OOP_Lab03

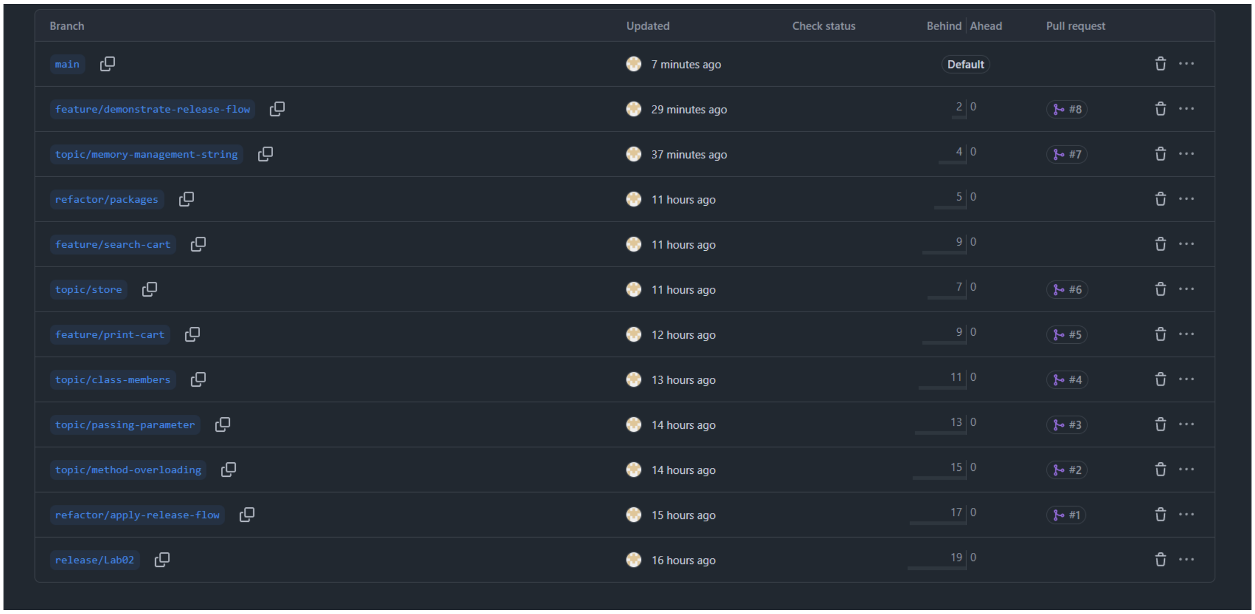
Ho và tên: Phan Đức Duy

MSSV: 20225831

Mã lớp: 744520

Contents

1. Branch your repository

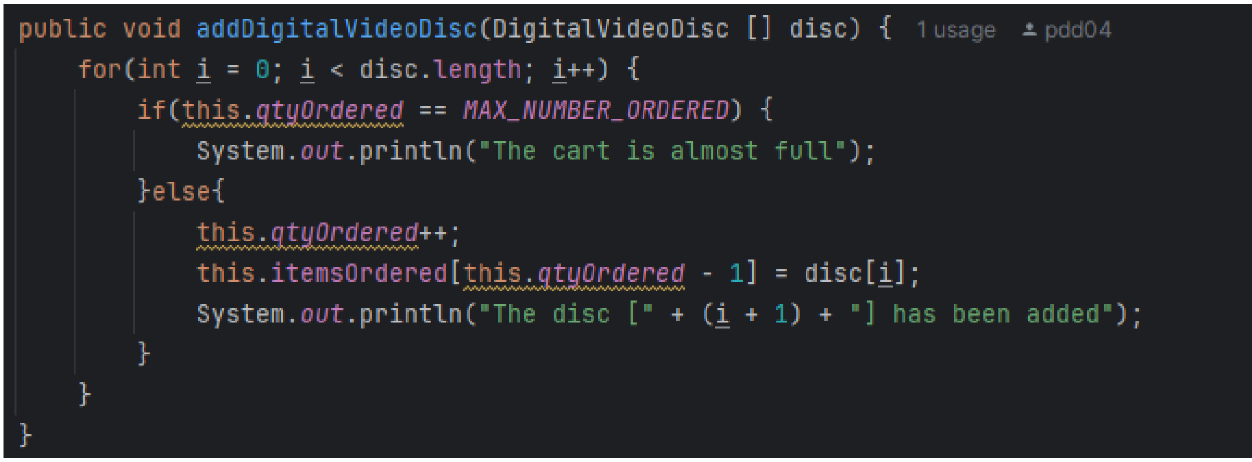


Branch	Updated	Check status	Behind	Ahead	Pull request
main	7 minutes ago		Default		
feature/demonstrate-release-flow	29 minutes ago		2	0	#8
topic/memory-management-string	37 minutes ago		4	0	#7
refactor/packages	11 hours ago		5	0	
feature/search-cart	11 hours ago		9	0	
topic/store	11 hours ago		7	0	#6
feature/print-cart	12 hours ago		9	0	#5
topic/class-members	13 hours ago		11	0	#4
topic/passing-parameter	14 hours ago		13	0	#3
topic/method-overloading	14 hours ago		15	0	#2
refactor/apply-release-flow	15 hours ago		17	0	#1
release/Lab02	16 hours ago		19	0	

2. Working with method overloading

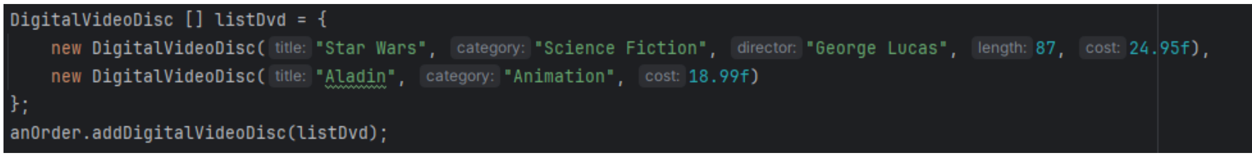
2.1. Overloading by differing types of parameter

Class cart.java:



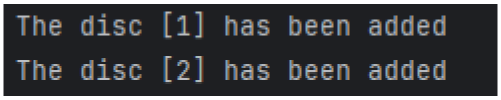
```java
public void addDigitalVideoDisc(DigitalVideoDisc [] disc) {
    for(int i = 0; i < disc.length; i++) {
        if(this.qtyOrdered == MAX_NUMBER_ORDERED) {
            System.out.println("The cart is almost full");
        }else{
            this.qtyOrdered++;
            this.itemsOrdered[this.qtyOrdered - 1] = disc[i];
            System.out.println("The disc [" + (i + 1) + "] has been added");
        }
    }
}
```

Class Aims.java:



```
DigitalVideoDisc [] listDvd = {
    new DigitalVideoDisc( title: "Star Wars",  category: "Science Fiction",  director: "George Lucas",  length: 87,  cost: 24.95f),
    new DigitalVideoDisc( title: "Aladin",  category: "Animation",  cost: 18.99f)
};
anOrder.addDigitalVideoDisc(listDvd);
```

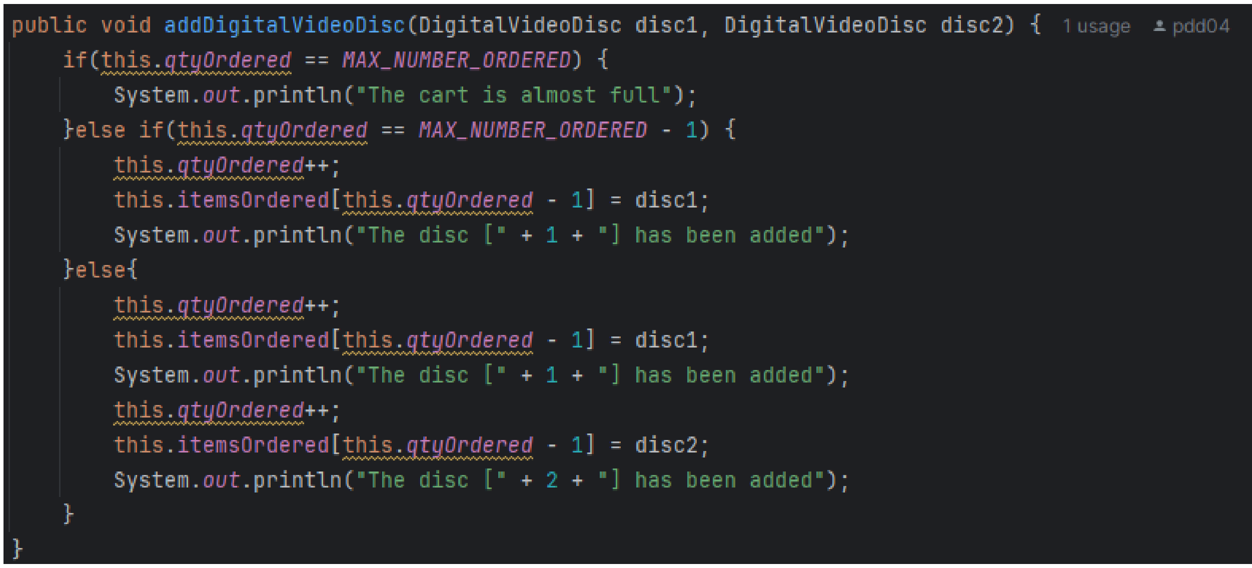
Kết quả:



```
The disc [1] has been added
The disc [2] has been added
```

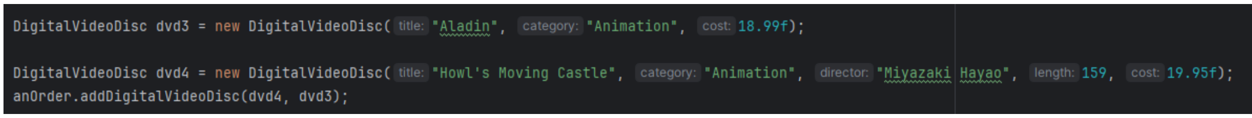
2.2. Overloading by differing the number of parameters

Class cart.java:



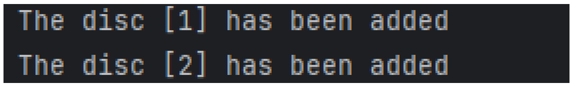
```
public void addDigitalVideoDisc(DigitalVideoDisc disc1, DigitalVideoDisc disc2) {  1 usage  pdd04
    if(this.qtyOrdered == MAX_NUMBER_ORDERED) {
        System.out.println("The cart is almost full");
    }else if(this.qtyOrdered == MAX_NUMBER_ORDERED - 1) {
        this.qtyOrdered++;
        this.itemsOrdered[this.qtyOrdered - 1] = disc1;
        System.out.println("The disc [" + 1 + "] has been added");
    }else{
        this.qtyOrdered++;
        this.itemsOrdered[this.qtyOrdered - 1] = disc1;
        System.out.println("The disc [" + 1 + "] has been added");
        this.qtyOrdered++;
        this.itemsOrdered[this.qtyOrdered - 1] = disc2;
        System.out.println("The disc [" + 2 + "] has been added");
    }
}
```

Class Aims.java:



```
DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin",  category: "Animation",  cost: 18.99f);

DigitalVideoDisc dvd4 = new DigitalVideoDisc( title: "Howl's Moving Castle",  category: "Animation",  director: "Miyazaki Hayao",  length: 159,  cost: 19.95f);
anOrder.addDigitalVideoDisc(dvd4, dvd3);
```

Kết quả:

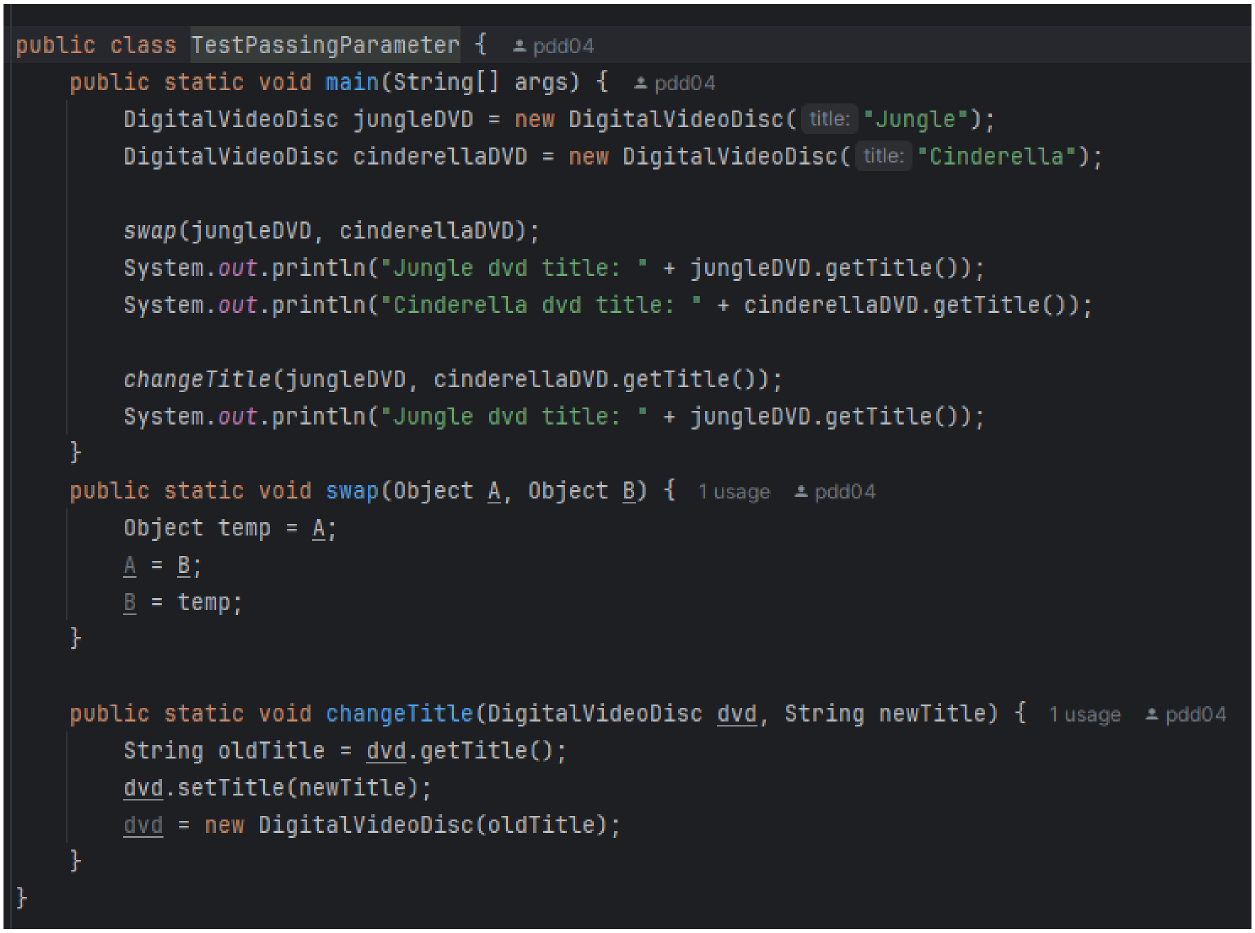


```
The disc [1] has been added
The disc [2] has been added
```

