**Lab 03: Basic Object-Oriented Techniques**

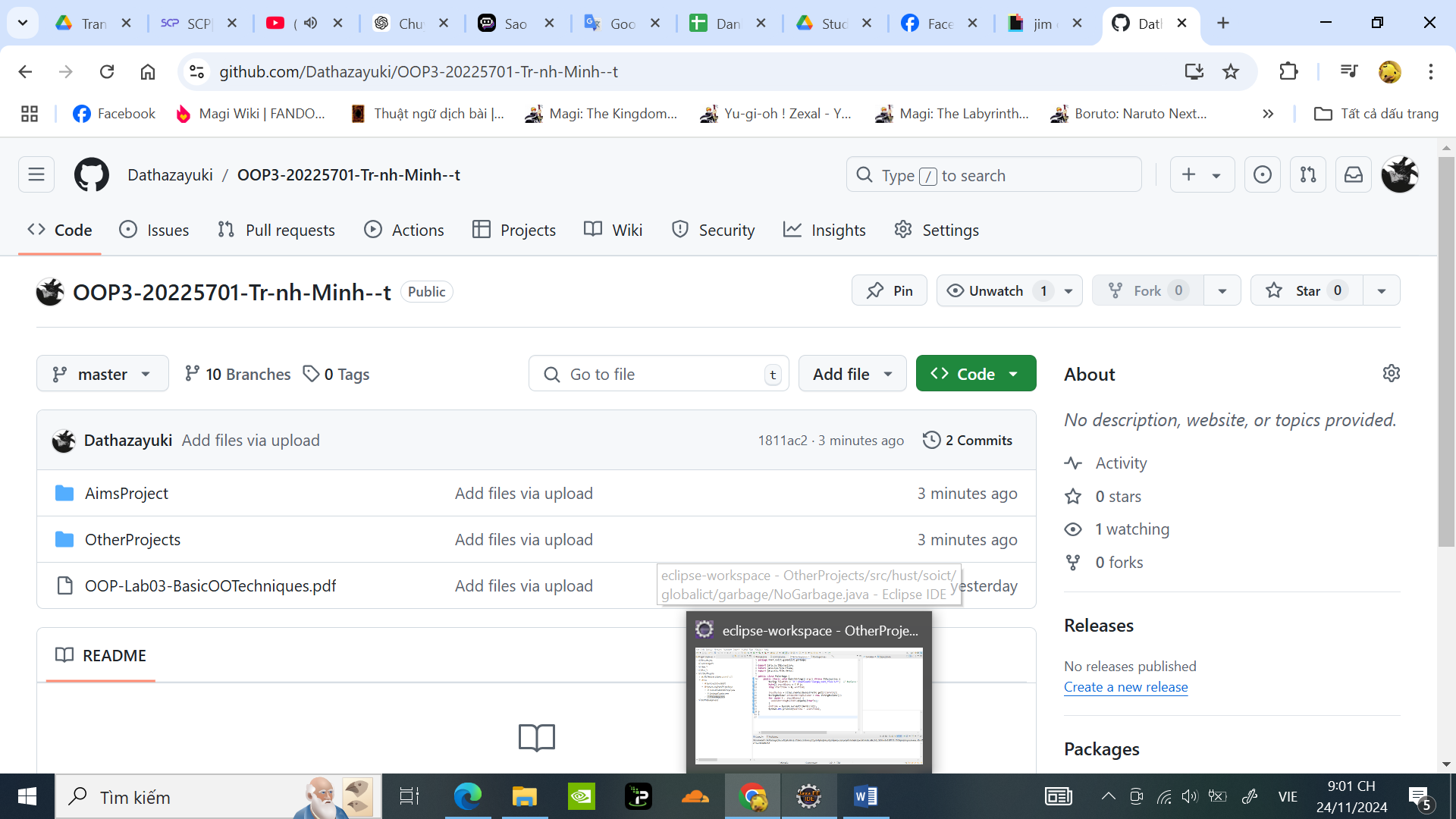
**Họ và Tên : Trịnh Minh Đạt**

**MSSV : 20225701**

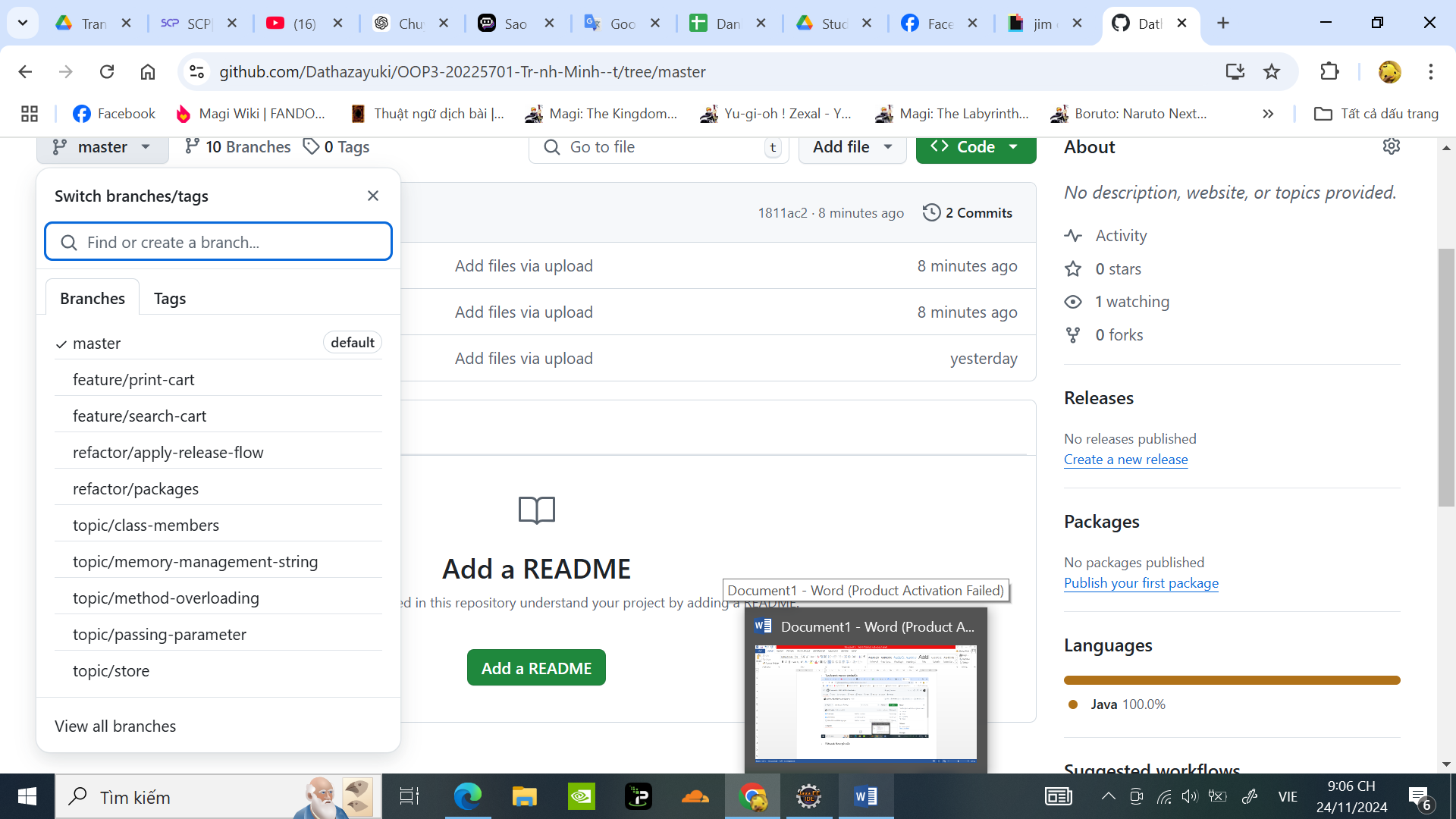
**Link Github: https://github.com/Dathazayuki/OOP3-20225701-Tr-nh-Minh--t/edit/feature/demonstrate-release-flow/README.md**

1. **Branch your repository**

* Tạo Branch master (default):



* 9 Branch theo yêu cầu



1. **Working with method overloading**
   1. **Overloading by differing types of parameter**

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

### Code:

// Add multiple DVDs using an array

**public** **int** addDigitalVideoDisc(DigitalVideoDisc[] dvdList) {

**int** addCount = 0;

**for** (DigitalVideoDisc disc : dvdList) {

**if** (qtyOrdered == ***MAX\_NUMBERS\_ORDER***) {

System.***out***.println("The cart is almost full.");

**break**;

} **else** {

itemOrdered[qtyOrdered] = disc;

qtyOrdered++;

System.***out***.println("The DVD " + "\"" + disc.getTitle() + "\"" + " has been added!");

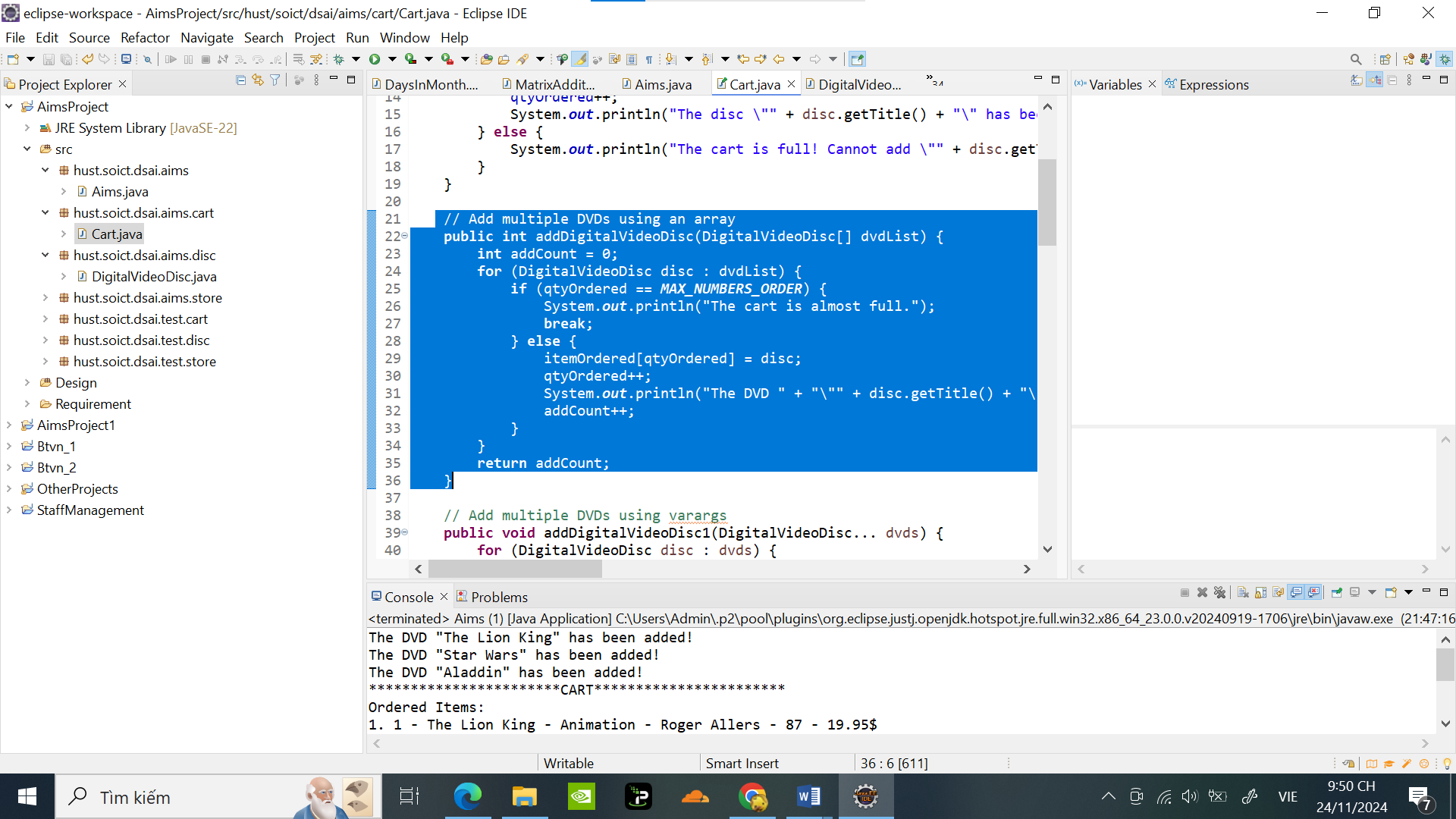
addCount++;

