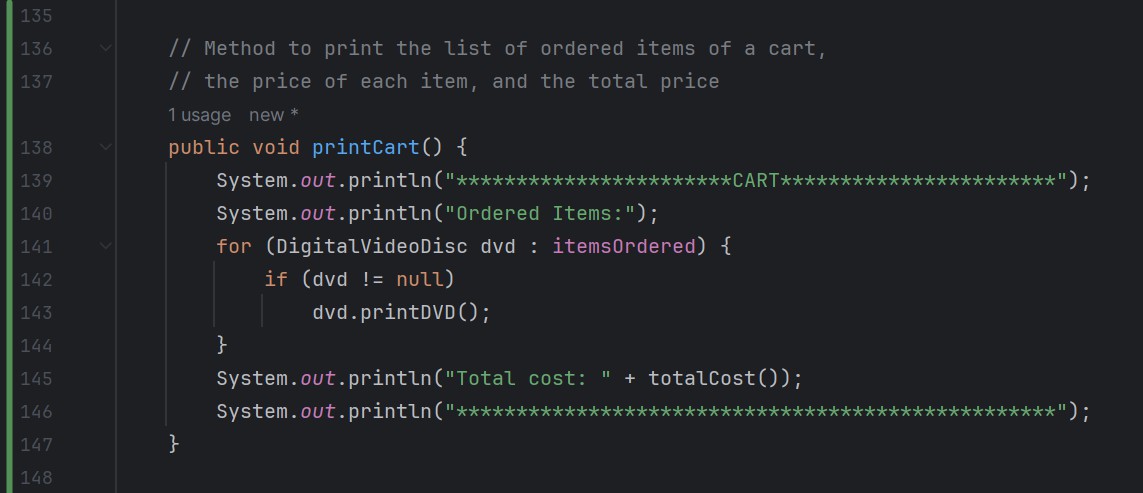


Figure 10 Method in class Cart to print the list

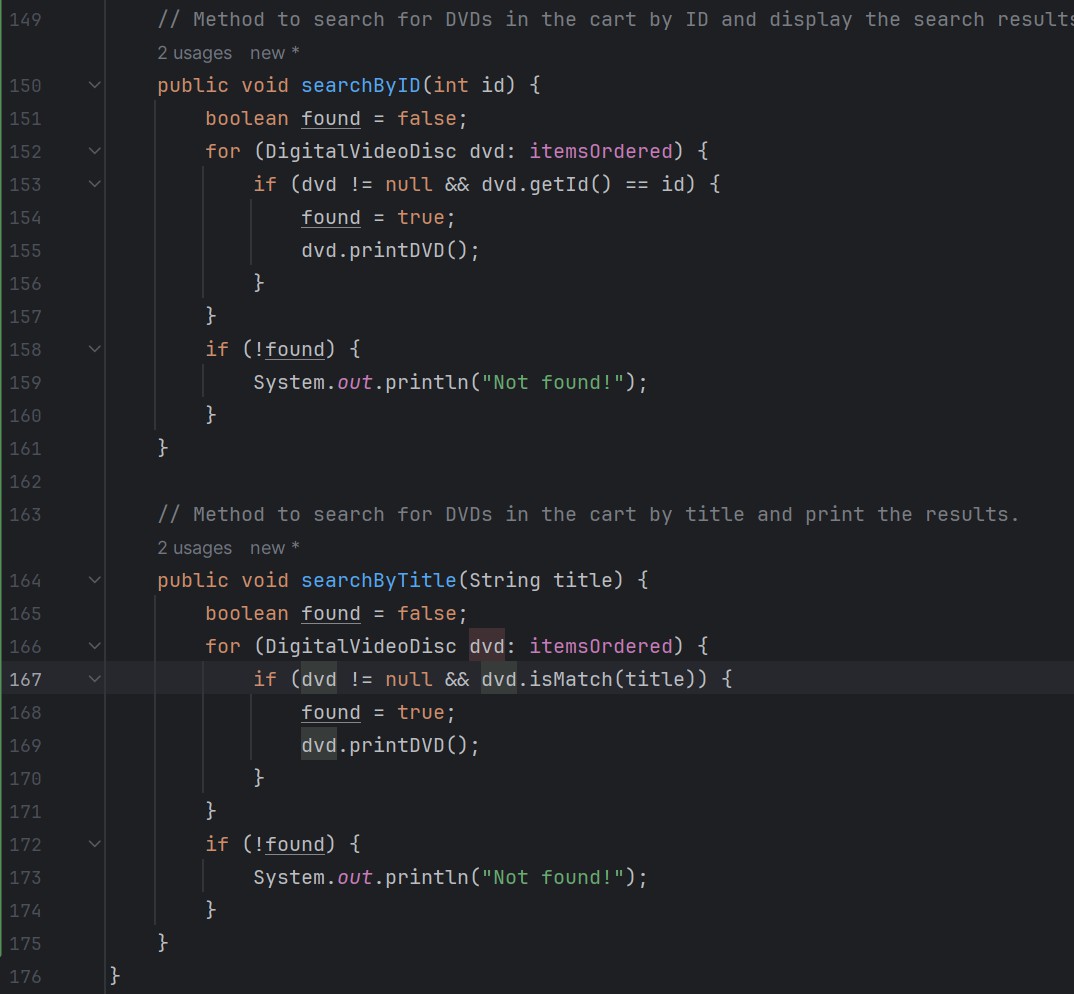


Figure 11 Method in class Cart to search