3. Passing parameter

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

Trả lời: java là ngôn ngữ Pass by value



```java
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 A, Object B) {
        Object temp = A;
        A = B;
        B = temp;
    }

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

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

Trả lời: khi ta gọi hàm **swap(jungleDVD, cinderellaDVD)** giá trị của jungleDVD và cinderellaDVD sẽ được sao chép và chuyền vào 2 tham số A và B.

Mọi thay đổi nào được thực hiện với A và B chỉ làm ảnh hưởng đến chính nó chứ không làm thay đổi jungleDVD và cinderellaDVD

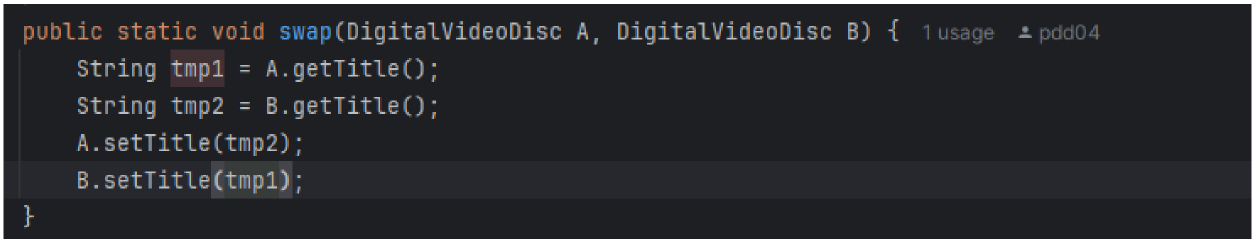
Question: After the call of changeTitle(jungleDVD, cinderellaDVD.getTitle()) why is the title of the JungleDVD changed?

Trả lời: khi ta truyền DigitalVideoDisc dvd vào hàm nghĩa là ta truyền đối tượng được biến đó tham chiếu tới.

Vậy nên khi sử dụng method (setter) của chính đối tượng được tham chiếu thì dữ liệu của nó cũng sẽ được thay đổi.

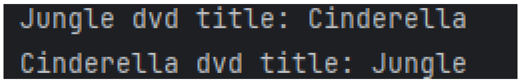
Please write a swap() method that can correctly swap the two objects.

Code:



```
public static void swap(DigitalVideoDisc A, DigitalVideoDisc B) {
    String tmp1 = A.getTitle();
    String tmp2 = B.getTitle();
    A.setTitle(tmp2);
    B.setTitle(tmp1);
}
```

Kết quả:



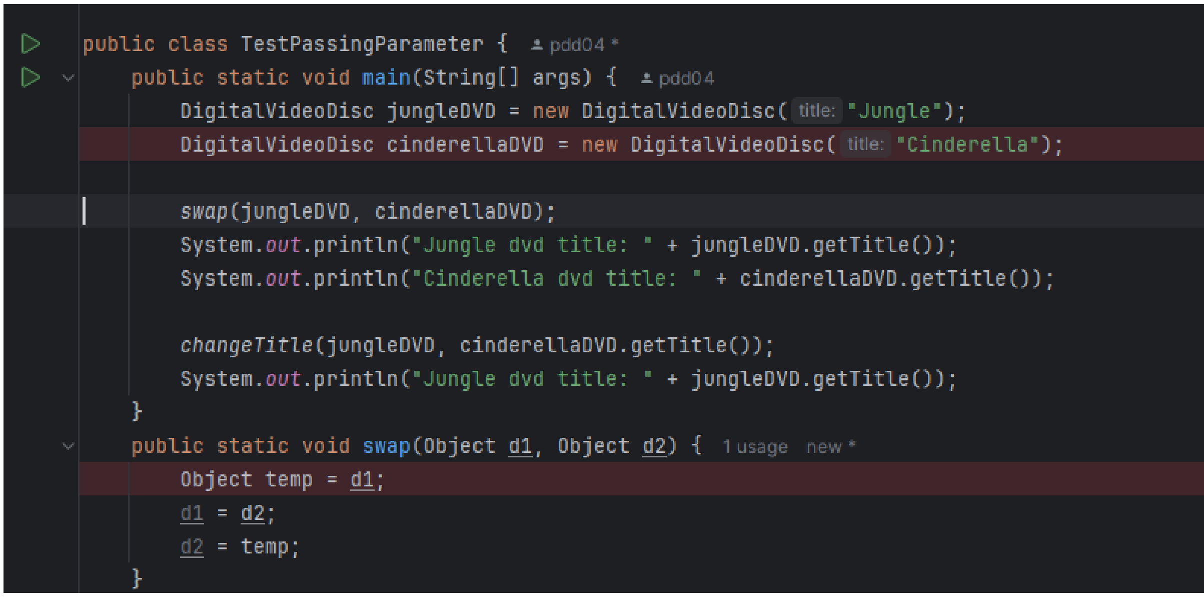
```
Jungle dvd title: Cinderella
Cinderella dvd title: Jungle
```

4. Use debug run

4.1. Debugging Java in Eclipse

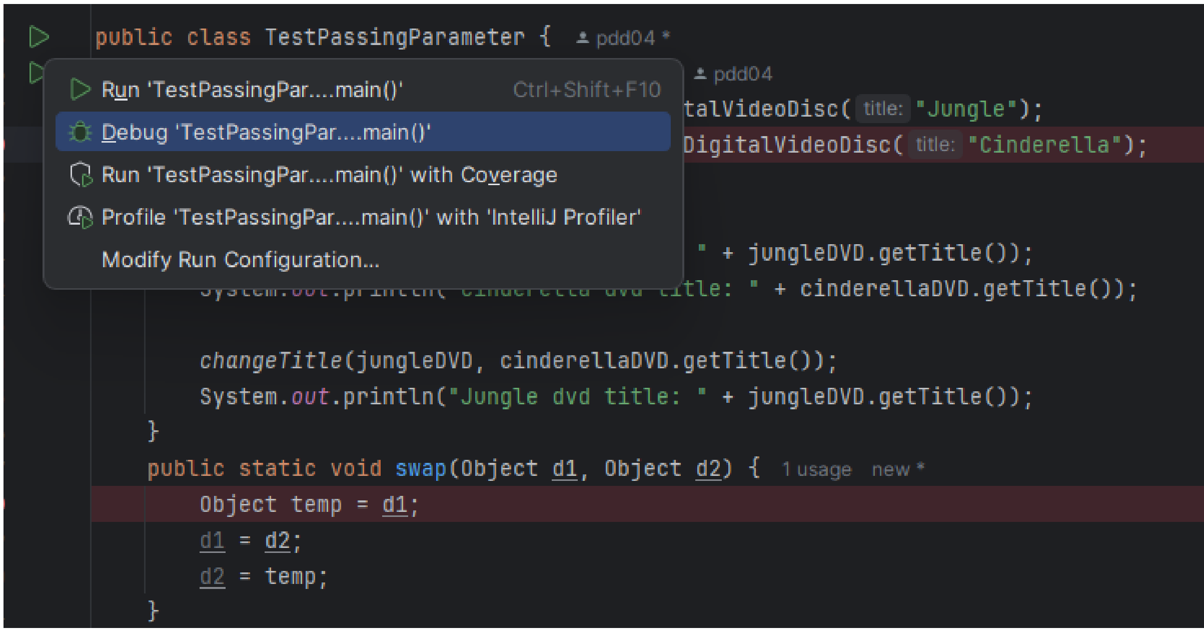
4.2. Example of debug run for the swap method of TestPassingParameter

4.2.1. Setting, deleting & deactivate breakpoints



```
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 d1, Object d2) {
        Object temp = d1;
        d1 = d2;
        d2 = temp;
    }
```

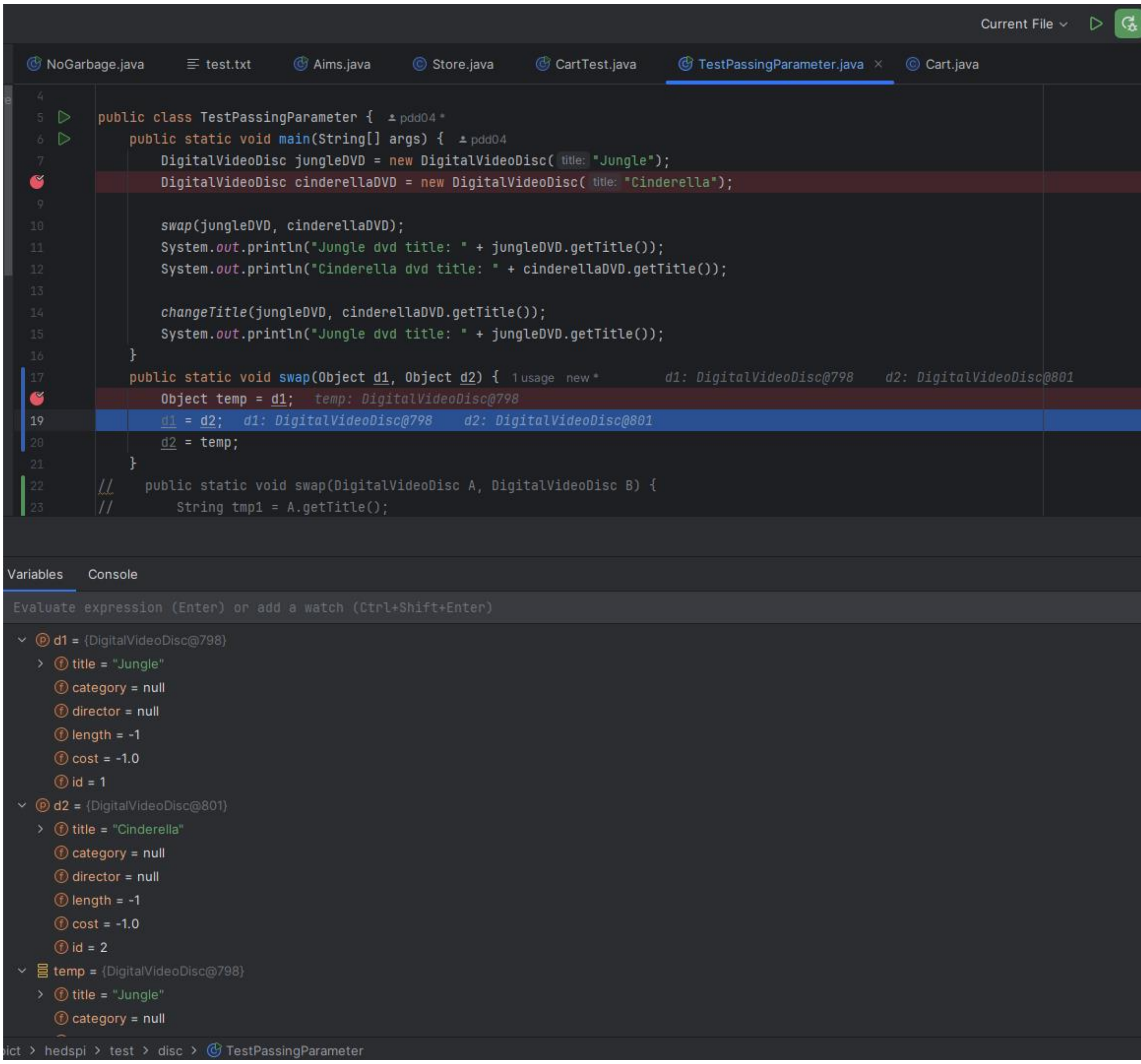
4.2.2. Run in Debug mode



Run 'TestPassingPar....main()'   Ctrl+Shift+F10
Debug 'TestPassingPar....main()'
Run 'TestPassingPar....main()' with Coverage
Profile 'TestPassingPar....main()' with 'IntelliJ Profiler'
Modify Run Configuration...
pdd04 *
pdd04
1 usage   new *

```
public class TestPassingParameter {
                                          talVideoDisc( title: "Jungle");
                                          DigitalVideoDisc( title: "Cinderella");