}

}

**return** addCount;

}



* Chạy trong Aims:

+ Câu lệnh trong Aims:

**public** **class** Aims {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

//Câu lệnh trong class Cart

Cart anOrder = **new** Cart();

DigitalVideoDisc[] dvd = **new** DigitalVideoDisc[3];

dvd[0] = **new** DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f);

dvd[1] = **new** DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f);

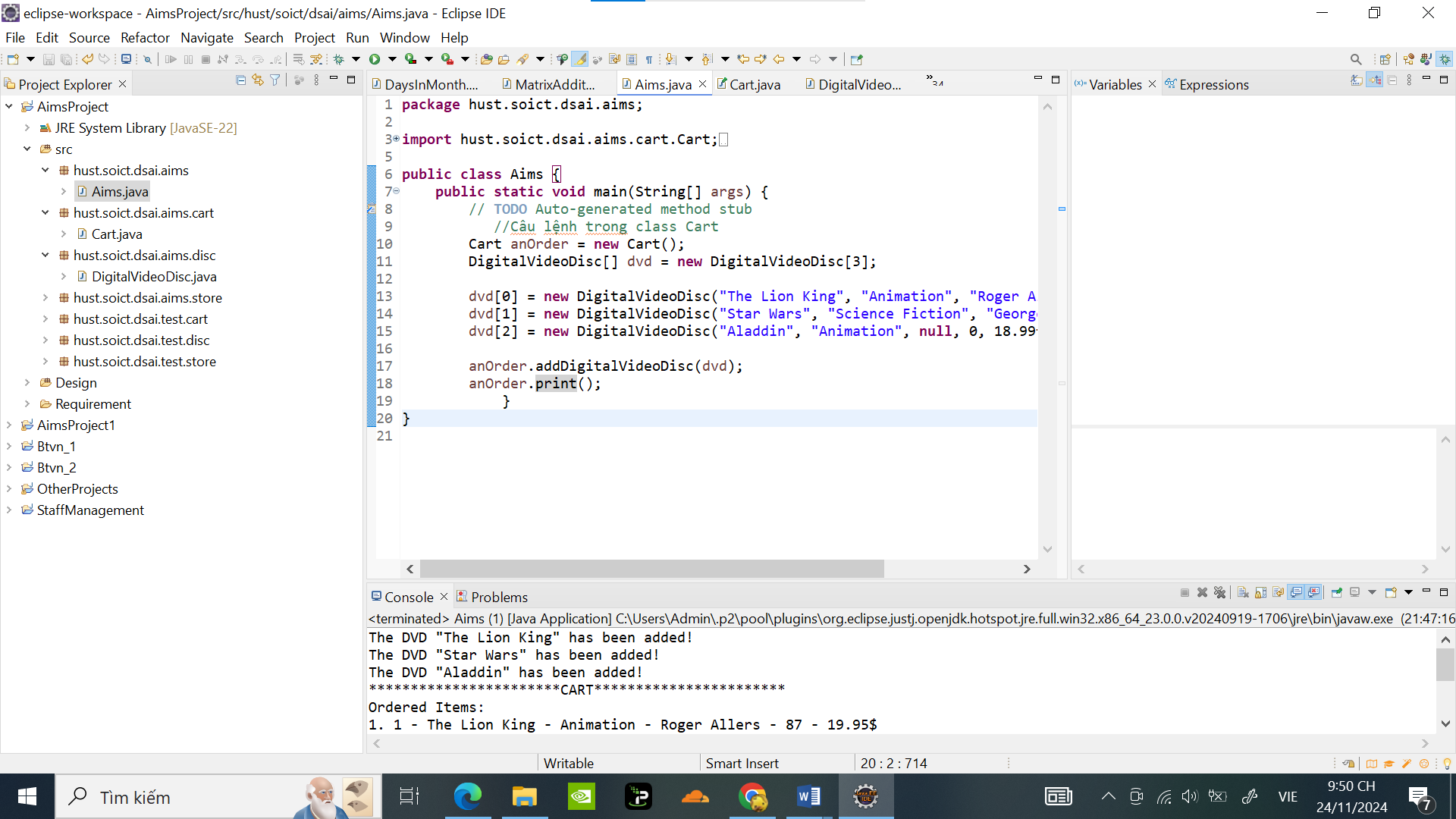
dvd[2] = **new** DigitalVideoDisc("Aladdin", "Animation", **null**, 0, 18.99f);

anOrder.addDigitalVideoDisc(dvd);

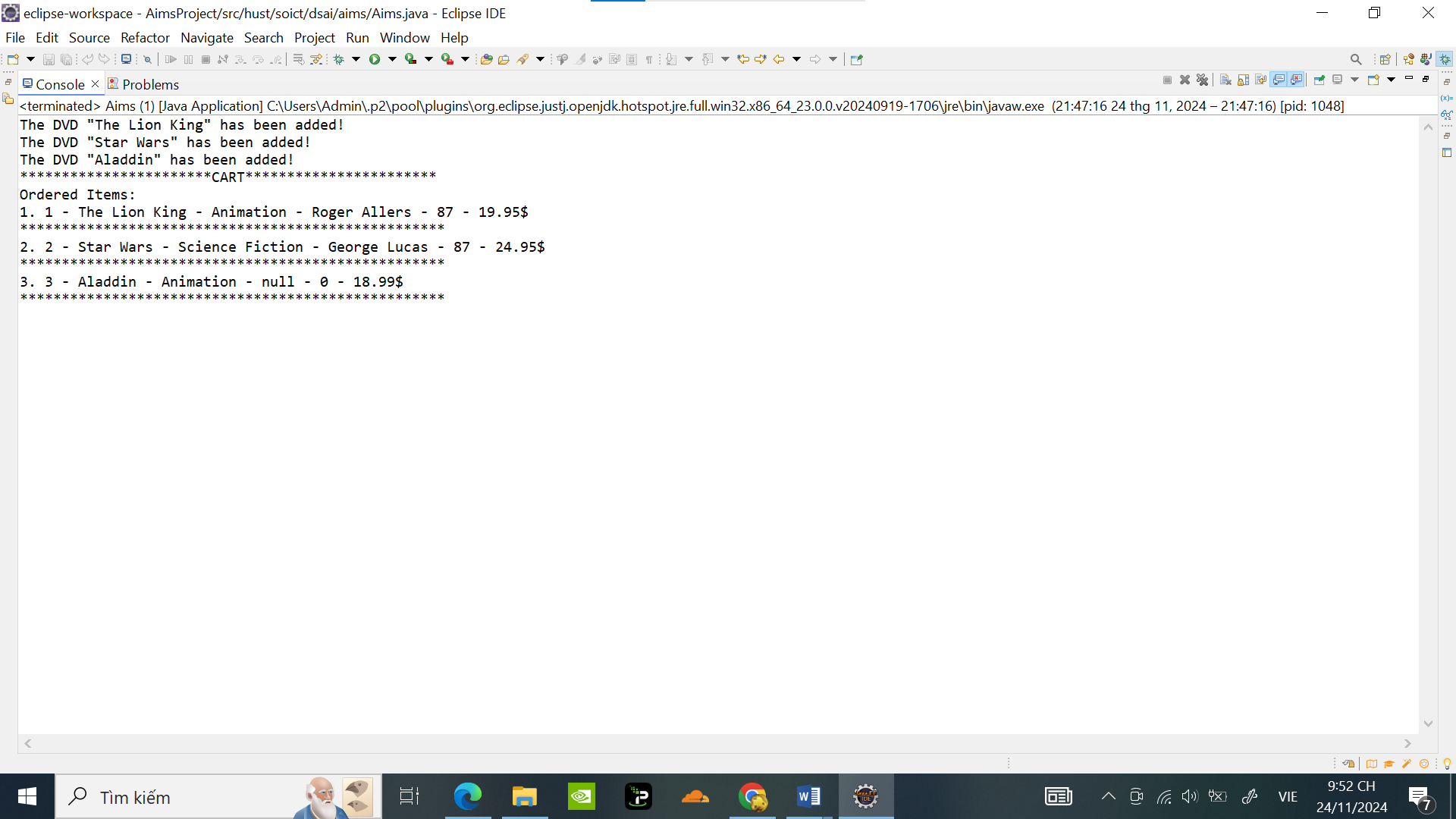
anOrder.print();

}

}



Kết quả :



Phương thức addDigitalVideoDisc(DigitalVideoDisc… dvdList) cho phép thêm một danh sách số đĩa bất kỳ các đĩa DVD vào giỏ hàng

Code :

**public** **int** addDigitalVideoDisc1(DigitalVideoDisc... dvdList) {

**int** addCount = 0;

**for** (DigitalVideoDisc disc : dvdList) {

**if** (qtyOrdered == ***MAX\_NUMBERS\_ORDER***) {

System.***out***.println("The cart is almost full.");

**break**;

} **else** {

itemOrdered[qtyOrdered] = disc;

qtyOrdered++;