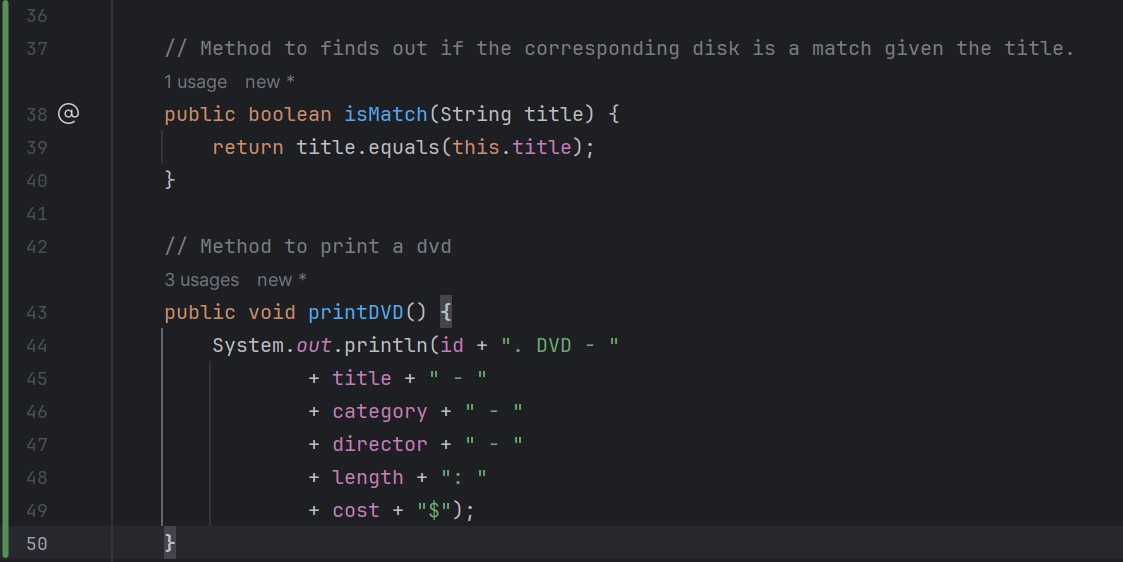


Figure 12 Method in class DigitalVideoDisc to check title and print a dvd

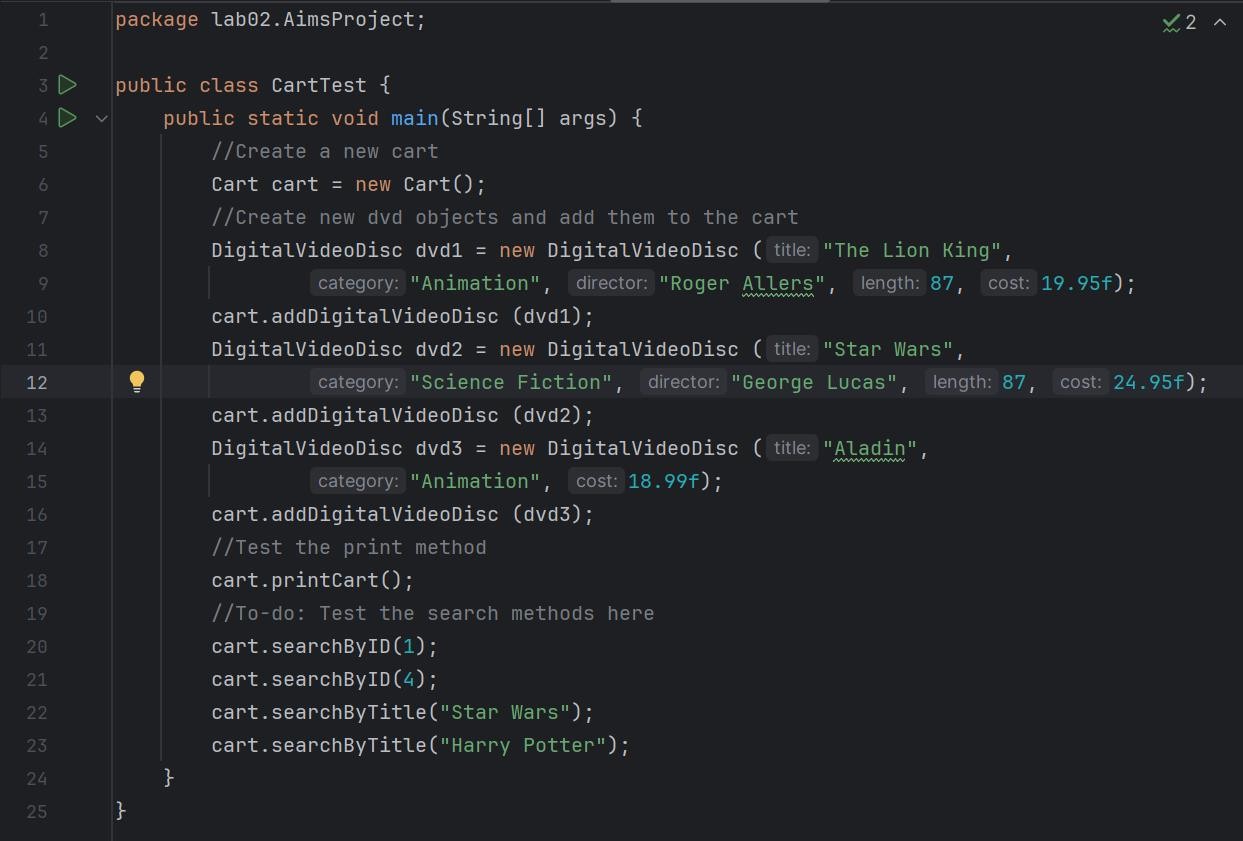