                                          " + 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 d1, Object d2) {
        Object temp = d1;
        d1 = d2;
        d2 = temp;
    }
```

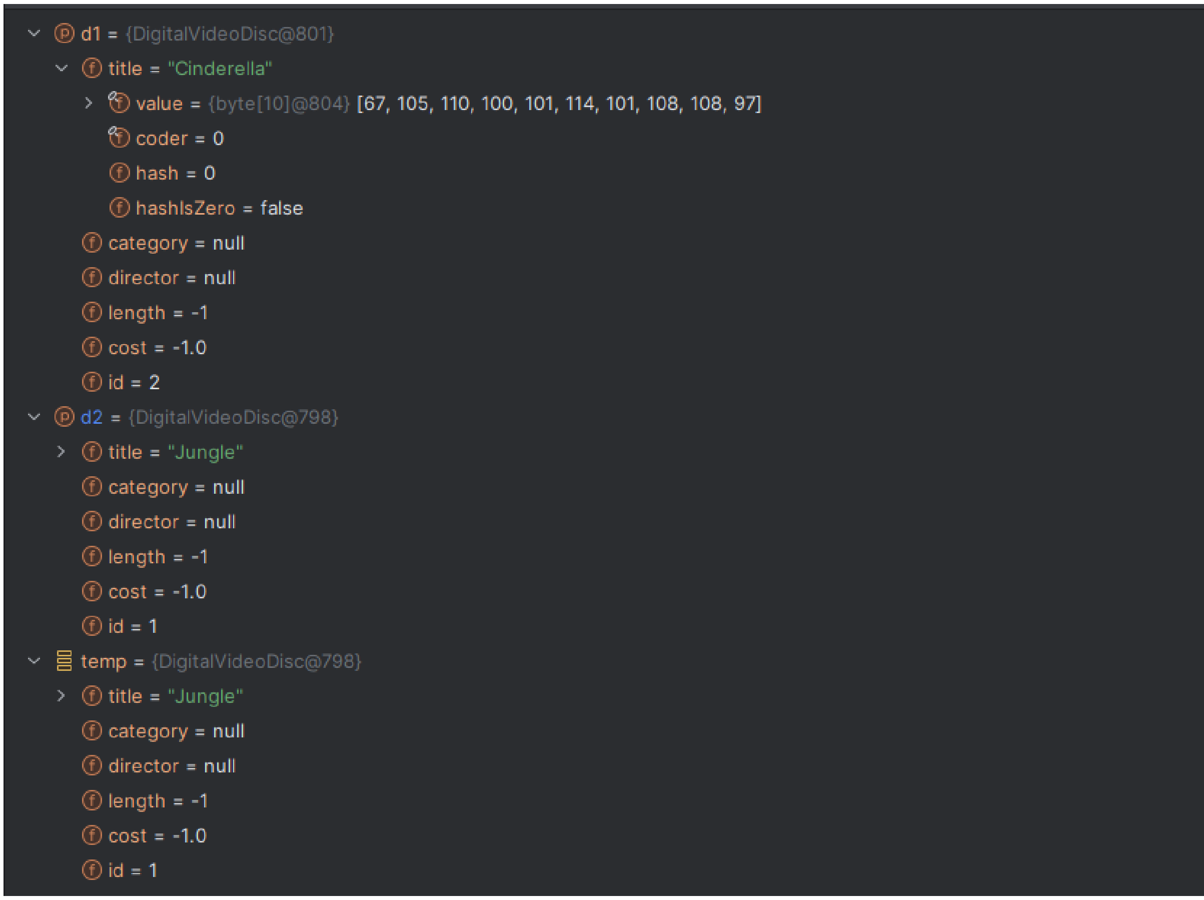
4.2.3. Step Into, Step Over, Step Return, Resume


Current File
NoGarbage.java
test.txt
Aims.java
Store.java
CartTest.java
TestPassingParameter.java
Cart.java
Variables
Console
Evaluate expression (Enter) or add a watch (Ctrl+Shift+Enter)
ict > hedspi > test > disc > TestPassingParameter

```
public class TestPassingParameter {  pdd04 *
    public static void main(String[] args) {  pdd04
        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("Cinderella dvd title: " + cinderellaDVD.getTitle());

        changeTitle(jungleDVD, cinderellaDVD.getTitle());
        System.out.println("Jungle dvd title: " + jungleDVD.getTitle());
    }
    public static void swap(Object d1, Object d2) {  1 usage  new *     d1: DigitalVideoDisc@798    d2: DigitalVideoDisc@801
        Object temp = d1;   temp: DigitalVideoDisc@798
        d1 = d2;   d1: DigitalVideoDisc@798    d2: DigitalVideoDisc@801
        d2 = temp;
    }
//    public static void swap(DigitalVideoDisc A, DigitalVideoDisc B) {
//        String tmp1 = A.getTitle();
```

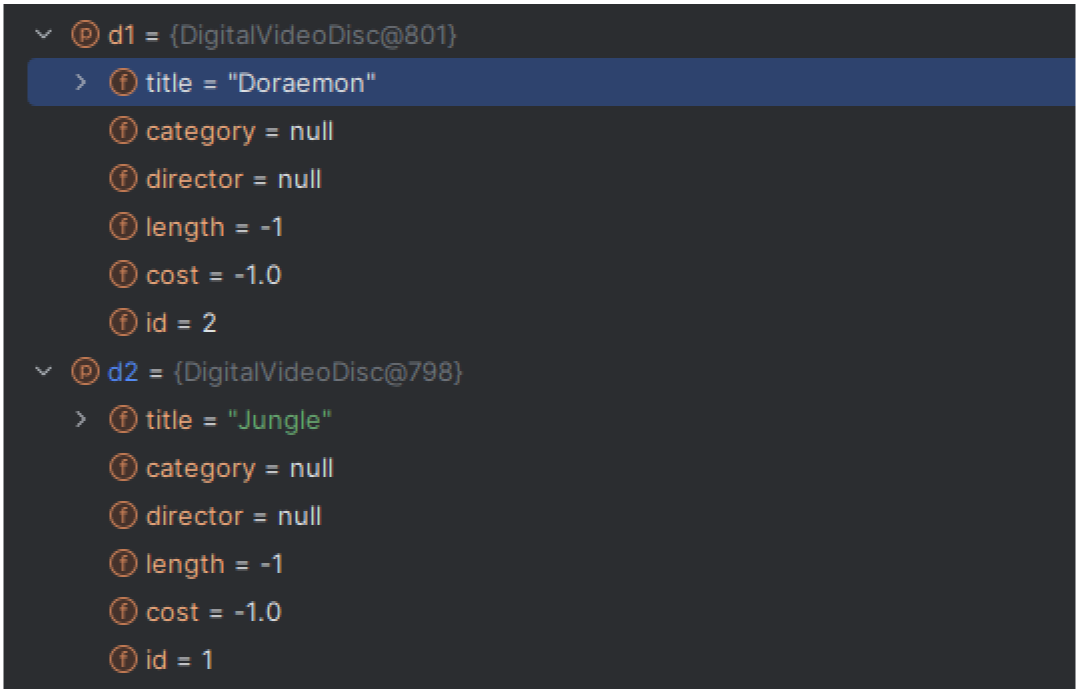
```
d1 = {DigitalVideoDisc@798}
  title = "Jungle"
  category = null
  director = null
  length = -1
  cost = -1.0
  id = 1
d2 = {DigitalVideoDisc@801}
  title = "Cinderella"
  category = null
  director = null
  length = -1
  cost = -1.0
  id = 2
temp = {DigitalVideoDisc@798}
  title = "Jungle"
  category = null
```

4.2.4. Investigate value of variables



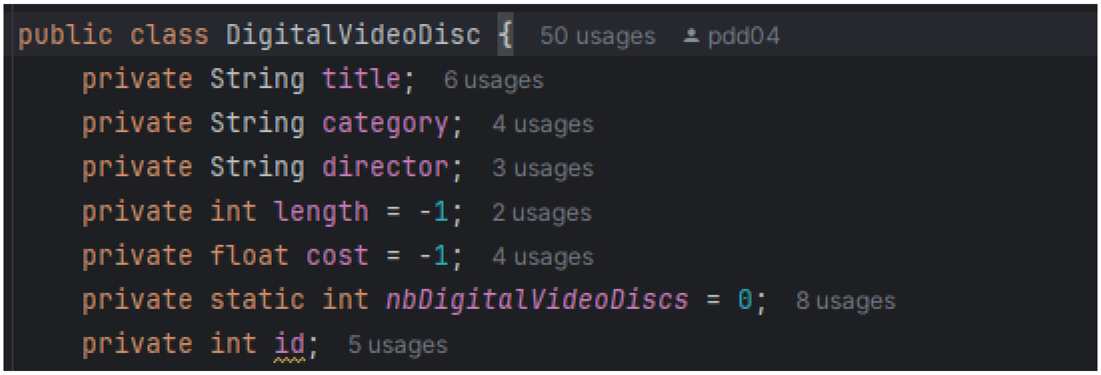
```
v (p) d1 = {DigitalVideoDisc@801}
    v (f) title = "Cinderella"
        > (f) value = {byte[10]@804} [67, 105, 110, 100, 101, 114, 101, 108, 108, 97]
          (f) coder = 0
          (f) hash = 0
          (f) hashIsZero = false
      (f) category = null
      (f) director = null
      (f) length = -1
      (f) cost = -1.0
      (f) id = 2
v (p) d2 = {DigitalVideoDisc@798}
    > (f) title = "Jungle"
      (f) category = null
      (f) director = null
      (f) length = -1
      (f) cost = -1.0
      (f) id = 1
v temp = {DigitalVideoDisc@798}
    > (f) title = "Jungle"
      (f) category = null
      (f) director = null
      (f) length = -1
      (f) cost = -1.0
      (f) id = 1
```

4.2.5. Change value of variables

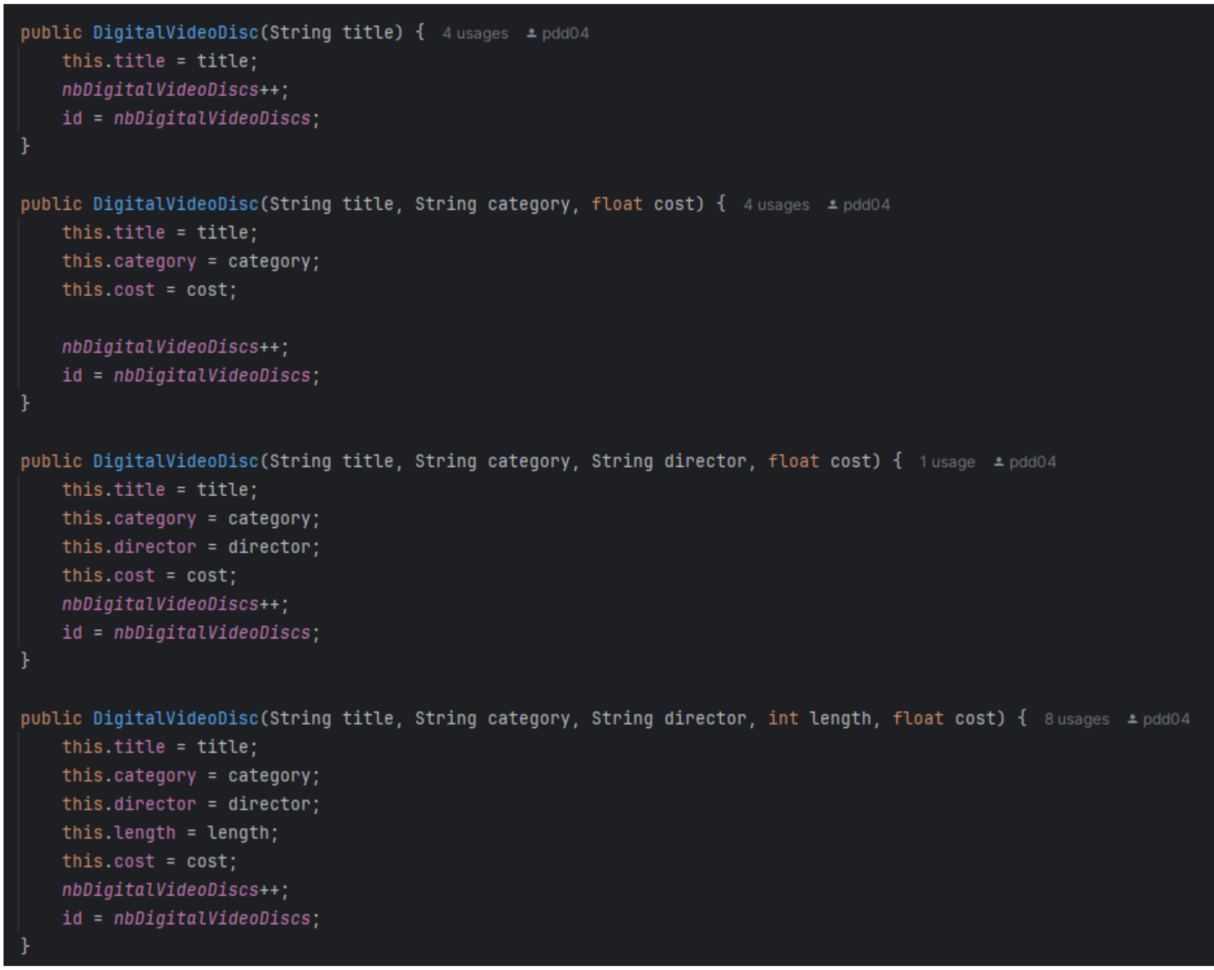