System.***out***.println("The DVD " + "\"" + disc.getTitle() + "\"" + " has been added!");

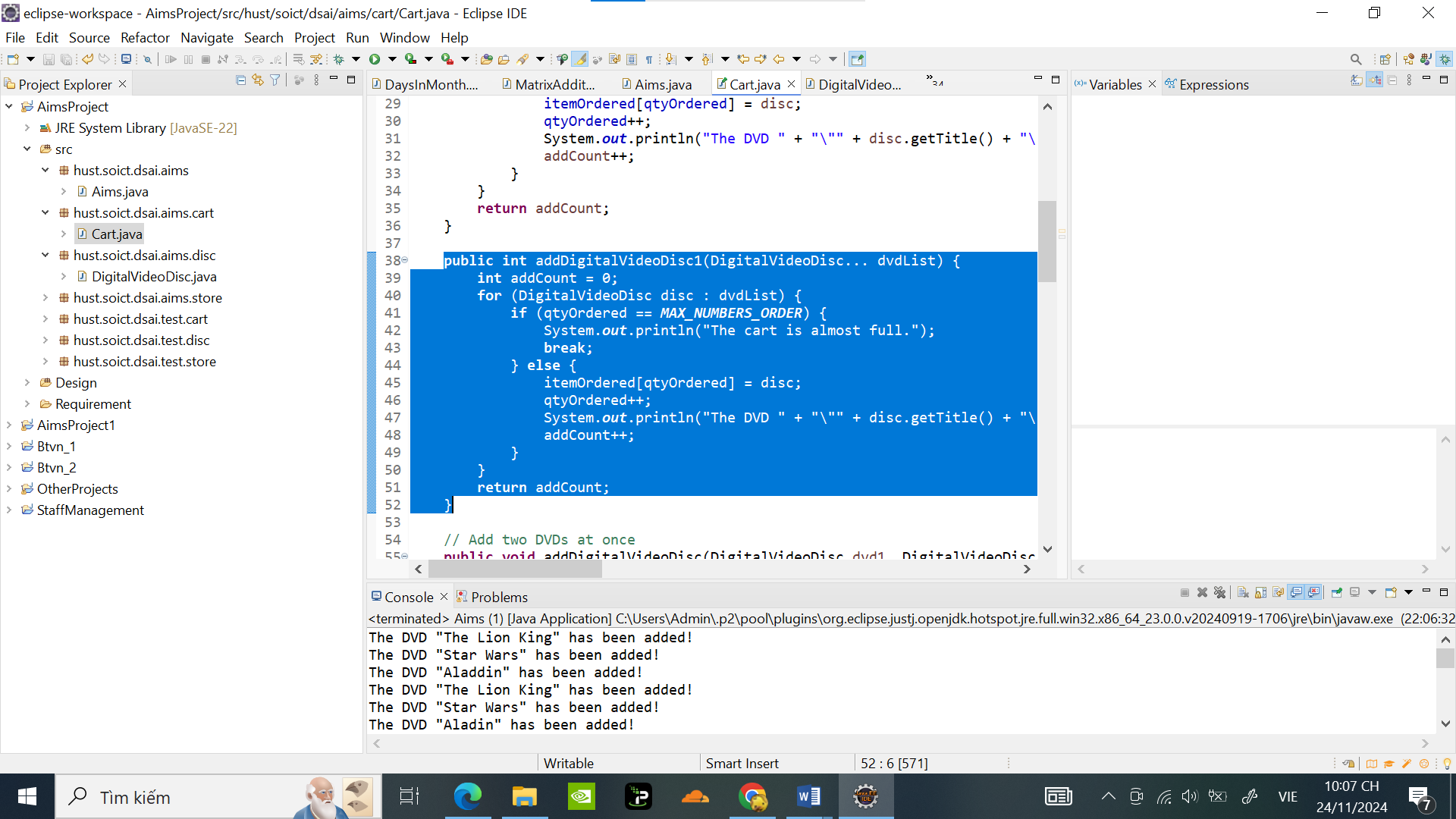
addCount++;

}

}

**return** addCount;

}



Chạy trong Aims:

Code:

**public** **class** Aims {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

//Câu lệnh trong class Cart

Cart anOrder = **new** Cart();

DigitalVideoDisc[] dvd = **new** DigitalVideoDisc[3];

dvd[0] = **new** DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f);

dvd[1] = **new** DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f);

dvd[2] = **new** DigitalVideoDisc("Aladdin", "Animation",**null**, 0, 18.99f);

DigitalVideoDisc dvd3 = **new** DigitalVideoDisc("The Lion King 2", "Animation", "Roger Allers", 87, 19.95f);

DigitalVideoDisc dvd4 = **new** DigitalVideoDisc("Star Wars the next day", "Science Fiction", "George Lucas", 87, 24.95f);

DigitalVideoDisc dvd5 = **new** DigitalVideoDisc("Aladin vs Superman", "Animation","", 0, 18.99f);

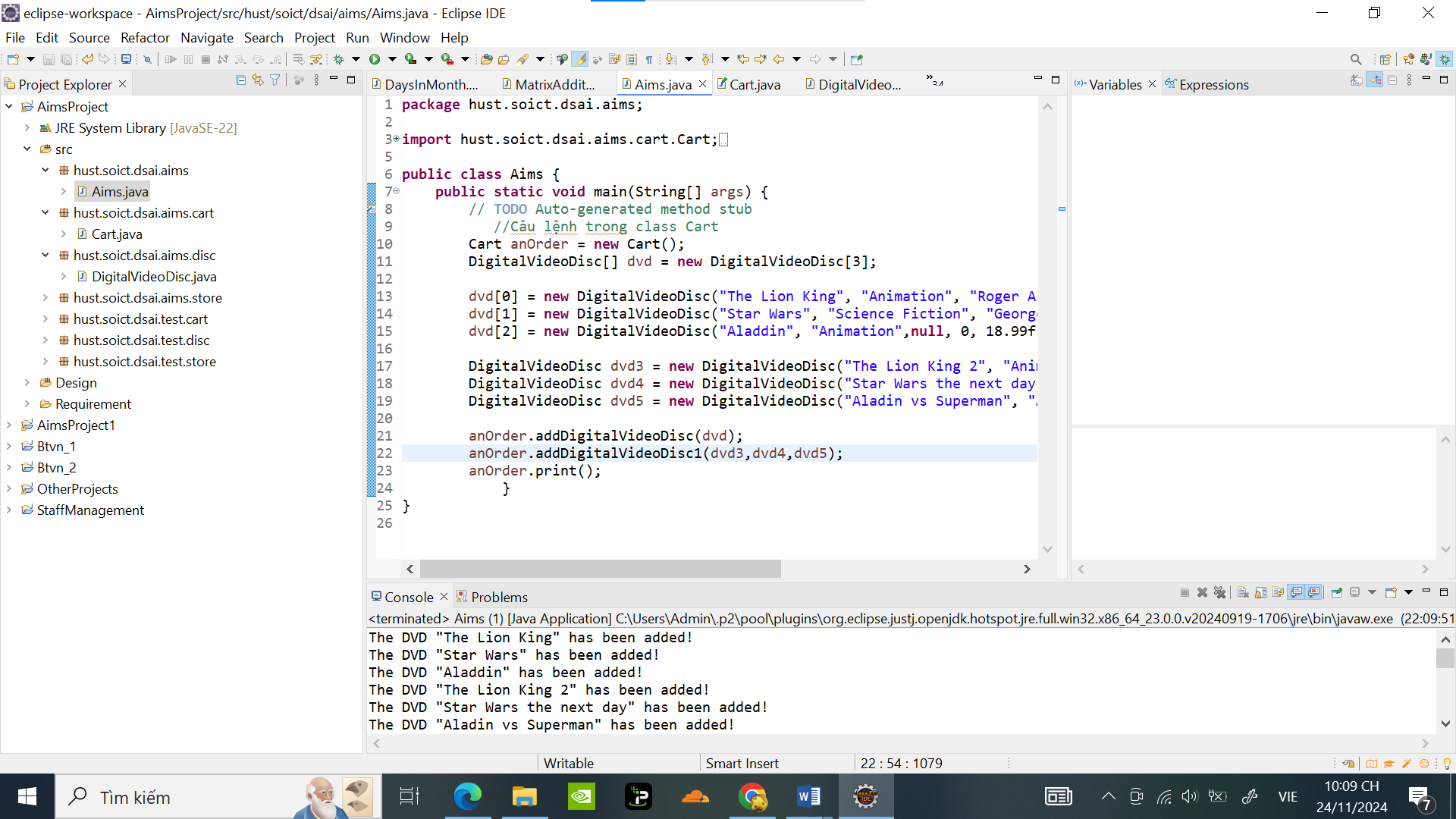
anOrder.addDigitalVideoDisc(dvd);

anOrder.addDigitalVideoDisc1(dvd3,dvd4,dvd5);

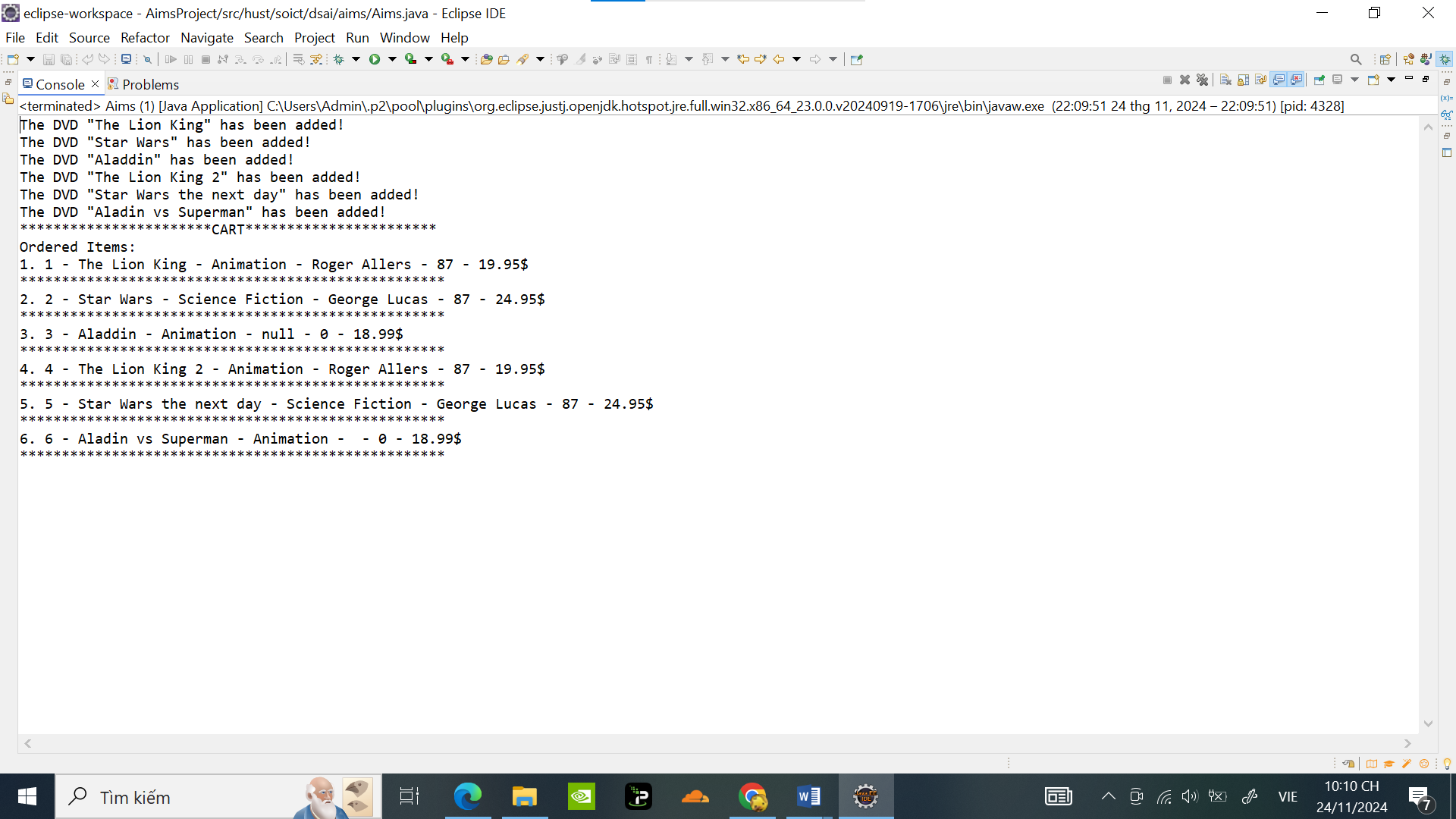
anOrder.print();