Figure 13 Test code

## Result:

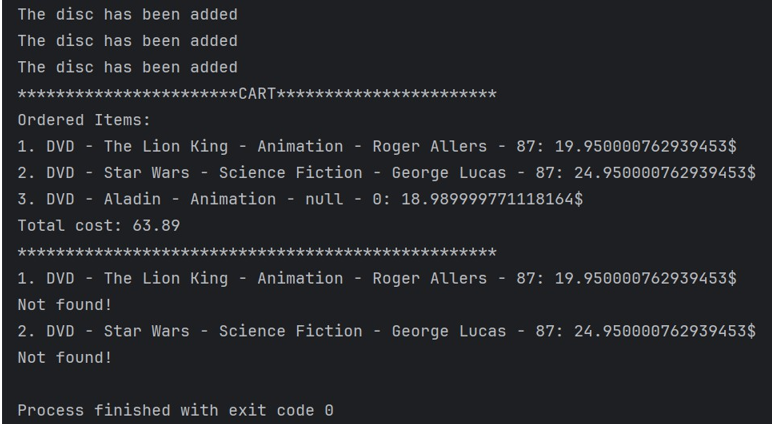


Figure 14 Result Open the Cart class

# Implement the Store class

## Code:



Figure 15 Code class Store

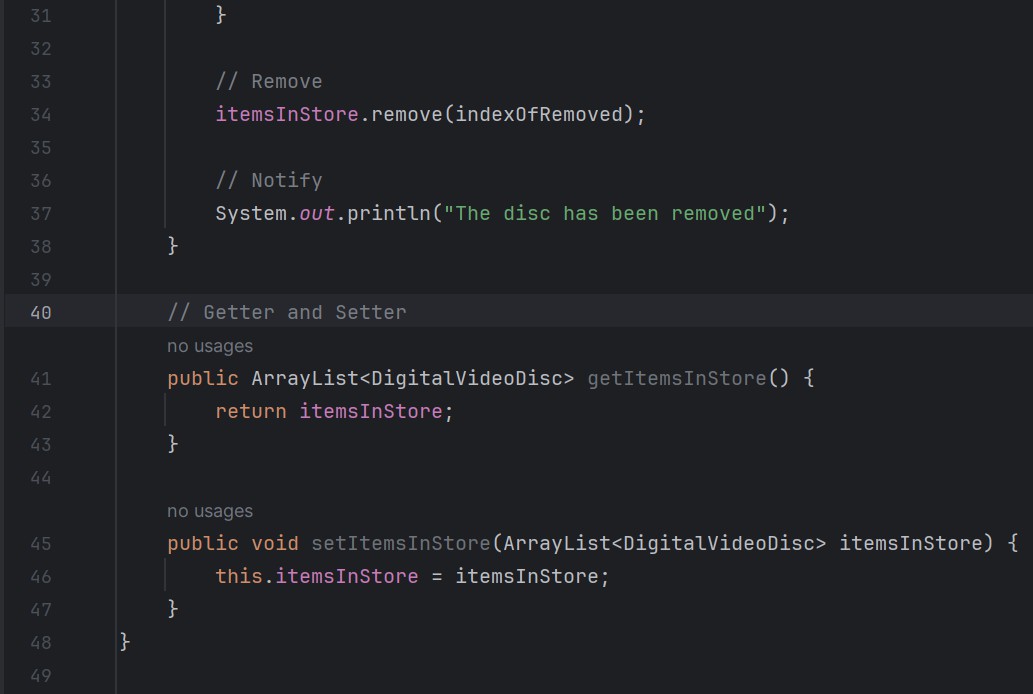


Figure 16 Code class Store

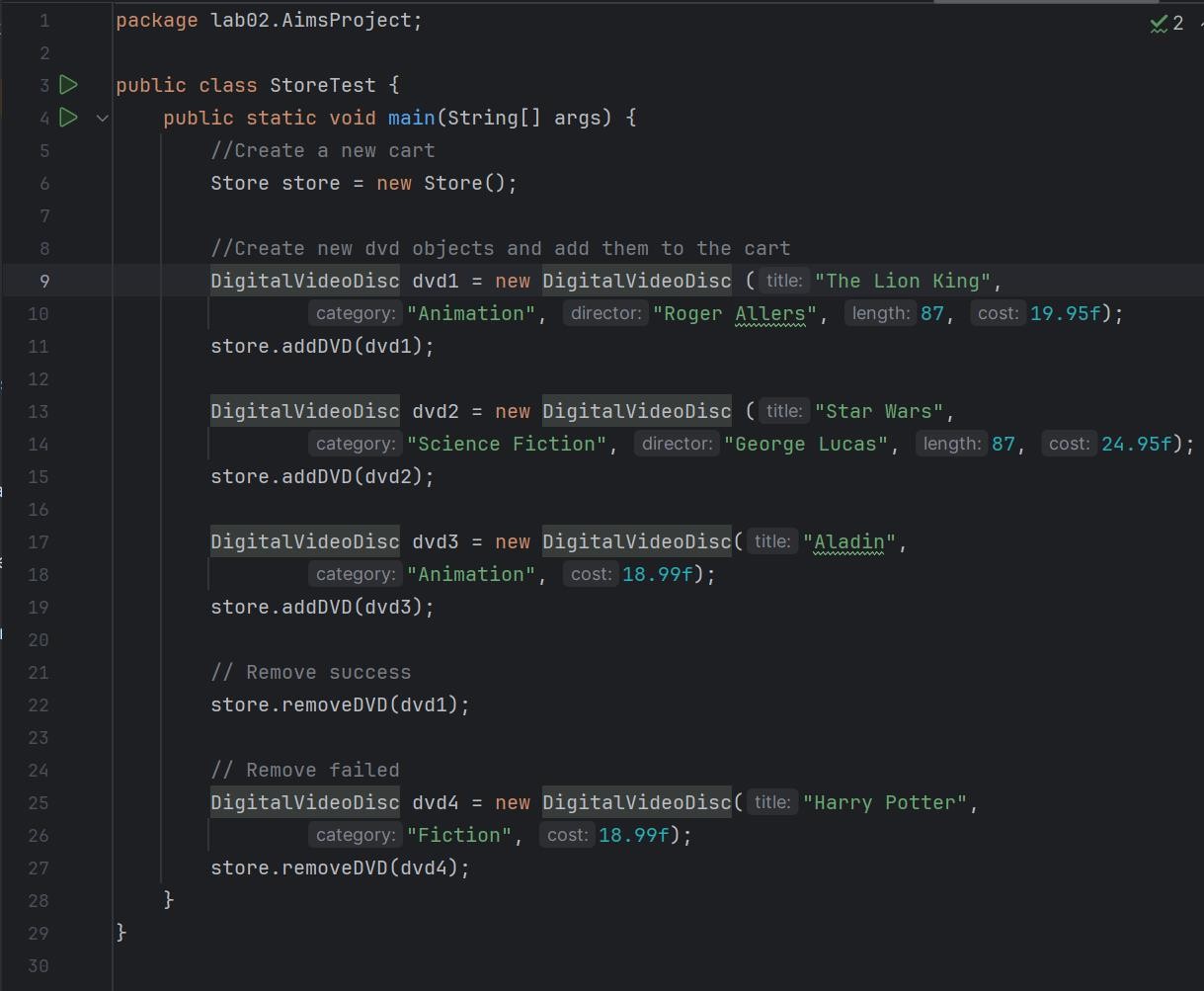


Figure 17 Code test class Store