```
v (p) d1 = {DigitalVideoDisc@801}
    > (f) title = "Doraemon"
      (f) category = null
      (f) director = null
      (f) length = -1
      (f) cost = -1.0
      (f) id = 2
v (p) d2 = {DigitalVideoDisc@798}
    > (f) title = "Jungle"
      (f) category = null
      (f) director = null
      (f) length = -1
      (f) cost = -1.0
      (f) id = 1
```

5. Classifier Member and Instance Member

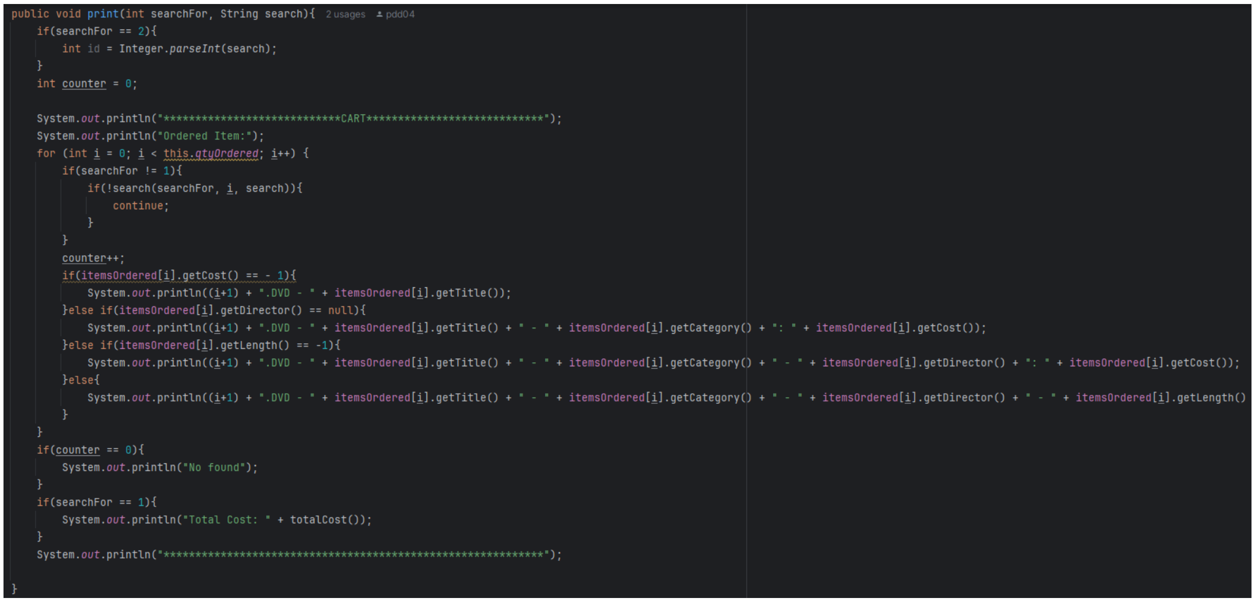


```
public class DigitalVideoDisc {  50 usages  pdd04
    private String title;  6 usages
    private String category;  4 usages
    private String director;  3 usages
    private int length = -1;  2 usages
    private float cost = -1;  4 usages
    private static int nbDigitalVideoDiscs = 0;  8 usages
    private int id;  5 usages
```



```
public DigitalVideoDisc(String title) {  4 usages  pdd04
    this.title = title;
    nbDigitalVideoDiscs++;
    id = nbDigitalVideoDiscs;
}

public DigitalVideoDisc(String title, String category, float cost) {  4 usages  pdd04
    this.title = title;
    this.category = category;
    this.cost = cost;

    nbDigitalVideoDiscs++;
    id = nbDigitalVideoDiscs;
}

public DigitalVideoDisc(String title, String category, String director, float cost) {  1 usage  pdd04
    this.title = title;
    this.category = category;
    this.director = director;
    this.cost = cost;
    nbDigitalVideoDiscs++;
    id = nbDigitalVideoDiscs;
}

public DigitalVideoDisc(String title, String category, String director, int length, float cost) {  8 usages  pdd04
    this.title = title;
    this.category = category;
    this.director = director;
    this.length = length;
    this.cost = cost;
    nbDigitalVideoDiscs++;
    id = nbDigitalVideoDiscs;
}
```

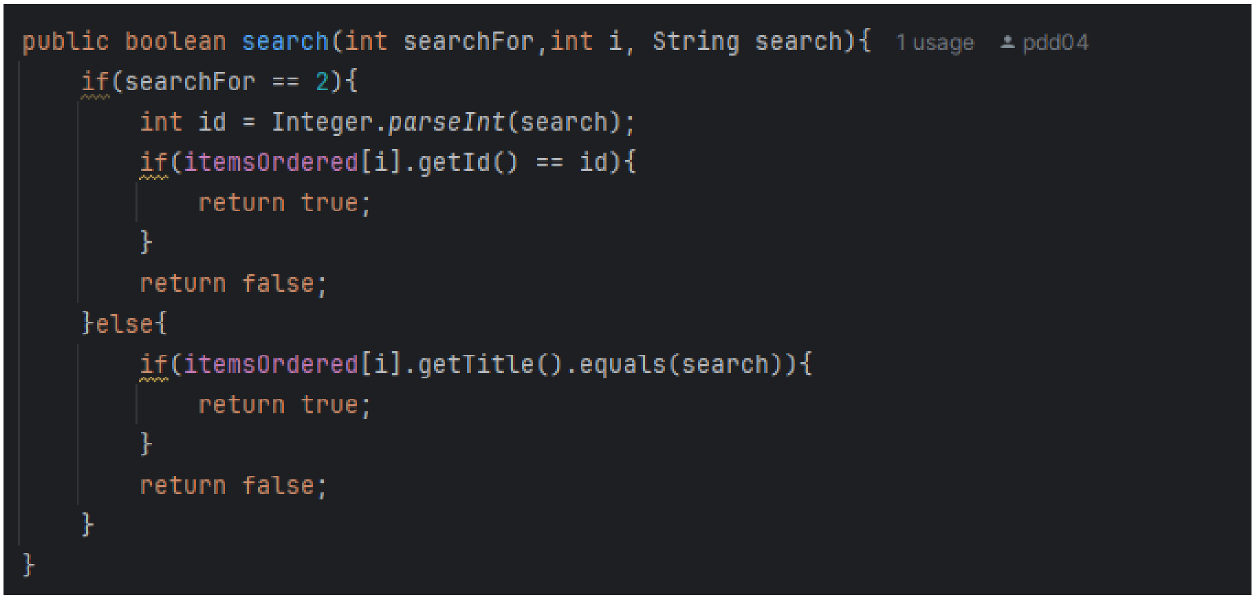
6. **Open the Cart class**
   Create a new method to print the list of ordered items of a cart, the price of each item, and the total price. Format the outline as below:



```java
public void print(int searchFor, String search){  2 usages  pdd04
    if(searchFor == 2){
        int id = Integer.parseInt(search);
    }
    int counter = 0;

    System.out.println("****************************CART****************************");
    System.out.println("Ordered Item:");
    for (int i = 0; i < this.qtyOrdered; i++) {
        if(searchFor != 1){
            if(!search(searchFor, i, search)){
                continue;
            }
        }
        counter++;
        if(itemsOrdered[i].getCost() == - 1){
            System.out.println((i+1) + ".DVD - " + itemsOrdered[i].getTitle());
        }else if(itemsOrdered[i].getDirector() == null){
            System.out.println((i+1) + ".DVD - " + itemsOrdered[i].getTitle() + " - " + itemsOrdered[i].getCategory() + ": " + itemsOrdered[i].getCost());
        }else if(itemsOrdered[i].getLength() == -1){
            System.out.println((i+1) + ".DVD - " + itemsOrdered[i].getTitle() + " - " + itemsOrdered[i].getCategory() + " - " + itemsOrdered[i].getDirector() + ": " + itemsOrdered[i].getCost());
        }else{
            System.out.println((i+1) + ".DVD - " + itemsOrdered[i].getTitle() + " - " + itemsOrdered[i].getCategory() + " - " + itemsOrdered[i].getDirector() + " - " + itemsOrdered[i].getLength()
        }
    }
    if(counter == 0){
        System.out.println("No found");
    }
    if(searchFor == 1){
        System.out.println("Total Cost: " + totalCost());
    }
    System.out.println("************************************************************");

}
```

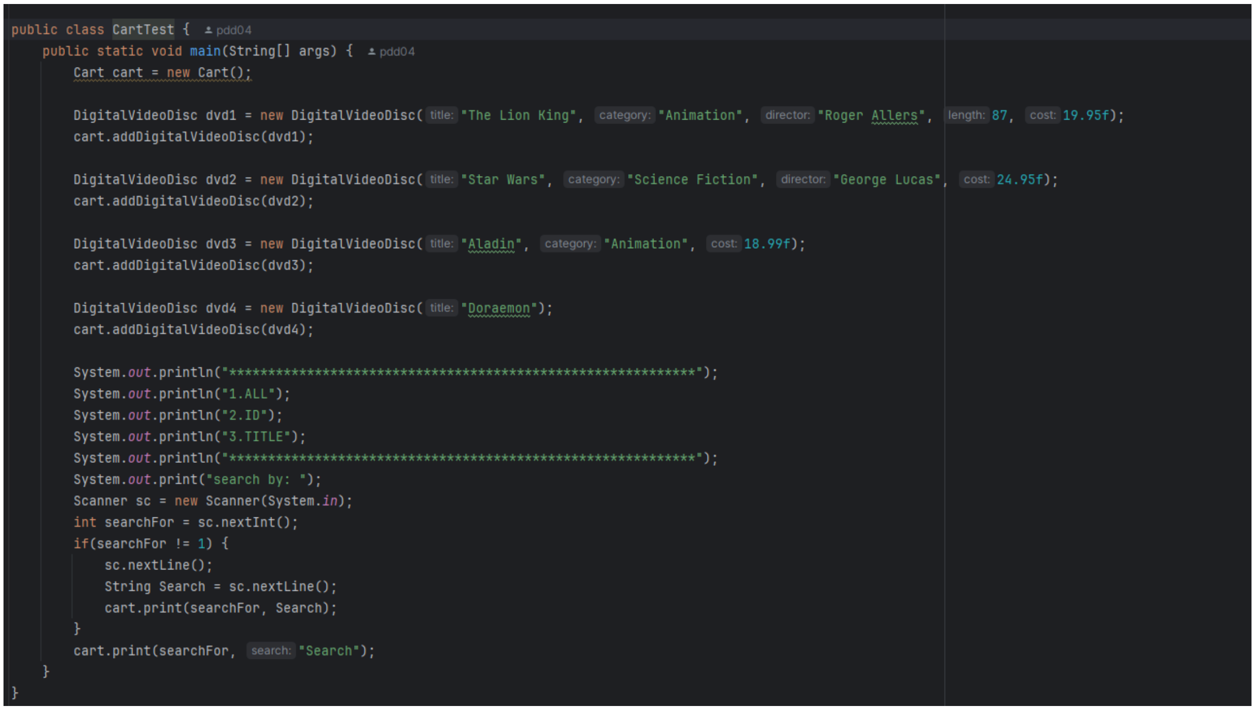
- Search for DVDs in the cart by ID and display the search results. Make sure to notify the user if no match is found.

- Search for DVDs in the cart by title and print the results. Make sure to notify the user if no match is found:



```java
public boolean search(int searchFor,int i, String search){  1 usage  pdd04
    if(searchFor == 2){
        int id = Integer.parseInt(search);
        if(itemsOrdered[i].getId() == id){
            return true;
        }
        return false;
    }else{
        if(itemsOrdered[i].getTitle().equals(search)){
            return true;
        }
        return false;
    }
}
```

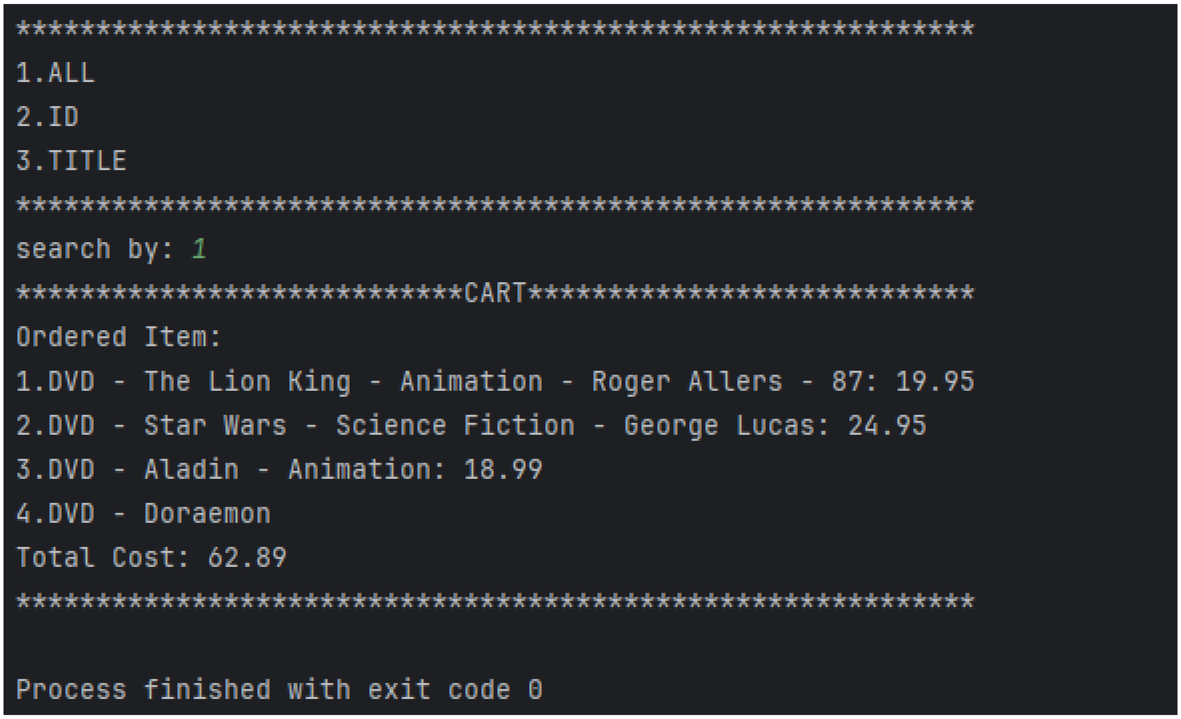
In the CartTest class, write codes to test all methods you have written in this exercise.