}

}



Kết quả :



Nhận xét: Cách sử dụng varargs sẽ tốt hơn vì nó cho phép thêm bất kỳ số lượng đĩa DVD nào mà không cần chỉ định mảng một cách rõ ràng, chúng ta có thể sử dụng danh sách tham số, hoặc là sử dụng mảng. Do đó hàm sẽ linh động hơn. Điều này làm cho mã nguồn sạch sẽ hơn và thuận tiện hơn khi gọi phương thức.

2**.2. Overloading by differing the number of parameter**

Code :

**public** **int** addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {

System.***out***.println("Add 2 discs");

**if** (qtyOrdered + 1 > ***MAX\_NUMBERS\_ORDER***) {

System.***out***.println(" The cart is almost full. Can't add more discs");

**return** 0;

} **else** {

itemOrdered[qtyOrdered] = dvd1;

qtyOrdered++;

System.***out***.println("The DVD " + "\"" + dvd1.getTitle() + "\"" + " has been added!");

itemOrdered[qtyOrdered] = dvd2;

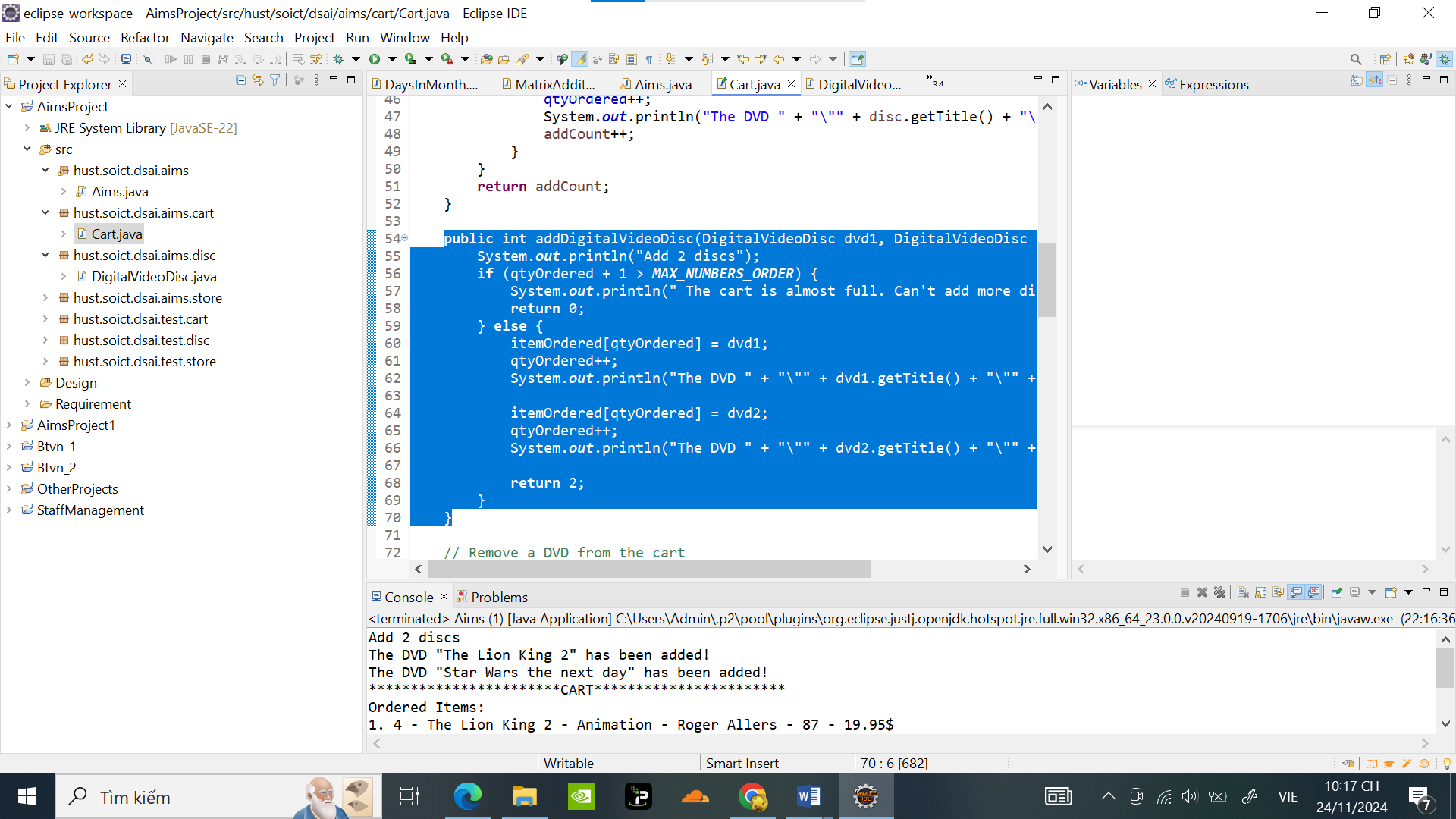
qtyOrdered++;

System.***out***.println("The DVD " + "\"" + dvd2.getTitle() + "\"" + " has been added!");

**return** 2;

}

}



Chạy trong Aims:

Code:

**public** **class** Aims {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

//Câu lệnh trong class Cart

Cart anOrder = **new** Cart();

DigitalVideoDisc[] dvd = **new** DigitalVideoDisc[3];

dvd[0] = **new** DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f);

dvd[1] = **new** DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f);

dvd[2] = **new** DigitalVideoDisc("Aladdin", "Animation",**null**, 0, 18.99f);

DigitalVideoDisc dvd3 = **new** DigitalVideoDisc("The Lion King 2", "Animation", "Roger Allers", 87, 19.95f);

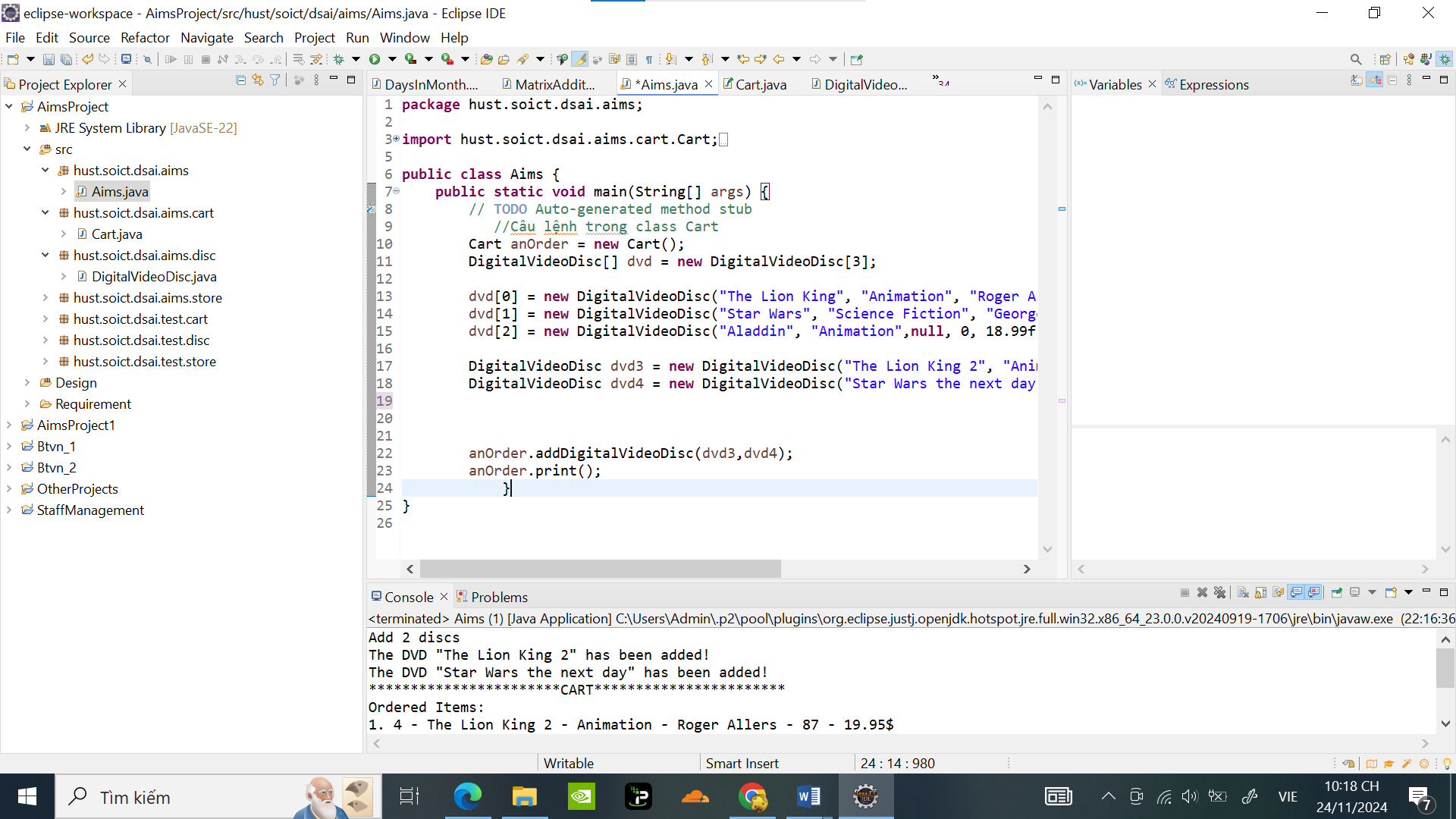
DigitalVideoDisc dvd4 = **new** DigitalVideoDisc("Star Wars the next day", "Science Fiction", "George Lucas", 87, 24.95f);

anOrder.addDigitalVideoDisc(dvd3,dvd4);