## Result:

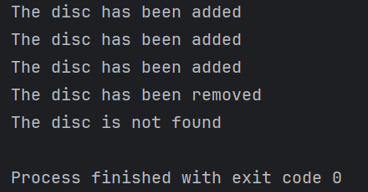
****

Figure 18 Result Implement the Store class

# String, StringBuilder and StringBuffer

## Code:

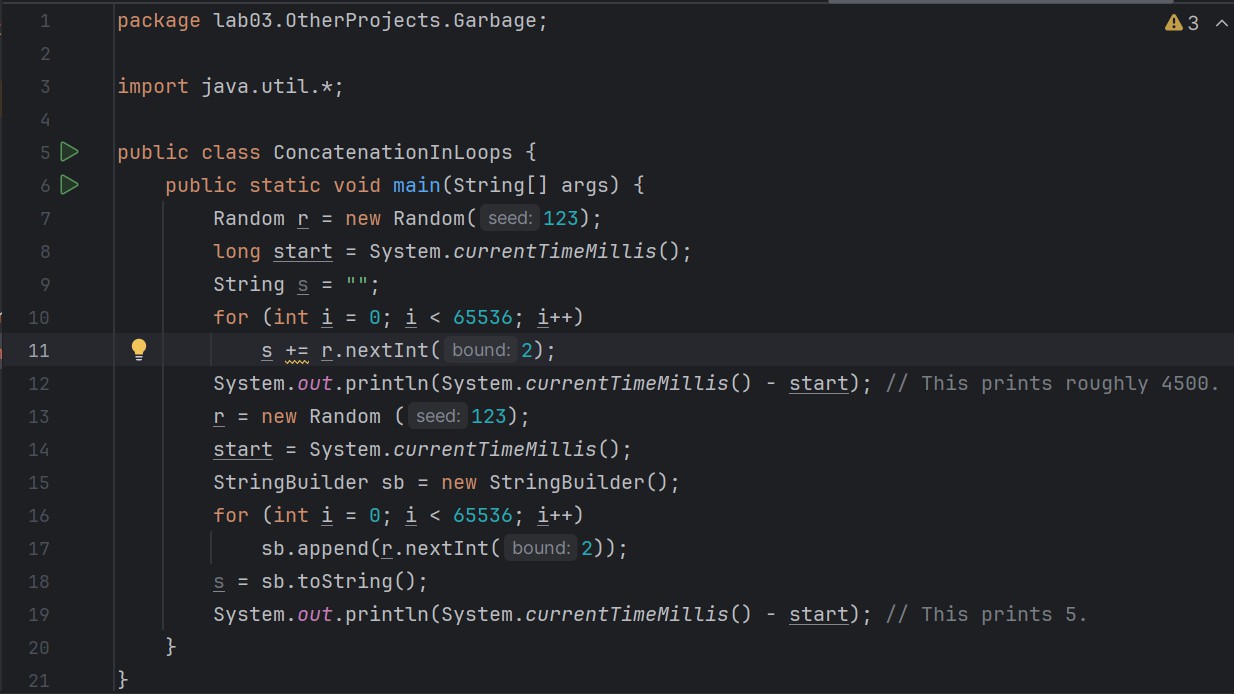


Figure 19 Code ConcatenationInLoops

## Result:

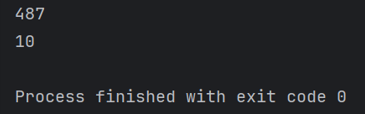
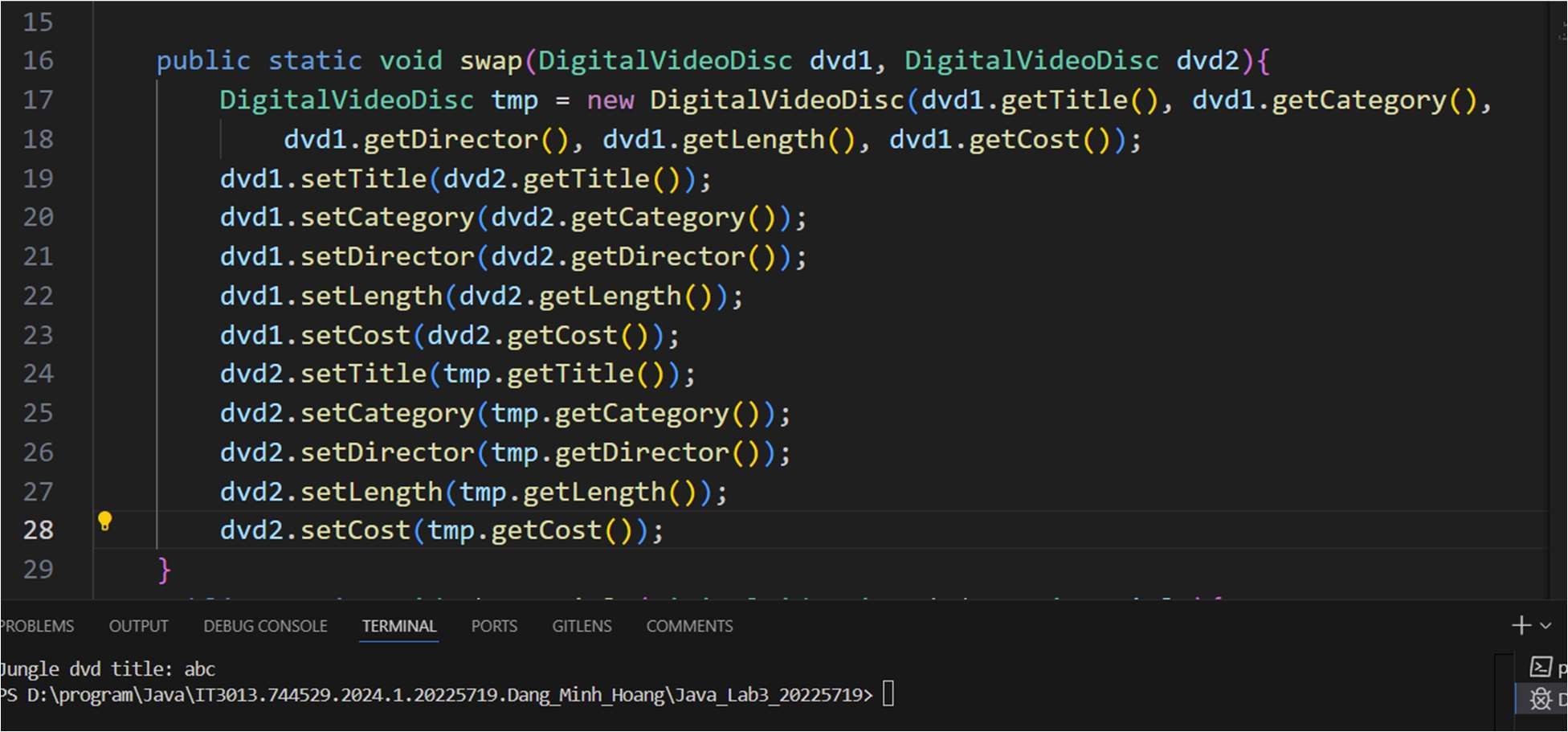
****