```
public class CartTest {  ≛ pdd04
    public static void main(String[] args) {  ≛ pdd04
        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",  cost: 24.95f);
        cart.addDigitalVideoDisc(dvd2);

        DigitalVideoDisc dvd3 = new DigitalVideoDisc( title: "Aladin",  category: "Animation",  cost: 18.99f);
        cart.addDigitalVideoDisc(dvd3);

        DigitalVideoDisc dvd4 = new DigitalVideoDisc( title: "Doraemon");
        cart.addDigitalVideoDisc(dvd4);

        System.out.println("************************************************************");
        System.out.println("1.ALL");
        System.out.println("2.ID");
        System.out.println("3.TITLE");
        System.out.println("************************************************************");
        System.out.print("search by: ");
        Scanner sc = new Scanner(System.in);
        int searchFor = sc.nextInt();
        if(searchFor != 1) {
            sc.nextLine();
            String Search = sc.nextLine();
            cart.print(searchFor, Search);
        }
        cart.print(searchFor,  search: "Search");
    }
}
```

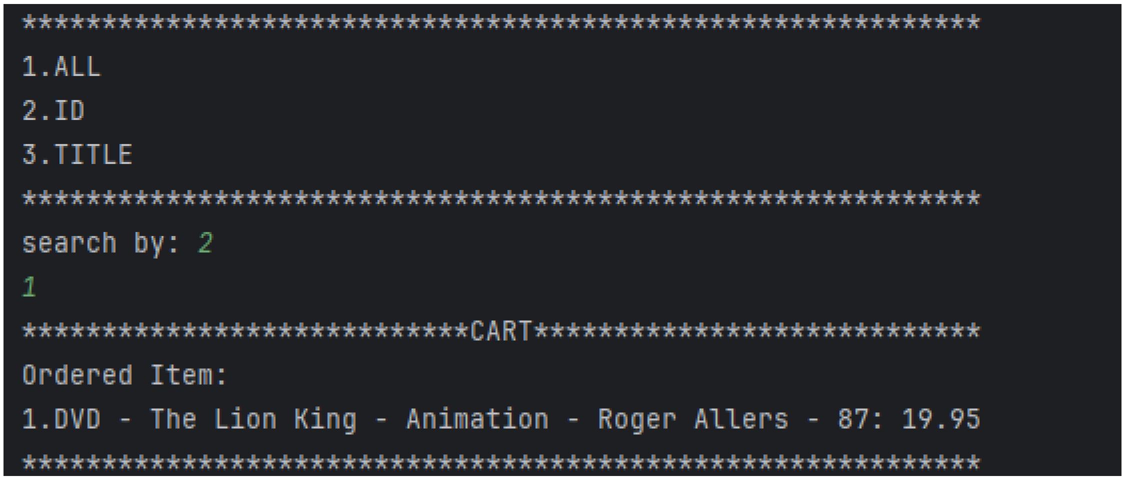
Kết quả:

6.1. Tìm tất cả



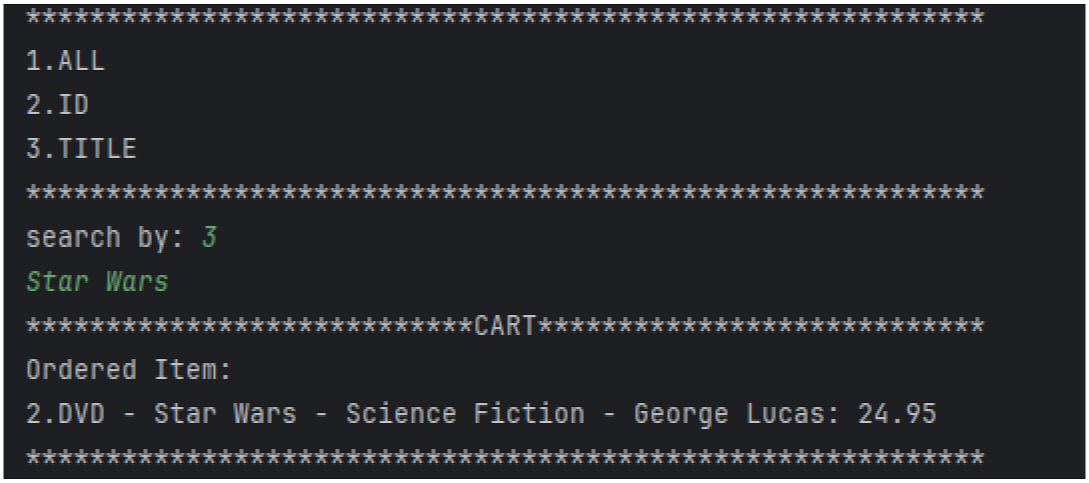
```
************************************************************
1.ALL
2.ID
3.TITLE
************************************************************
search by: 1
****************************CART****************************
Ordered Item:
1.DVD - The Lion King - Animation - Roger Allers - 87: 19.95
2.DVD - Star Wars - Science Fiction - George Lucas: 24.95
3.DVD - Aladin - Animation: 18.99
4.DVD - Doraemon
Total Cost: 62.89
************************************************************

Process finished with exit code 0
```

6.2. Tìm theo id



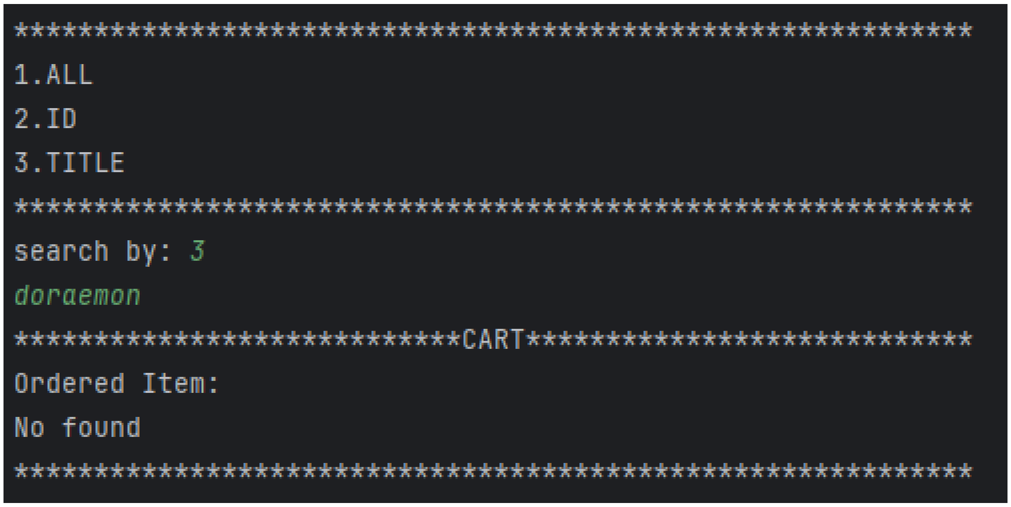
```
************************************************************
1.ALL
2.ID
3.TITLE
************************************************************
search by: 2
1
****************************CART****************************
Ordered Item:
1.DVD - The Lion King - Animation - Roger Allers - 87: 19.95
************************************************************
```

6.3. Tìm theo title



```
************************************************************
1.ALL
2.ID
3.TITLE
************************************************************
search by: 3
Star Wars
****************************CART****************************
Ordered Item:
2.DVD - Star Wars - Science Fiction - George Lucas: 24.95
************************************************************
```

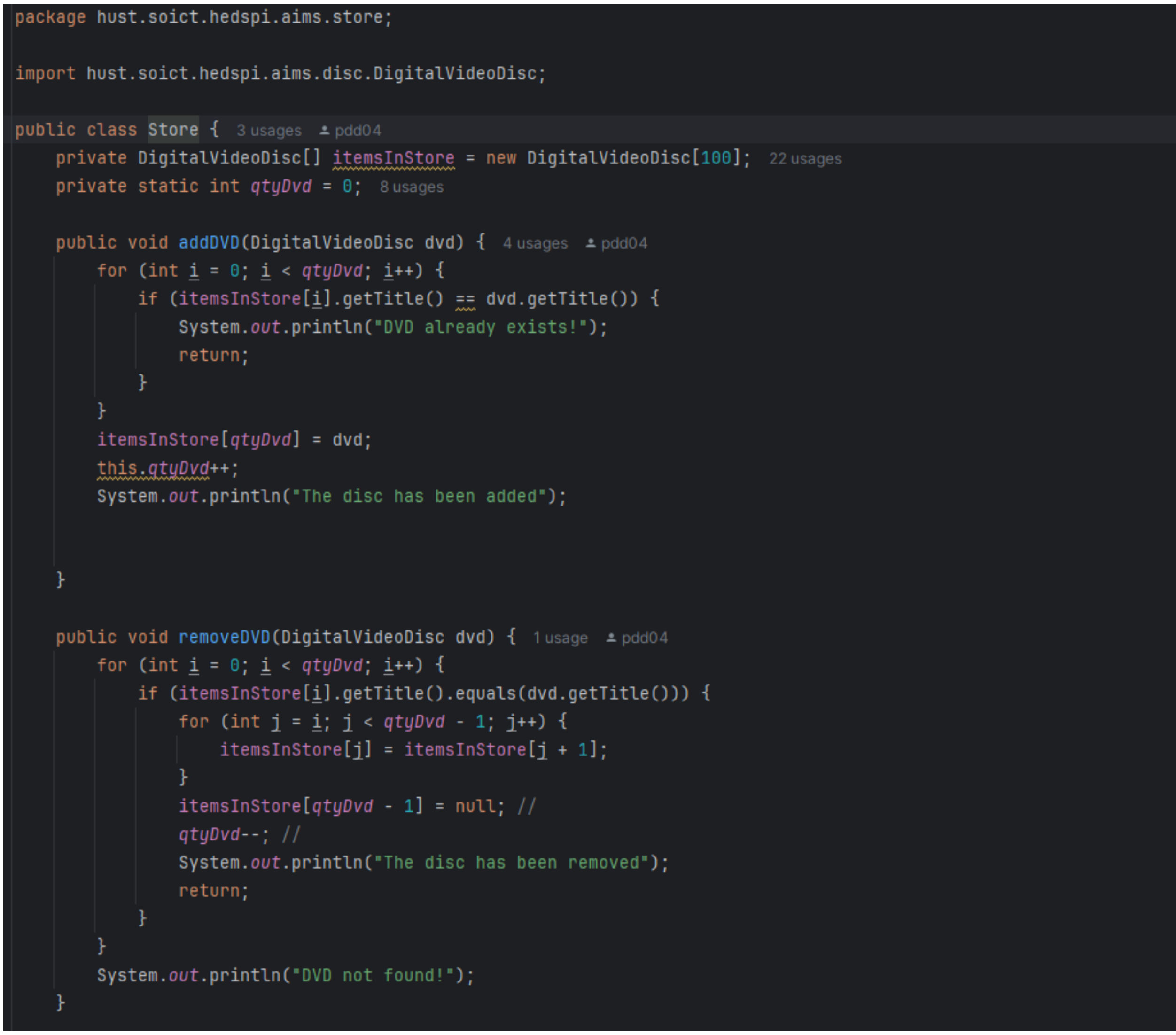
6.4. không tìm thấy



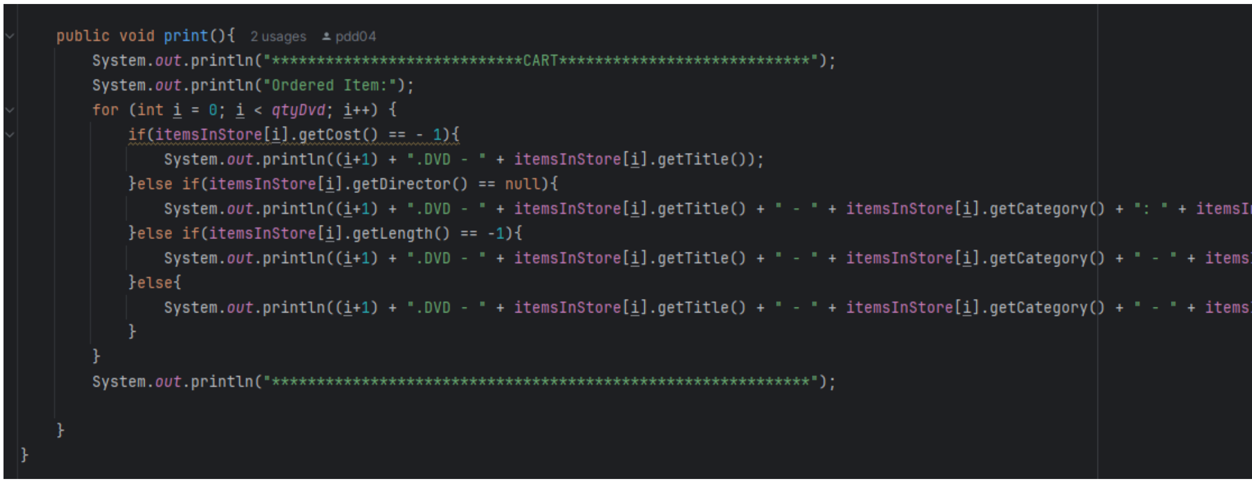
```
************************************************************
1.ALL
2.ID
3.TITLE
************************************************************
search by: 3
doraemon
****************************CART****************************
Ordered Item:
No found
************************************************************
```

7. Implement the Store class

7.1. Class Store

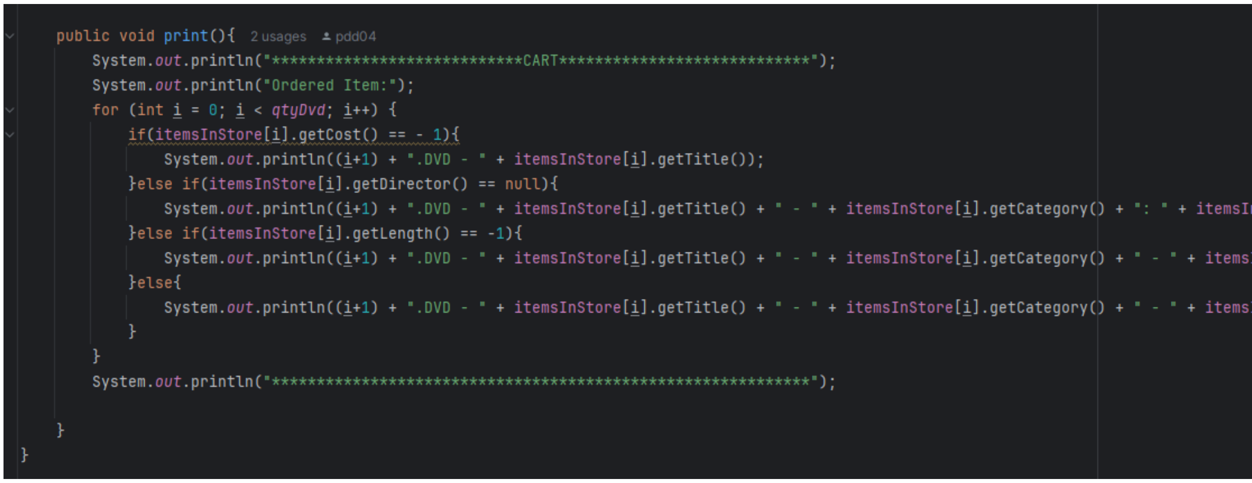