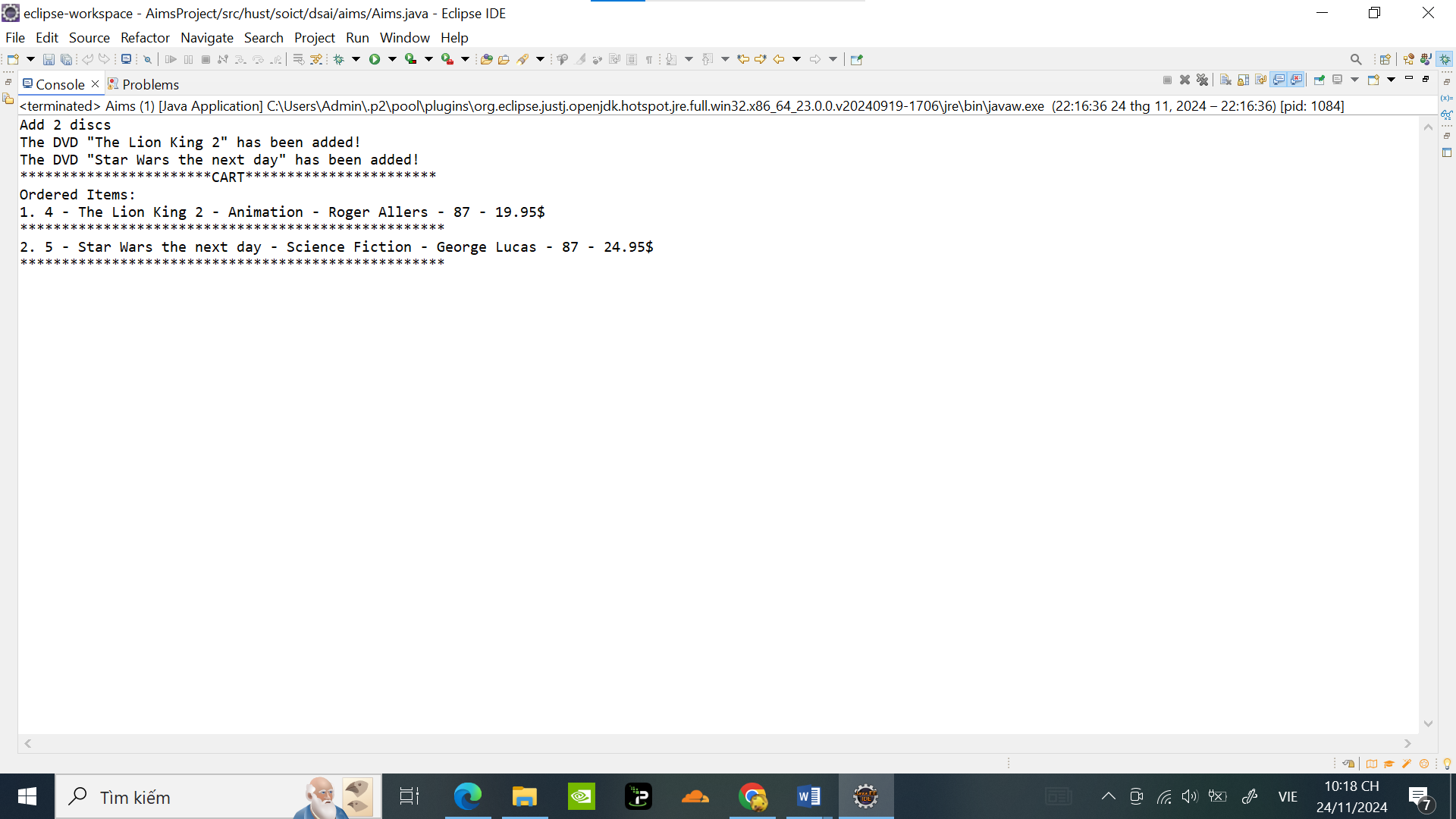
anOrder.print();

}

}



Kết quả :



1. Passing parameter

Code ban đầu :

**package** hust.soict.dsai.test.disc;

**import** hust.soict.dsai.aims.disc.DigitalVideoDisc;

**public** **class** TestPassingParameter {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

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) {

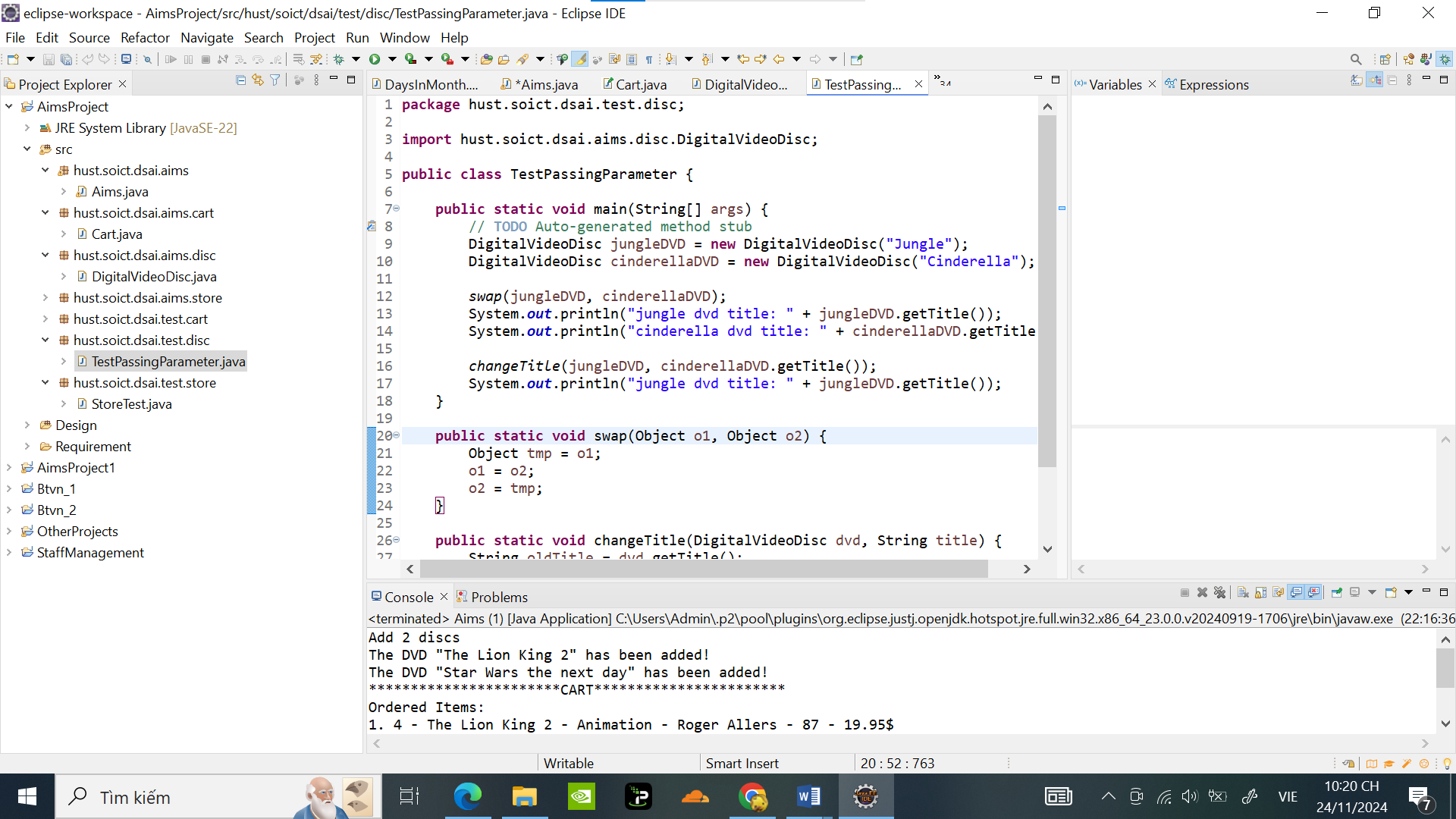
String oldTitle = dvd.getTitle();

dvd.setTitle(title);