Figure 20 Result ConcatenationInLoops

# Answer the Question

# Truyền tham số

## Java là ngôn ngữ lập trình truyền bằng tham chiếu hay truyền bằng tham trị?

* + - Trả lời: Java là ngôn ngữ lập trình truyền bằng tham trị. Tham trị (pass-by-value) là một khái niệm trong lập trình, chỉ cách mà tham số được truyền vào một phương thức hoặc hàm. Trong cơ chế truyền tham trị, giá trị thực sự của tham số được sao chép và truyền cho phương thức, tức là phương thức làm việc với một bản sao của tham số đó.
  1. Sau khi gọi hàm **swap(jungleDVD, cinderellaDVD)**, tại sao tiêu đề của hai đối tượng này vẫn giữ nguyên?
     + Trả lời: Điều này xảy ra vì trong Java, các tham số được truyền vào hàm theo **tham trị** (pass by value), chứ không phải **tham chiếu** (pass by reference). Khi gọi swap(jungleDVD, cinderellaDVD), hai biến object1 và object2 trong hàm swap chỉ là **bản sao** của jungleDVD và cinderellaDVD. Chúng trỏ tới cùng đối tượng, nhưng là các tham chiếu riêng biệt và **không ảnh hưởng** đến các biến gốc bên ngoài. Vì vậy, việc hoán đổi object1 và object2 trong hàm swap không làm thay đổi giá trị của jungleDVD và cinderellaDVD bên ngoài hàm.
  2. Vì sao sau khi gọi *changeTitle(jungleDVD, cinderellaDVD.getTitle())*, biến *title* của *jungleDVD* thay đổi?
     + Trả lời: phương thức *changeTitle()* gọi đến phương thức *setTitle()* của biến *dvd* được truyền vào, lúc này *dvd* và *jungleDVD* đang trỏ đến cùng một đối tượng, điều này khiến cho biến *title* của đối tượng gốc bị thay đổi.
  3. Sửa lại hàm *swap()*

*Hình 11: Hàm swap() sau khi sửa lại*

# Class Diagram