```
package hust.soict.hedspi.aims.store;

import hust.soict.hedspi.aims.disc.DigitalVideoDisc;

public class Store {  3 usages  ≗ pdd04
    private DigitalVideoDisc[] itemsInStore = new DigitalVideoDisc[100];  22 usages
    private static int qtyDvd = 0;  8 usages

    public void addDVD(DigitalVideoDisc dvd) {  4 usages  ≗ pdd04
        for (int i = 0; i < qtyDvd; i++) {
            if (itemsInStore[i].getTitle() == dvd.getTitle()) {
                System.out.println("DVD already exists!");
                return;
            }
        }
        itemsInStore[qtyDvd] = dvd;
        this.qtyDvd++;
        System.out.println("The disc has been added");


    }

    public void removeDVD(DigitalVideoDisc dvd) {  1 usage  ≗ pdd04
        for (int i = 0; i < qtyDvd; i++) {
            if (itemsInStore[i].getTitle().equals(dvd.getTitle())) {
                for (int j = i; j < qtyDvd - 1; j++) {
                    itemsInStore[j] = itemsInStore[j + 1];
                }
                itemsInStore[qtyDvd - 1] = null; //
                qtyDvd--; //
                System.out.println("The disc has been removed");
                return;
            }
        }
        System.out.println("DVD not found!");
    }
```



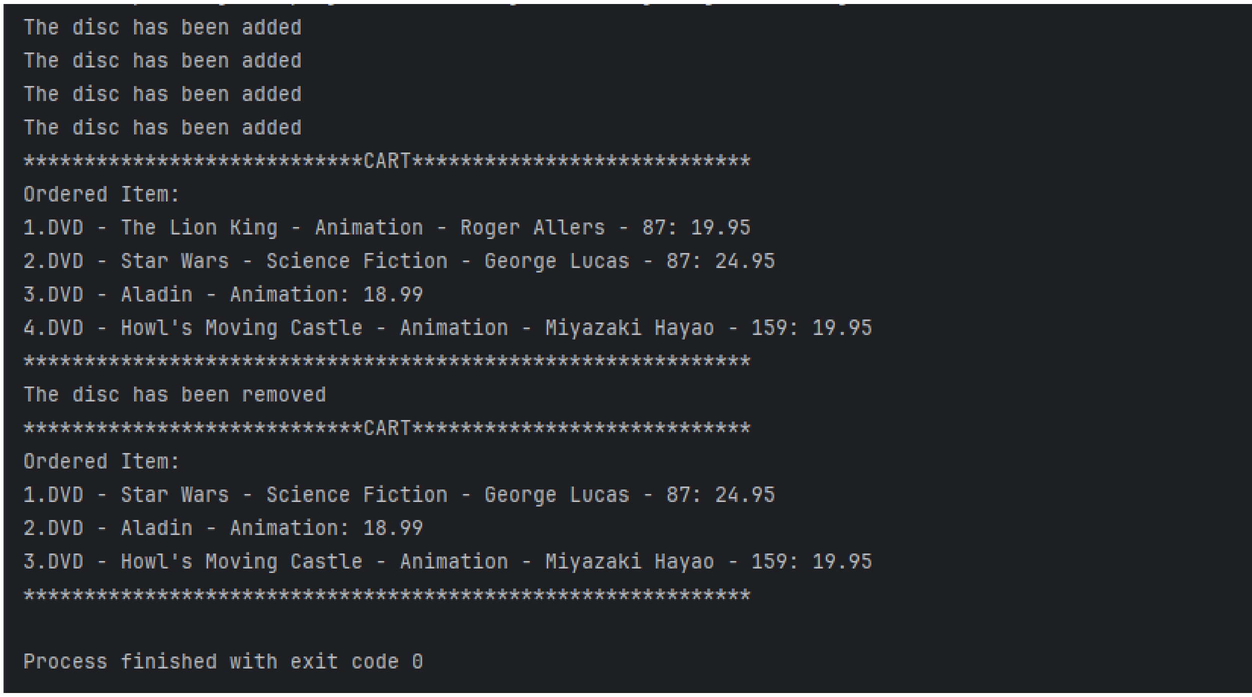
```
    public void print(){  2 usages  ≗ pdd04
        System.out.println("****************************CART****************************");
        System.out.println("Ordered Item:");
        for (int i = 0; i < qtyDvd; i++) {
            if(itemsInStore[i].getCost() == - 1){
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle());
            }else if(itemsInStore[i].getDirector() == null){
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle() + " - " + itemsInStore[i].getCategory() + ": " + itemsI
            }else if(itemsInStore[i].getLength() == -1){
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle() + " - " + itemsInStore[i].getCategory() + " - " + items
            }else{
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle() + " - " + itemsInStore[i].getCategory() + " - " + items
            }
        }
        System.out.println("************************************************************");

    }
}
```

7.2. Class StoreTess



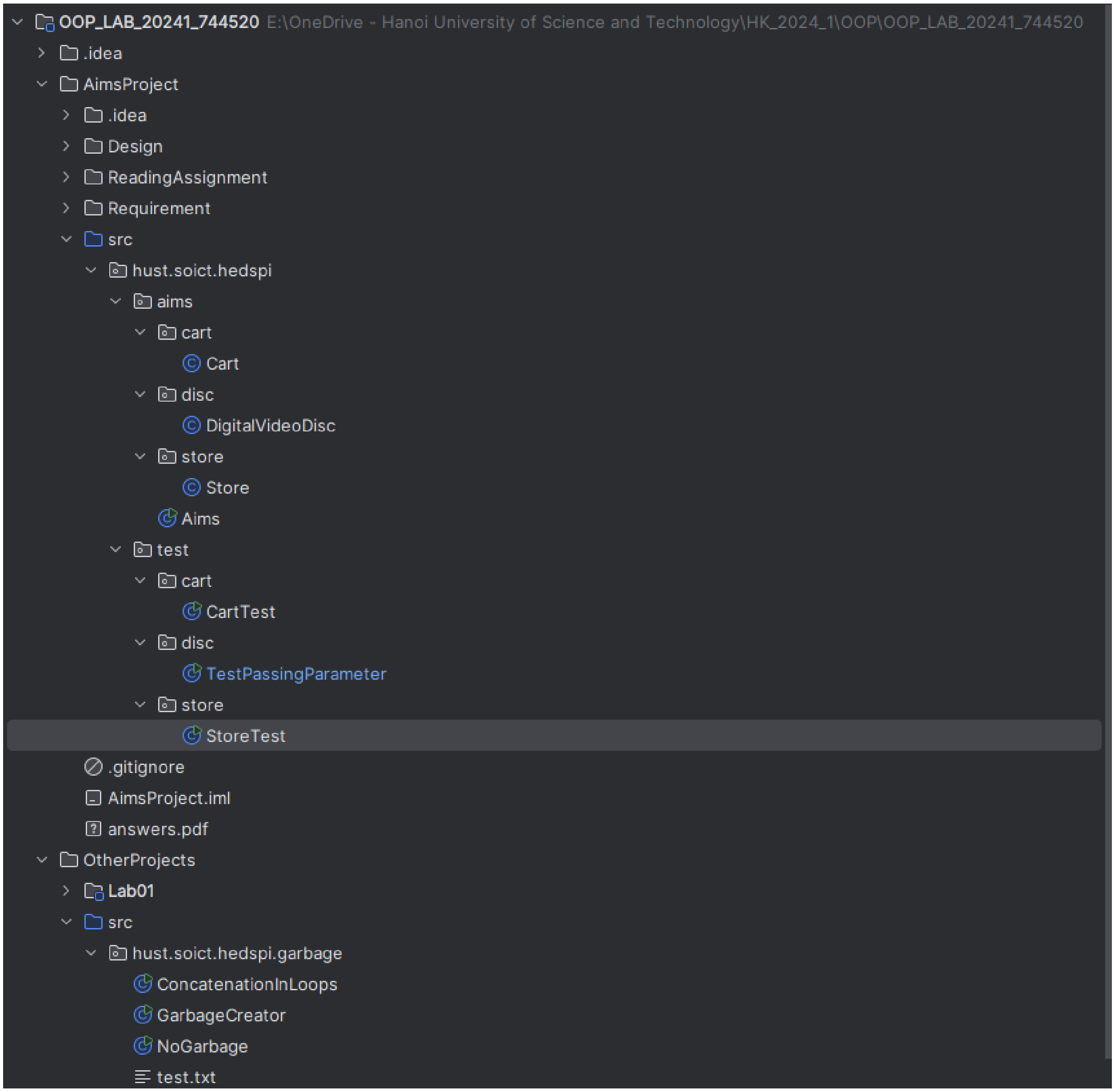
```
    public void print(){  2 usages  pdd04
        System.out.println("***************************CART***************************");
        System.out.println("Ordered Item:");
        for (int i = 0; i < qtyDvd; i++) {
            if(itemsInStore[i].getCost() == - 1){
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle());
            }else if(itemsInStore[i].getDirector() == null){
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle() + " - " + itemsInStore[i].getCategory() + ": " + itemsI
            }else if(itemsInStore[i].getLength() == -1){
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle() + " - " + itemsInStore[i].getCategory() + " - " + items
            }else{
                System.out.println((i+1) + ".DVD - " + itemsInStore[i].getTitle() + " - " + itemsInStore[i].getCategory() + " - " + items
            }
        }
        System.out.println("***********************************************************");

    }
}
```

Kết quả :



```
The disc has been added
The disc has been added
The disc has been added
The disc has been added
****************************CART****************************
Ordered Item:
1.DVD - The Lion King - Animation - Roger Allers - 87: 19.95
2.DVD - Star Wars - Science Fiction - George Lucas - 87: 24.95
3.DVD - Aladin - Animation: 18.99
4.DVD - Howl's Moving Castle - Animation - Miyazaki Hayao - 159: 19.95
************************************************************
The disc has been removed
****************************CART****************************
Ordered Item:
1.DVD - Star Wars - Science Fiction - George Lucas - 87: 24.95
2.DVD - Aladin - Animation: 18.99
3.DVD - Howl's Moving Castle - Animation - Miyazaki Hayao - 159: 19.95
************************************************************

Process finished with exit code 0
```

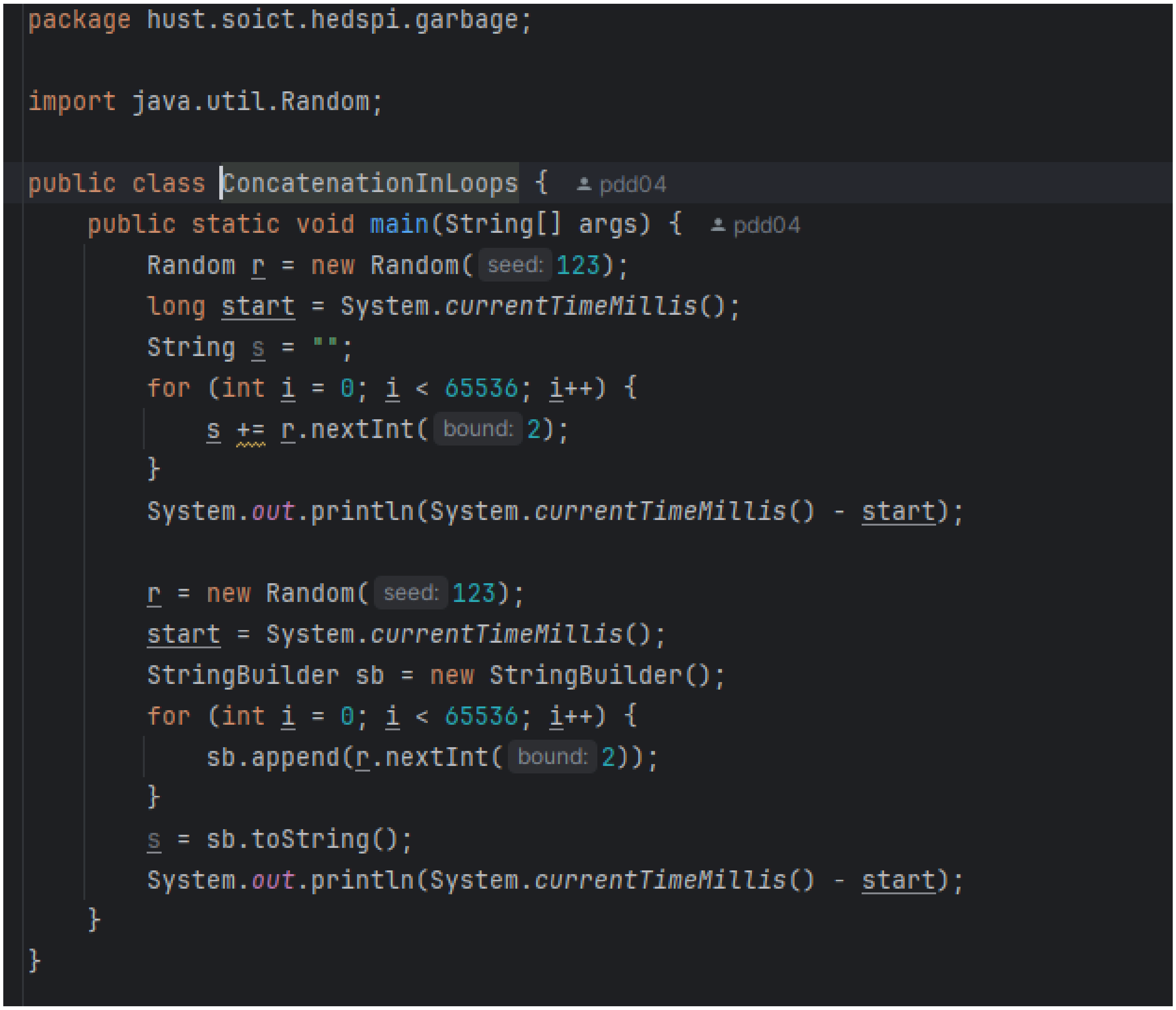
8. Re-organize your projects