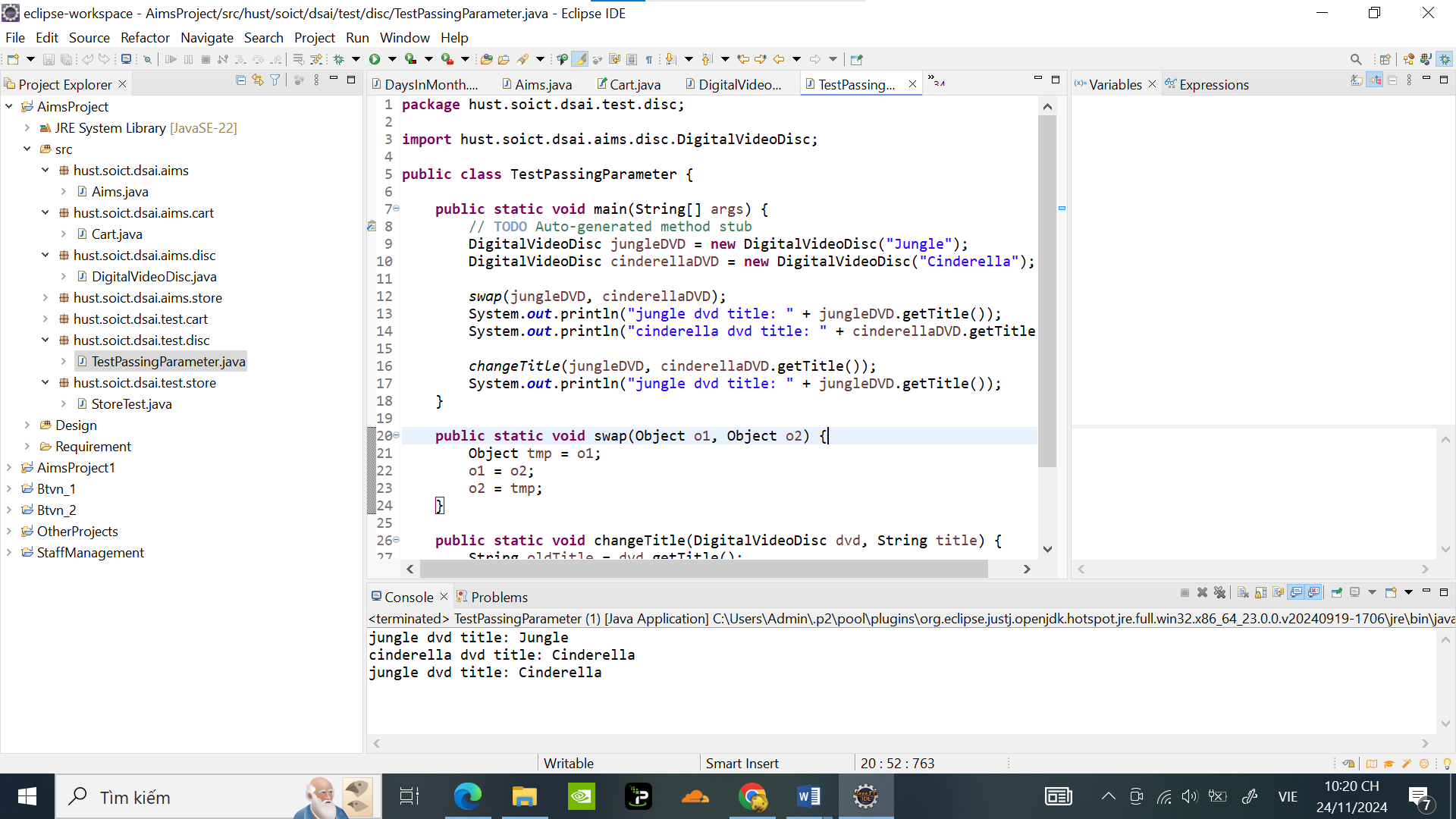
dvd = **new** DigitalVideoDisc(oldTitle);

}

}



Kết quả :



Ta có thể thấy rằng: Hàm swap đã không hoán đổi giá trị hai biến cho nhau.

**Sửa lại code sử dụng DEBUG ở 4.USE DEBUG RUN :**

**package** hust.soict.dsai.test.disc;

**import** hust.soict.dsai.aims.disc.DigitalVideoDisc;

**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("HieuDHM - Jungle dvd title: " + jungleDVD.getTitle());

}

**public** **static** **void** swap(DigitalVideoDisc dvd1,DigitalVideoDisc dvd2) {

DigitalVideoDisc tmp = **new** DigitalVideoDisc(

dvd1.getTitle(), dvd1.getCategory(),

dvd1.getDirector(), dvd1.getLength(), dvd1.getCost());

dvd1.setTitle(dvd2.getTitle());

dvd1.setCategory(dvd2.getCategory());

dvd1.setDirector(dvd2.getDirector());

dvd1.setLength(dvd2.getLength());

dvd1.setCost(dvd2.getCost());

dvd2.setTitle(tmp.getTitle());

dvd2.setCategory(tmp.getCategory());

dvd2.setDirector(tmp.getDirector());

dvd2.setLength(tmp.getLength());

dvd2.setCost(tmp.getCost());

}

**public** **static** **void** changeTitle(DigitalVideoDisc dvd, String title) {

String oldTitle = dvd.getTitle();

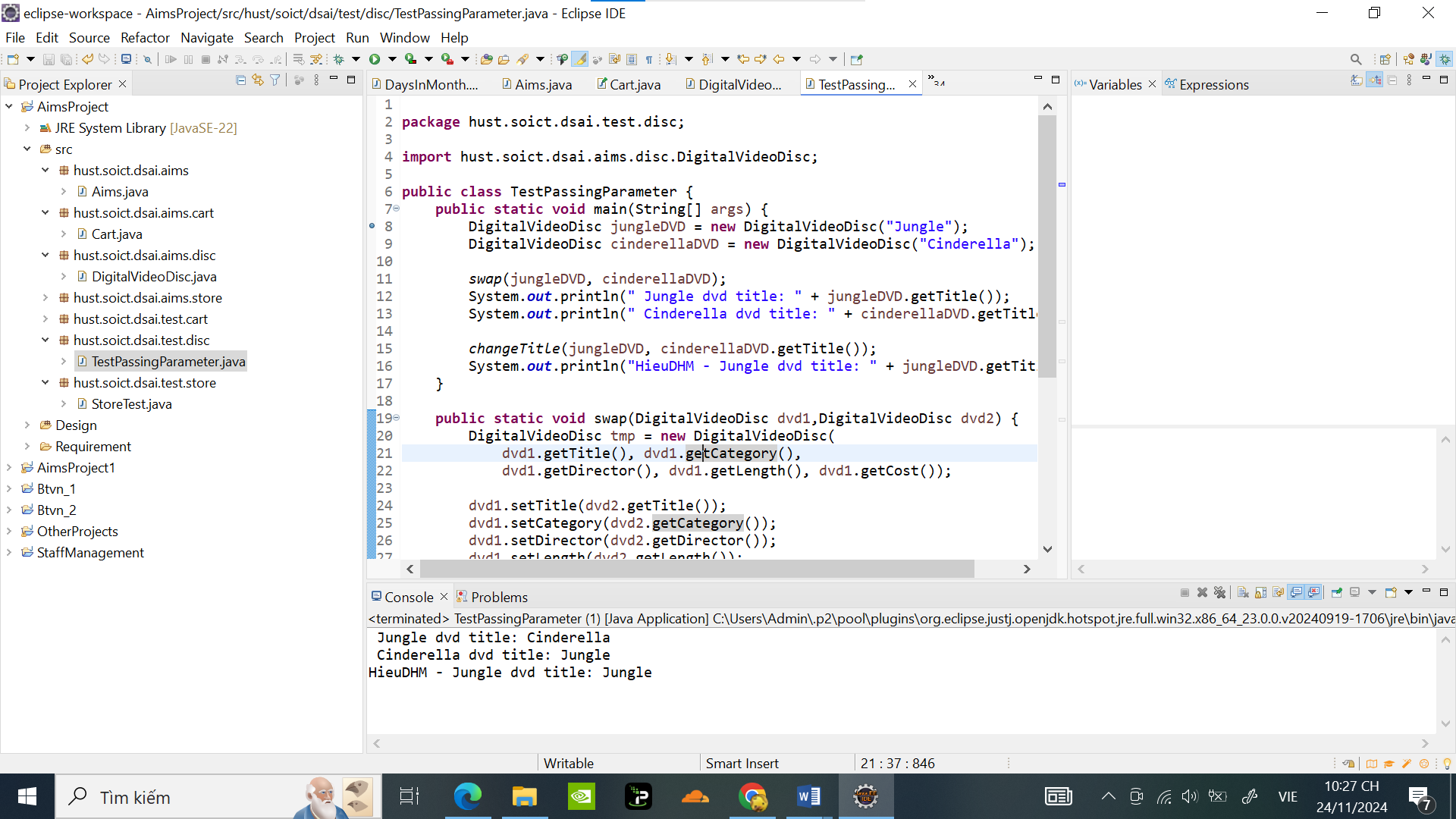
dvd.setTitle(title);

dvd = **new** DigitalVideoDisc(oldTitle);

}

}

Kết quả :



5 . Classifier Member and Instance Member

Code:

**package** hust.soict.dsai.aims.disc;

**public** **class** DigitalVideoDisc {

**private** String title;

**private** **int** length;

**private** String director;

**private** String category;

**private** **float** cost;

**private** **int** id;

**private** **static** **int** *nbDigitalVideoDiscs* = 0;

**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*;

}

**public** DigitalVideoDisc(String title) {

**this**.title = title;

**this**.category = "";

**this**.director = "";

**this**.length = 0;

**this**.cost = 0.0f;

// Tăng số lượng DVD và gán một ID duy nhất

*nbDigitalVideoDiscs*++;

**this**.id = *nbDigitalVideoDiscs*;

}

**public** String getTitle() {

**return** title;

}

**public** **int** getLength() {

**return** length;

}

**public** String getDirector() {

**return** director;

}

**public** String getCategory() {

**return** category;

}

**public** **float** getCost() {

**return** cost;

}

**public** **int** getId() {

**return** id;

}

**public** **void** setTitle(String titleSet) {

**this**.title = titleSet;

}

**public** **void** setDirector(String directorSet) {

**this**.director = directorSet;

}

**public** **void** setCategory(String categorySet) {

**this**.category = categorySet;

}

**public** **void** setLength(**int** lengthSet) {

**this**.length = lengthSet;

}

**public** **void** setCost(**float** costSet) {

**this**.cost = costSet;

}

**public** **void** setid(**int** idSet) {

**this**.id = idSet;

}

**public** **static** **int** getNbDigitalVideoDiscs() {

**return** *nbDigitalVideoDiscs*;

}

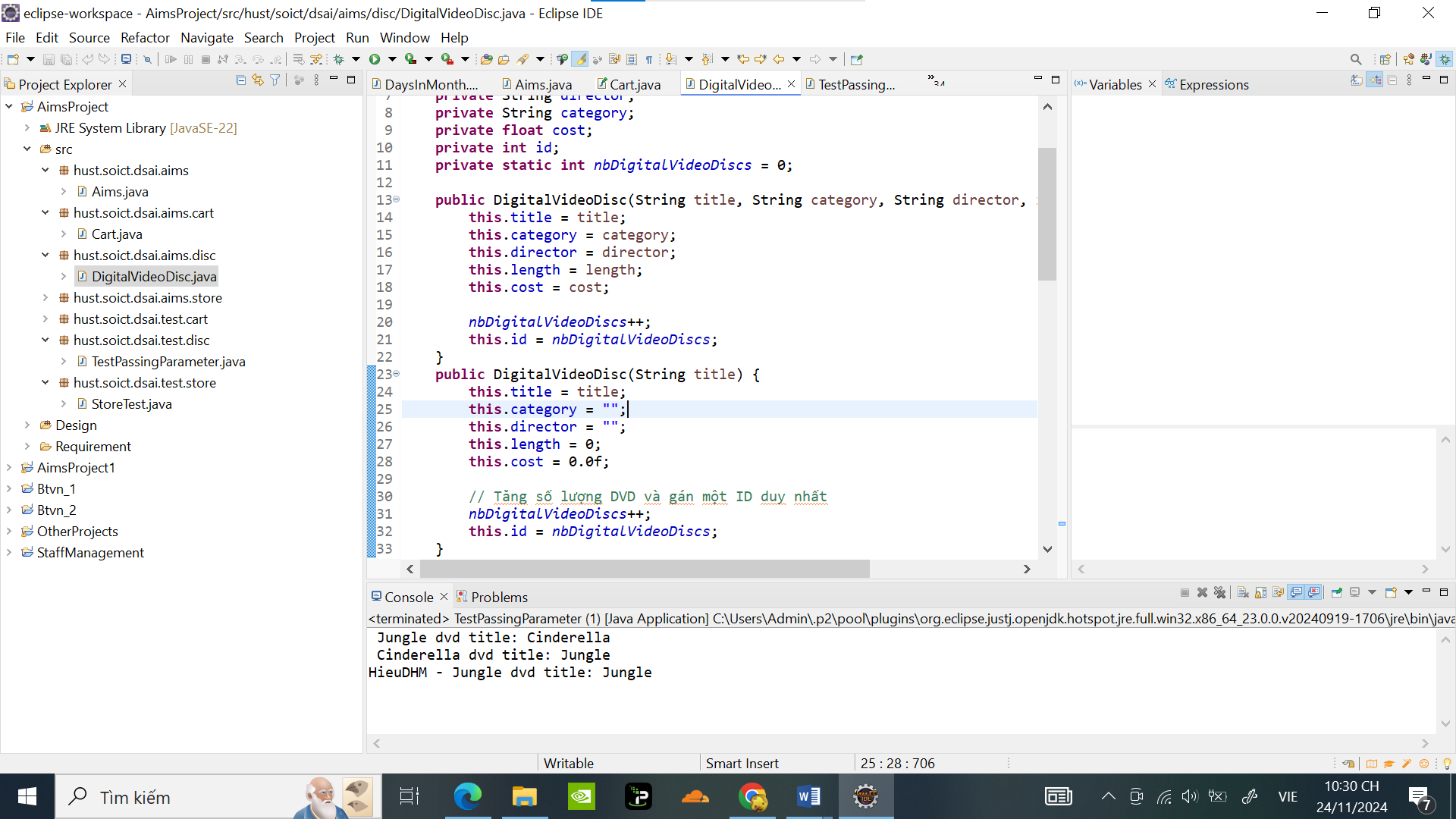
**public** **boolean** isMatch(String title2) {

// **TODO** Auto-generated method stub

**return** **false**;

}

}



**6. . Open the Cart class**

Code:

**package** hust.soict.dsai.test.cart;

**import** hust.soict.dsai.aims.cart.Cart;

**import** hust.soict.dsai.aims.disc.DigitalVideoDisc;

**public** **class** CartTest {

**public** **static** **void** main(String[] args) {

Cart cart = **new** Cart();

DigitalVideoDisc dvd1 = **new** DigitalVideoDisc("The Lion King", "Adventure", "John Doe", 120, 15.99f);

DigitalVideoDisc dvd2 = **new** DigitalVideoDisc("Star Wars", "Animation", "Jane Smith", 90, 12.99f);

DigitalVideoDisc dvd3 = **new** DigitalVideoDisc("Alladin", "Action", "Joe Russo", 180, 19.99f);

// Add DVDs to the cart

cart.addDigitalVideoDisc(dvd1);

cart.addDigitalVideoDisc(dvd2);

cart.addDigitalVideoDisc(dvd3);

// Print the cart

cart.print();

// Search by title

cart.searchByTitle("Solomon");

cart.searchByTitle("Frozen");

// Search by ID

cart.searchById(2);

cart.searchById(10);

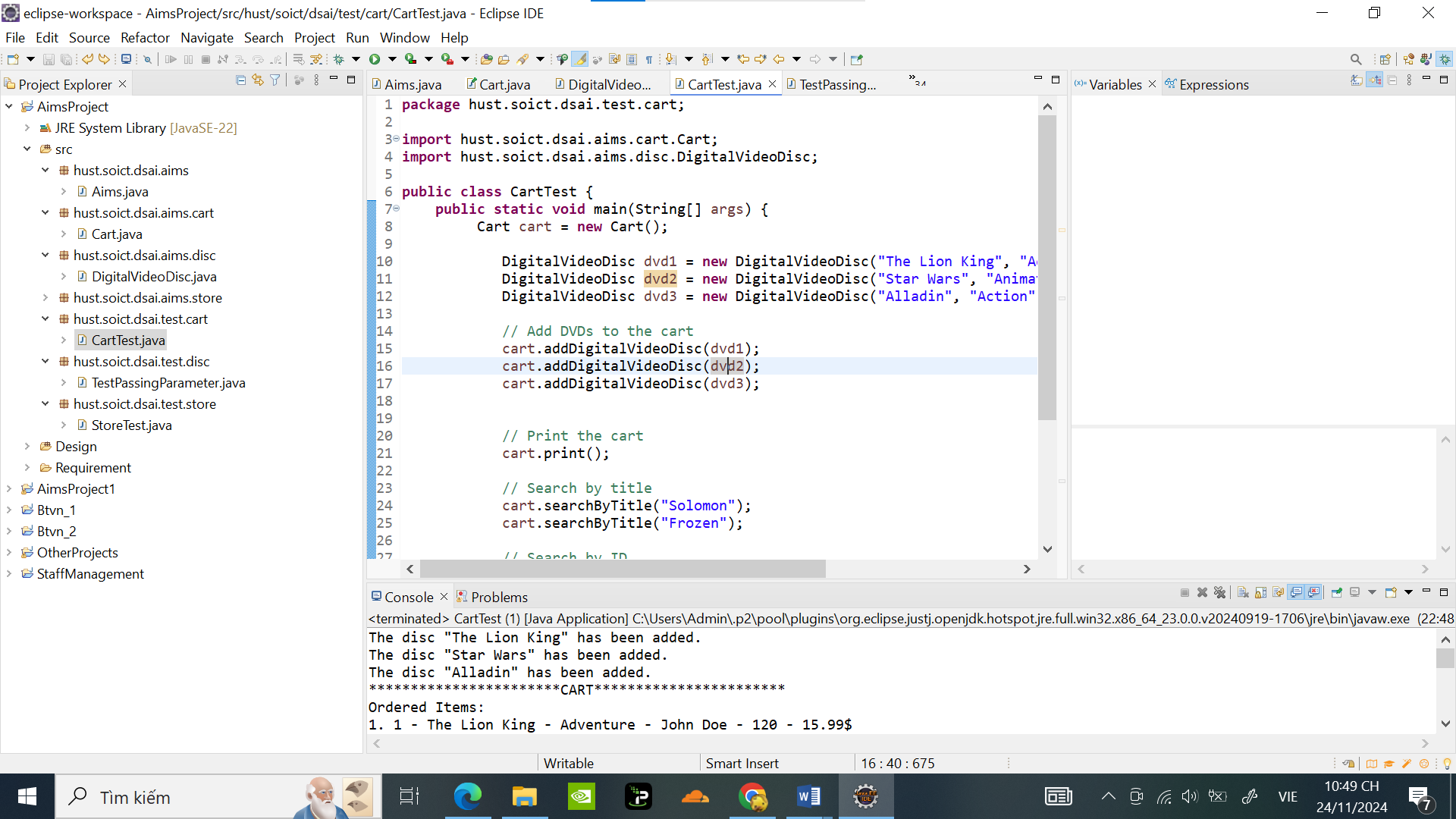
// Remove a DVD

cart.removeDigitalVideoDisc(dvd2);

cart.print();

}

}





**7. Implement the Store class**

Code:

package hust.soict.dsai.aims.store;

import java.util.LinkedList;

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

public class Store {

private LinkedList<DigitalVideoDisc> itemsInStore;

public Store() {

itemsInStore = new LinkedList<>();

}

boolean checkDVD(DigitalVideoDisc disc) {

return itemsInStore.contains(disc);

}

public void removeDVD(DigitalVideoDisc disc) {

if (checkDVD(disc)) {

itemsInStore.remove(disc);

System.out.println("DVD removed from the store.");

} else {

System.out.println("DVD not found in the store.");

}

}

public void addDVD(DigitalVideoDisc disc) {

if (!checkDVD(disc)) {

itemsInStore.add(disc);

System.out.println("DVD added to the store.");

} else {

System.out.println("DVD already exists in the store.");

}

}

@Override

public String toString() {

StringBuilder storeContents = new StringBuilder("Store contents:\n");

for (DigitalVideoDisc disc : itemsInStore) {

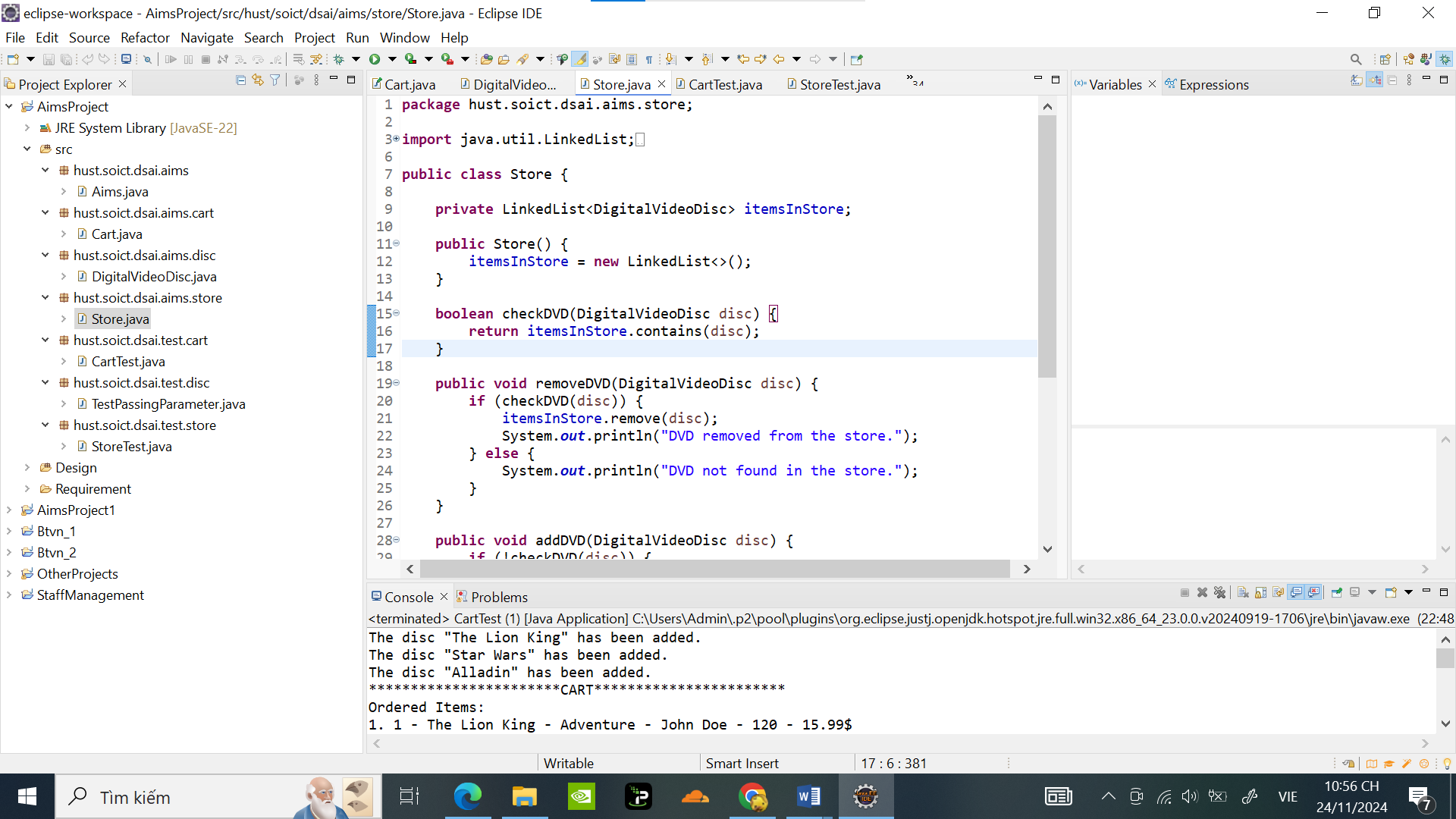
storeContents.append(disc.getTitle()).append("\n");