```
OOP_LAB_20241_744520 E:\OneDrive - Hanoi University of Science and Technology\HK_2024_1\OOP\OOP_LAB_20241_744520
  .idea
  AimsProject
    .idea
    Design
    ReadingAssignment
    Requirement
    src
      hust.soict.hedspi
        aims
          cart
            Cart
          disc
            DigitalVideoDisc
          store
            Store
          Aims
        test
          cart
            CartTest
          disc
            TestPassingParameter
          store
            StoreTest
    .gitignore
    AimsProject.iml
    answers.pdf
  OtherProjects
    Lab01
    src
      hust.soict.hedspi.garbage
        ConcatenationInLoops
        GarbageCreator
        NoGarbage
        test.txt
```

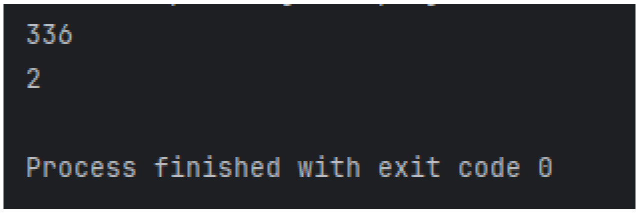
9. String, StringBuilder and StringBuffer

- **Class ConcatenationInLoops**



```
package hust.soict.hedspi.garbage;

import java.util.Random;

public class ConcatenationInLoops {  pdd04
    public static void main(String[] args) {  pdd04
        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);
    }
}
```

Kết quả:

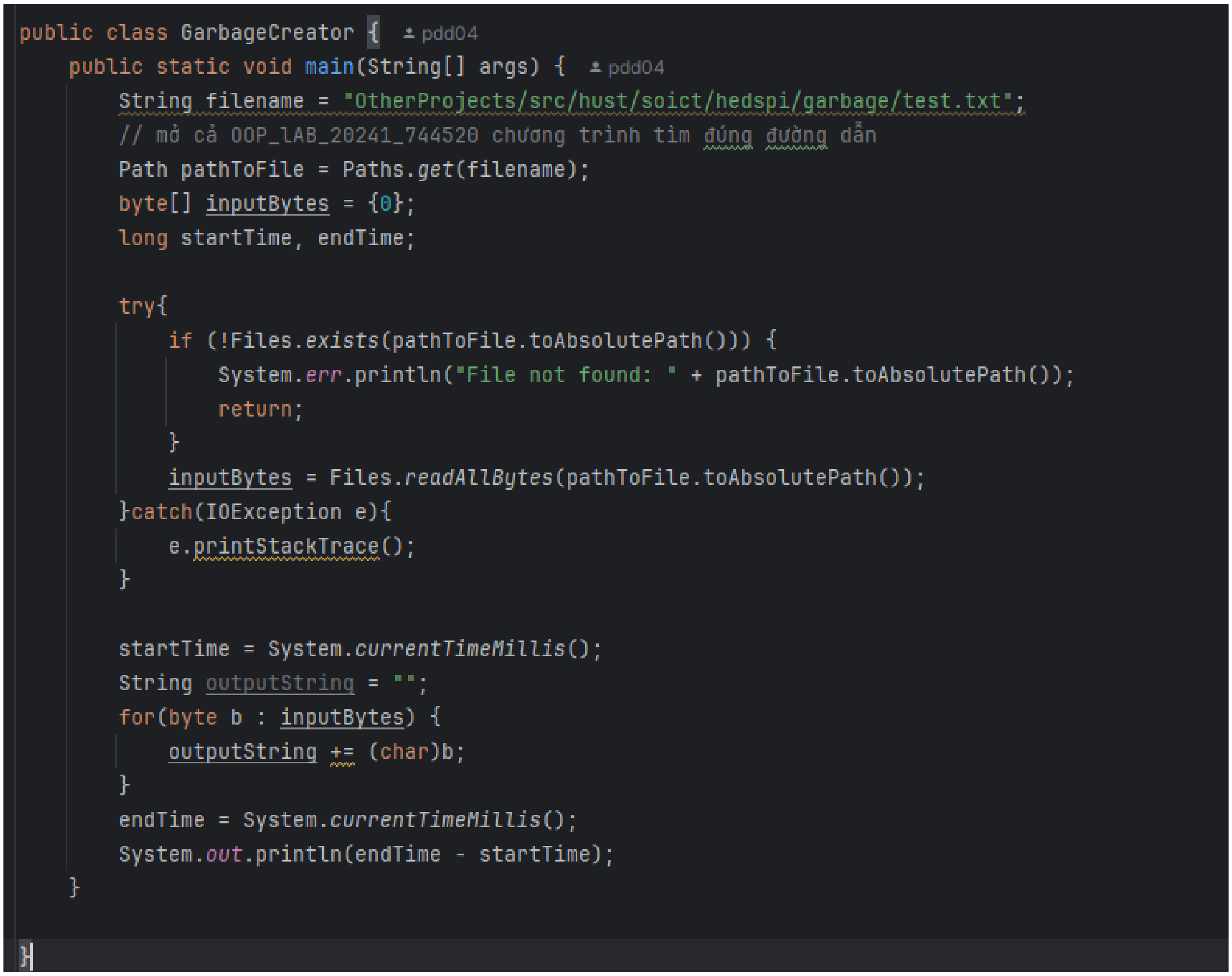


```
336
2

Process finished with exit code 0
```

khi sử dụng toán tử "+" với String java sẽ tạo một đối tượng mới để lưu chuỗi mới vì vậy nếu như ta tạo một vòng lặp 65536 thì sẽ tương đương với 65536 đối tượng được tạo ra -> hiệu suất kém

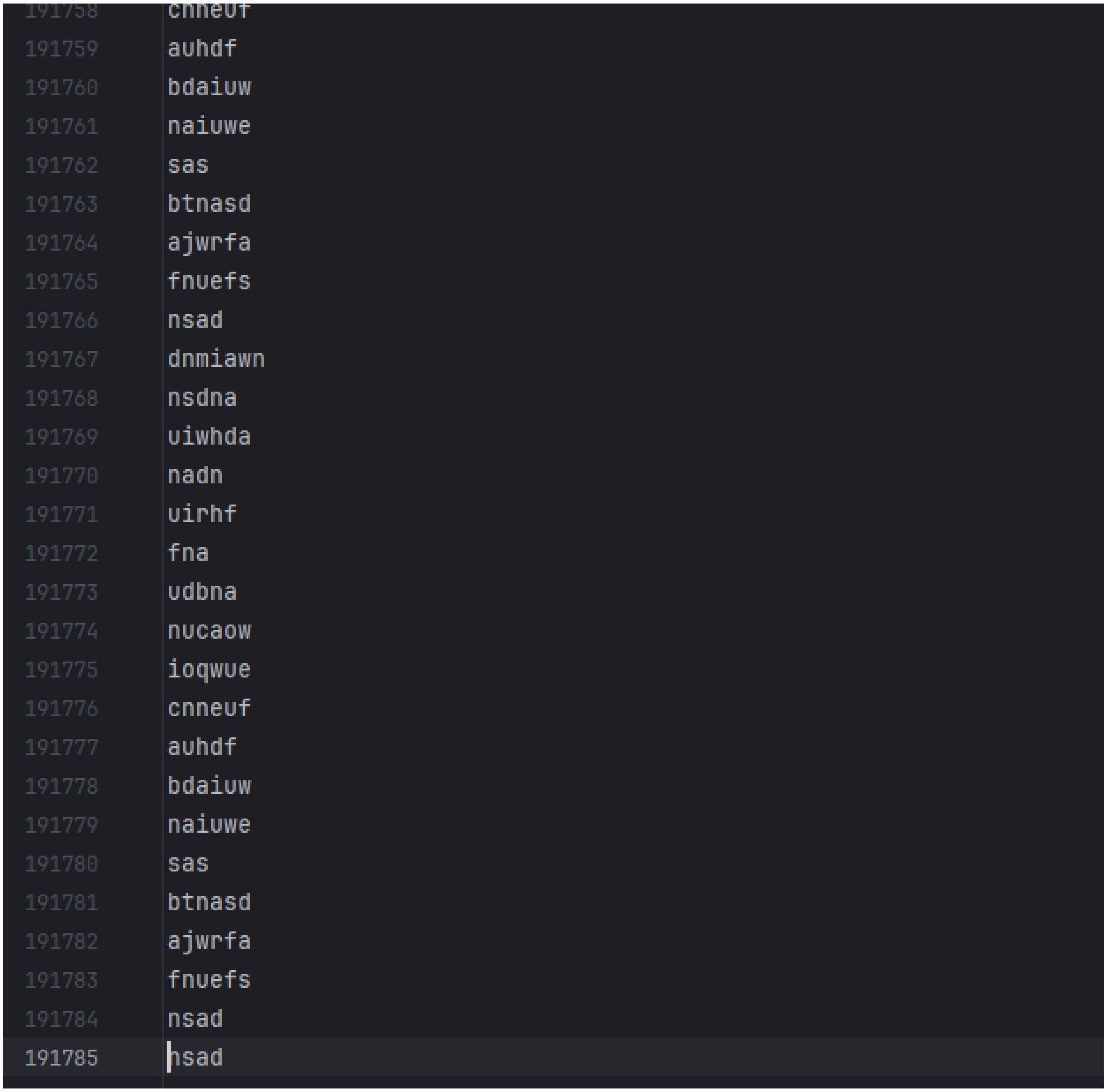
sử dụng StrinBuilder sẽ không tạo đối tượng mới mỗi lần thay đổi chuỗi -> tốc dộ nhanh, hiệu quả.

- **Class GarbageCreator**



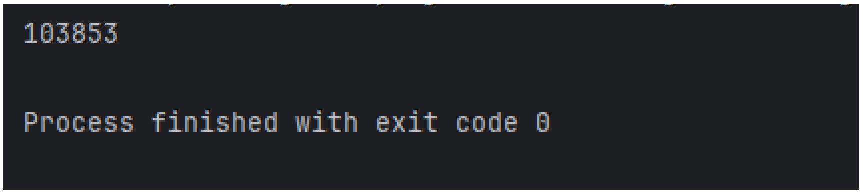
```
public class GarbageCreator {  pdd04
    public static void main(String[] args) {  pdd04
        String filename = "OtherProjects/src/hust/soict/hedspi/garbage/test.txt";
        // mở cả OOP_LAB_20241_744520 chương trình tìm đúng đường dẫn
        Path pathToFile = Paths.get(filename);
        byte[] inputBytes = {0};
        long startTime, endTime;

        try{
            if (!Files.exists(pathToFile.toAbsolutePath())) {
                System.err.println("File not found: " + pathToFile.toAbsolutePath());
                return;
            }
            inputBytes = Files.readAllBytes(pathToFile.toAbsolutePath());
        }catch(IOException e){
            e.printStackTrace();
        }

        startTime = System.currentTimeMillis();
        String outputString = "";
        for(byte b : inputBytes) {
            outputString += (char)b;
        }
        endTime = System.currentTimeMillis();
        System.out.println(endTime - startTime);
    }

}
```

Và với 191785 chuỗi rác được tạo ra trong test



```
cnneuf
auhdf
bdaiuw
naiuwe
sas
btnasd
ajwrfa
fnuefs
nsad
dnmiawn
nsdna
uiwhda
nadn
uirhf
fna
udbna
nucaow
ioqwue
cnneuf
auhdf
bdaiuw
naiuwe
sas
btnasd
ajwrfa
fnuefs
nsad
nsad
```

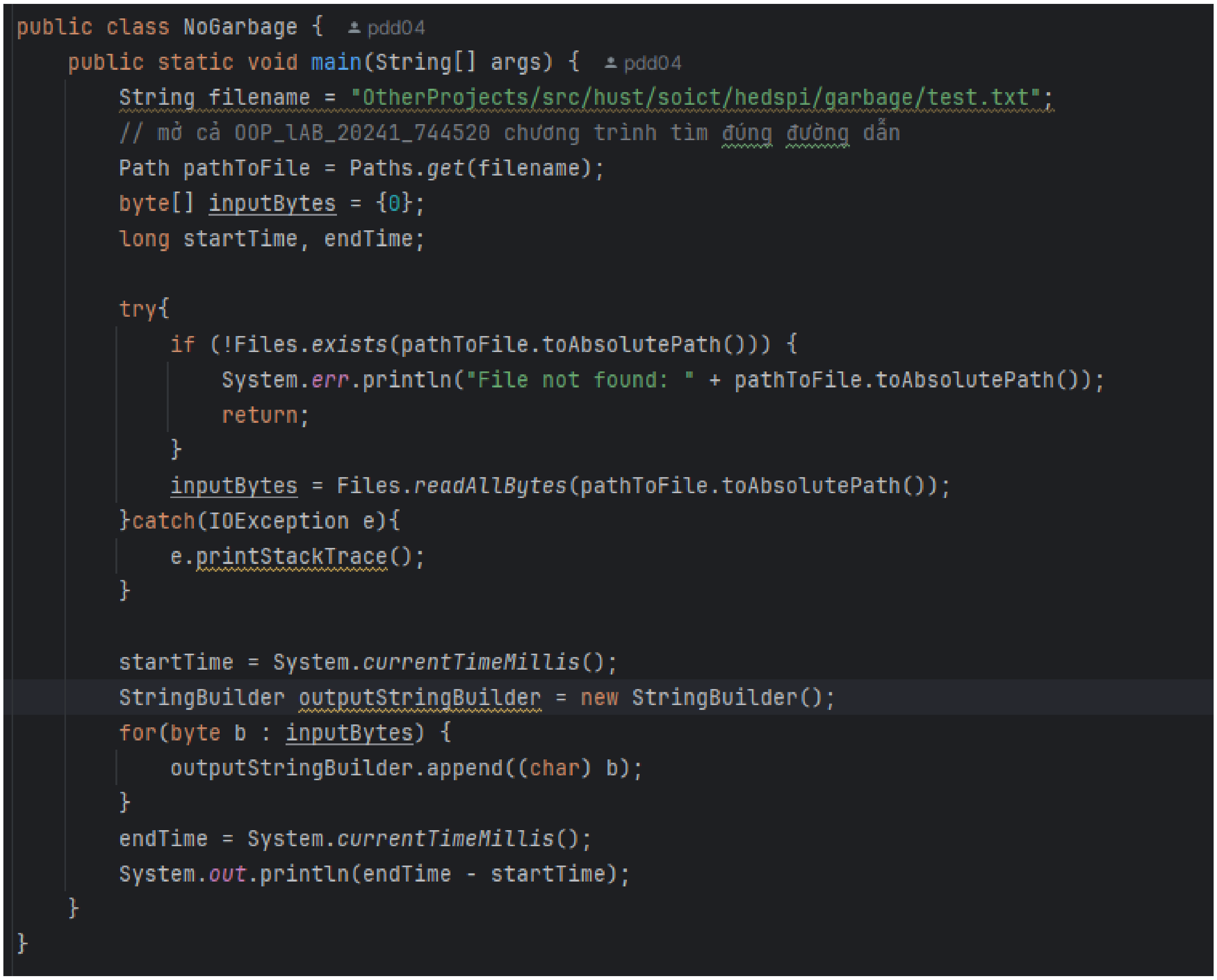
Kết quả



```
103853

Process finished with exit code 0
```

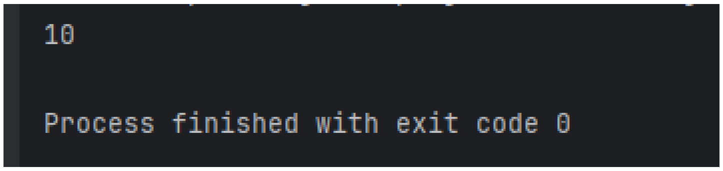
Mất rất nhiều thời gian để chương trình thực hiện xong

- **NoGarbage**



```java
public class NoGarbage {
    public static void main(String[] args) {
        String filename = "OtherProjects/src/hust/soict/hedspi/garbage/test.txt";
        // mở cả OOP_lAB_20241_744520 chương trình tìm đúng đường dẫn
        Path pathToFile = Paths.get(filename);
        byte[] inputBytes = {0};
        long startTime, endTime;

        try{
            if (!Files.exists(pathToFile.toAbsolutePath())) {
                System.err.println("File not found: " + pathToFile.toAbsolutePath());
                return;
            }
            inputBytes = Files.readAllBytes(pathToFile.toAbsolutePath());
        }catch(IOException e){
            e.printStackTrace();
        }

        startTime = System.currentTimeMillis();
        StringBuilder outputStringBuilder = new StringBuilder();
        for(byte b : inputBytes) {
            outputStringBuilder.append((char) b);
        }
        endTime = System.currentTimeMillis();
        System.out.println(endTime - startTime);
    }
}
```

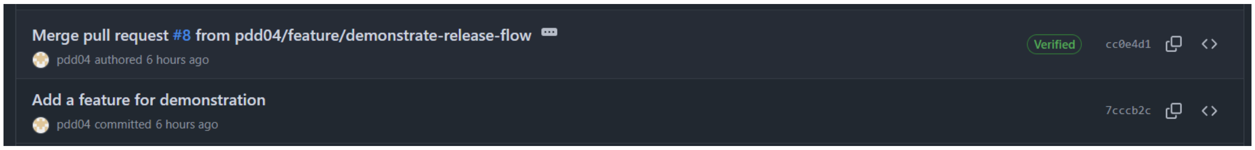
Kết quả:



```
10

Process finished with exit code 0
```

Chương trình xử lý nhanh hơn rất nhiều

10. Release flow demonstration



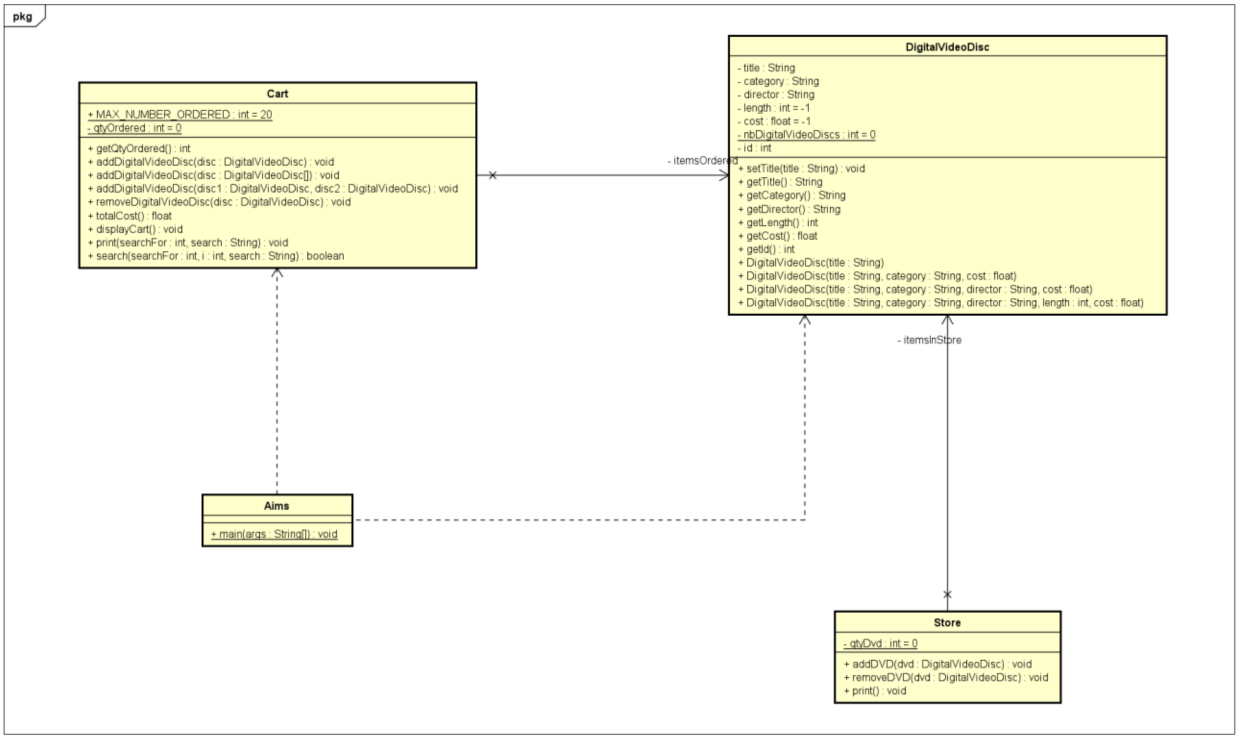
Merge pull request #8 from pdd04/feature/demonstrate-release-flow

pdd04 authored 6 hours ago

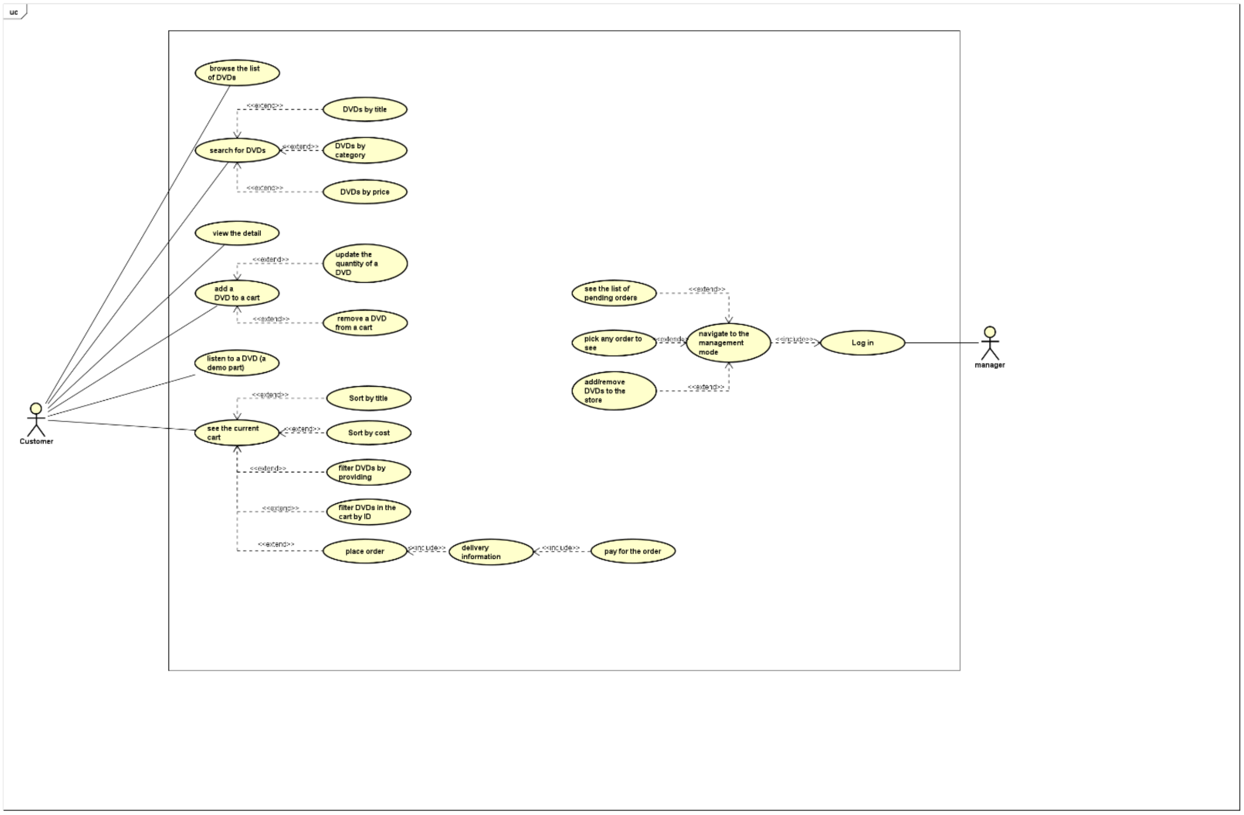
Add a feature for demonstration

pdd04 committed 6 hours ago

11. Update UML


pkg
Cart
+ MAX_NUMBER_ORDERED : int = 20
- qtyOrdered : int = 0
+ getQtyOrdered() : int
+ addDigitalVideoDisc(disc : DigitalVideoDisc) : void
+ addDigitalVideoDisc(disc : DigitalVideoDisc[]) : void
+ addDigitalVideoDisc(disc1 : DigitalVideoDisc, disc2 : DigitalVideoDisc) : void
+ removeDigitalVideoDisc(disc : DigitalVideoDisc) : void
+ totalCost() : float
+ displayCart() : void
+ print(searchFor : int, search : String) : void
+ search(searchFor : int, i : int, search : String) : boolean
- itemsOrdered
DigitalVideoDisc
- title : String
- category : String
- director : String
- length : int = -1
- cost : float = -1
- nbDigitalVideoDiscs : int = 0
- id : int
+ setTitle(title : String) : void
+ getTitle() : String
+ getCategory() : String
+ getDirector() : String
+ getLength() : int
+ getCost() : float
+ getId() : int
+ DigitalVideoDisc(title : String)
+ DigitalVideoDisc(title : String, category : String, cost : float)
+ DigitalVideoDisc(title : String, category : String, director : String, cost : float)
+ DigitalVideoDisc(title : String, category : String, director : String, length : int, cost : float)
- itemsInStore
Aims
+ main(args : String[]) : void
Store
- qtyDvd : int = 0
+ addDVD(dvd : DigitalVideoDisc) : void
+ removeDVD(dvd : DigitalVideoDisc) : void
+ print() : void

12. Use case


uc
browse the list of DVDs
<<extend>>
DVDs by title
search for DVDs
<<extend>>
DVDs by category
<<extend>>
DVDs by price
view the detail
<<extend>>
update the quantity of a DVD
add a DVD to a cart
<<extend>>
remove a DVD from a cart
listen to a DVD (a demo part)
<<extend>>
Sort by title
see the current cart
<<extend>>
Sort by cost
<<extend>>
filter DVDs by providing
<<extend>>
filter DVDs in the cart by ID
<<extend>>
place order
<<include>>
delivery information
<<include>>
pay for the order
Customer
see the list of pending orders
<<extend>>
pick any order to see
<<extend>>
navigate to the management mode
<<include>>
Log in
add/remove DVDs to the store
<<extend>>
manager