}

return storeContents.toString();

}

}



**Store test :**

package hust.soict.dsai.test.store;

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

import hust.soict.dsai.aims.store.Store;

public class StoreTest {

public static void main(String[] args) {

// Create a Store instance

Store store = new Store();

// Create some DigitalVideoDisc instances

DigitalVideoDisc dvd1 = new DigitalVideoDisc("Jungle", "Adventure", "John Doe", 120, 15.99f);

DigitalVideoDisc dvd2 = new DigitalVideoDisc("Cinderella", "Animation", "Jane Smith", 90, 12.99f);

DigitalVideoDisc dvd3 = new DigitalVideoDisc("Avengers", "Action", "Joe Russo", 180, 19.99f);

// Test adding DVDs to the store

System.out.println("Adding DVDs to the store...");

store.addDVD(dvd1);

store.addDVD(dvd2);

store.addDVD(dvd3);

// Attempt to add a duplicate DVD

System.out.println("\nAttempting to add a duplicate DVD...");

store.addDVD(dvd1);

// Display the DVDs in the store

System.out.println("\nStore contents after adding DVDs:");

System.out.println(store);

// Test removing a DVD from the store

System.out.println("Removing DVD 'Cinderella' from the store...");

store.removeDVD(dvd2);

// Display the DVDs in the store after removal

System.out.println("\nStore contents after removing 'Cinderella':");

System.out.println(store);

// Attempt to remove a DVD not in the store

System.out.println("Attempting to remove a DVD not in the store...");

DigitalVideoDisc nonExistentDVD = new DigitalVideoDisc("Frozen", "Animation", "Chris Buck", 90, 13.99f);

store.removeDVD(nonExistentDVD);

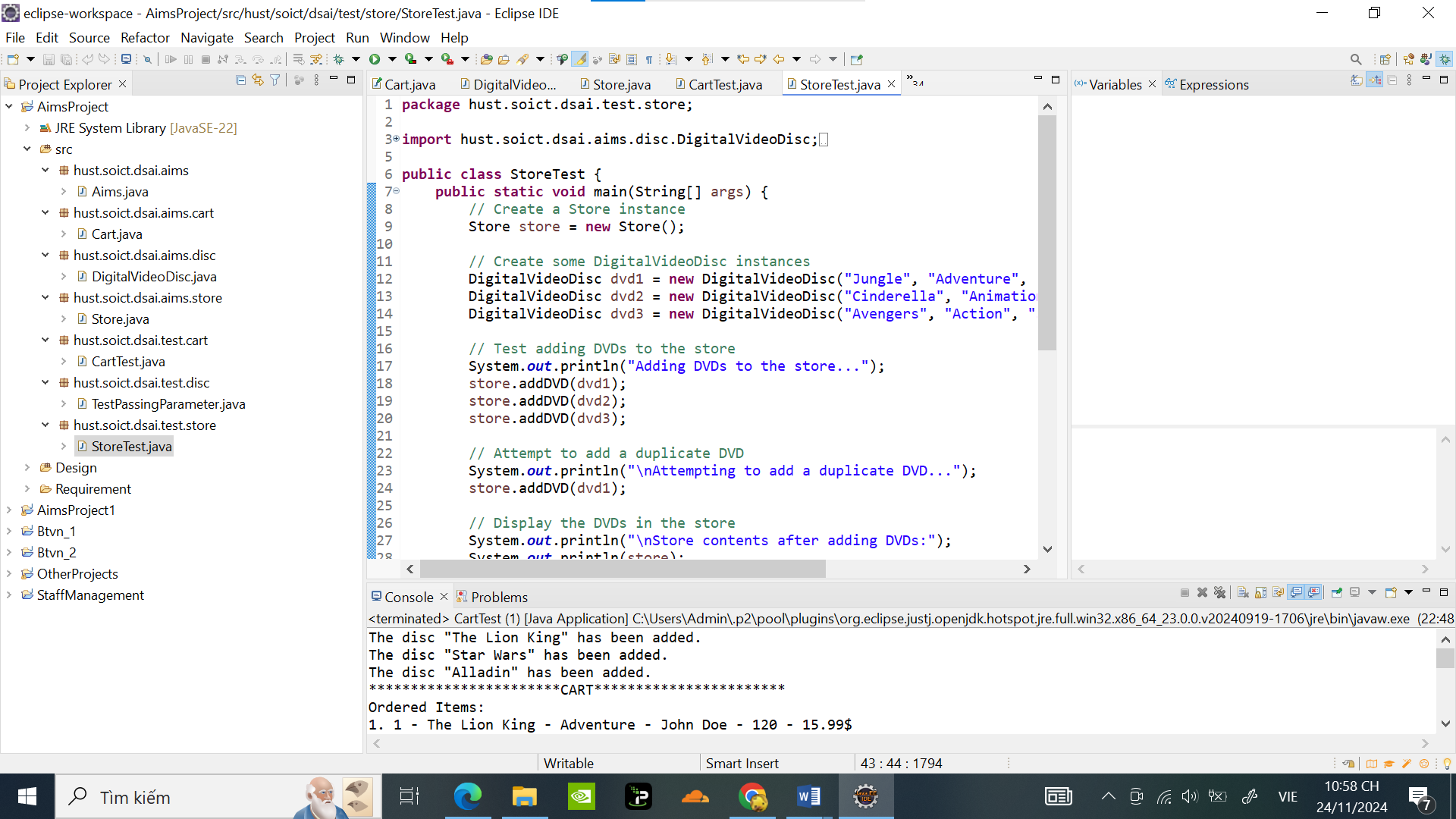
// Display the final store contents

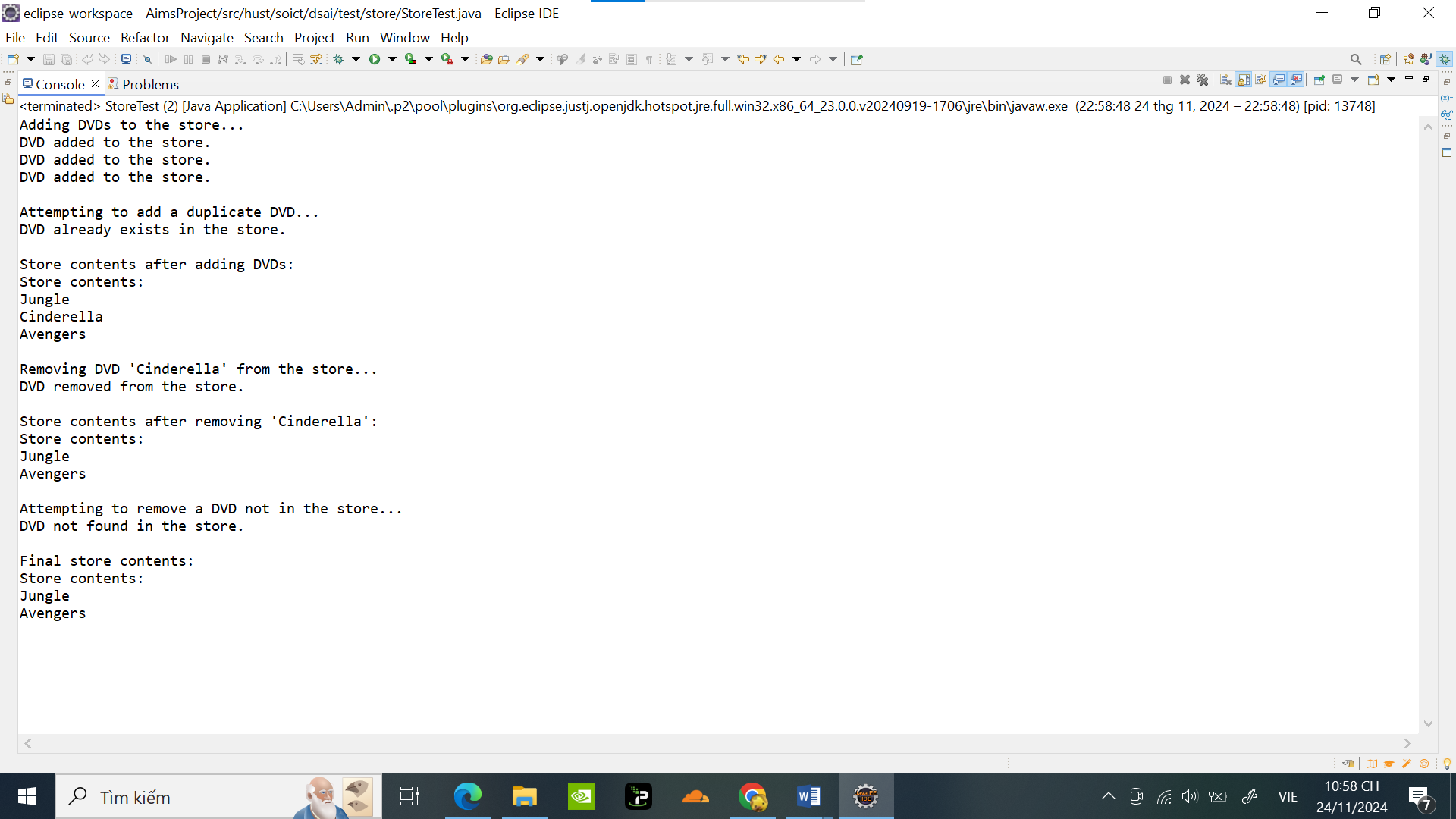
System.out.println("\nFinal store contents:");

System.out.println(store);

}

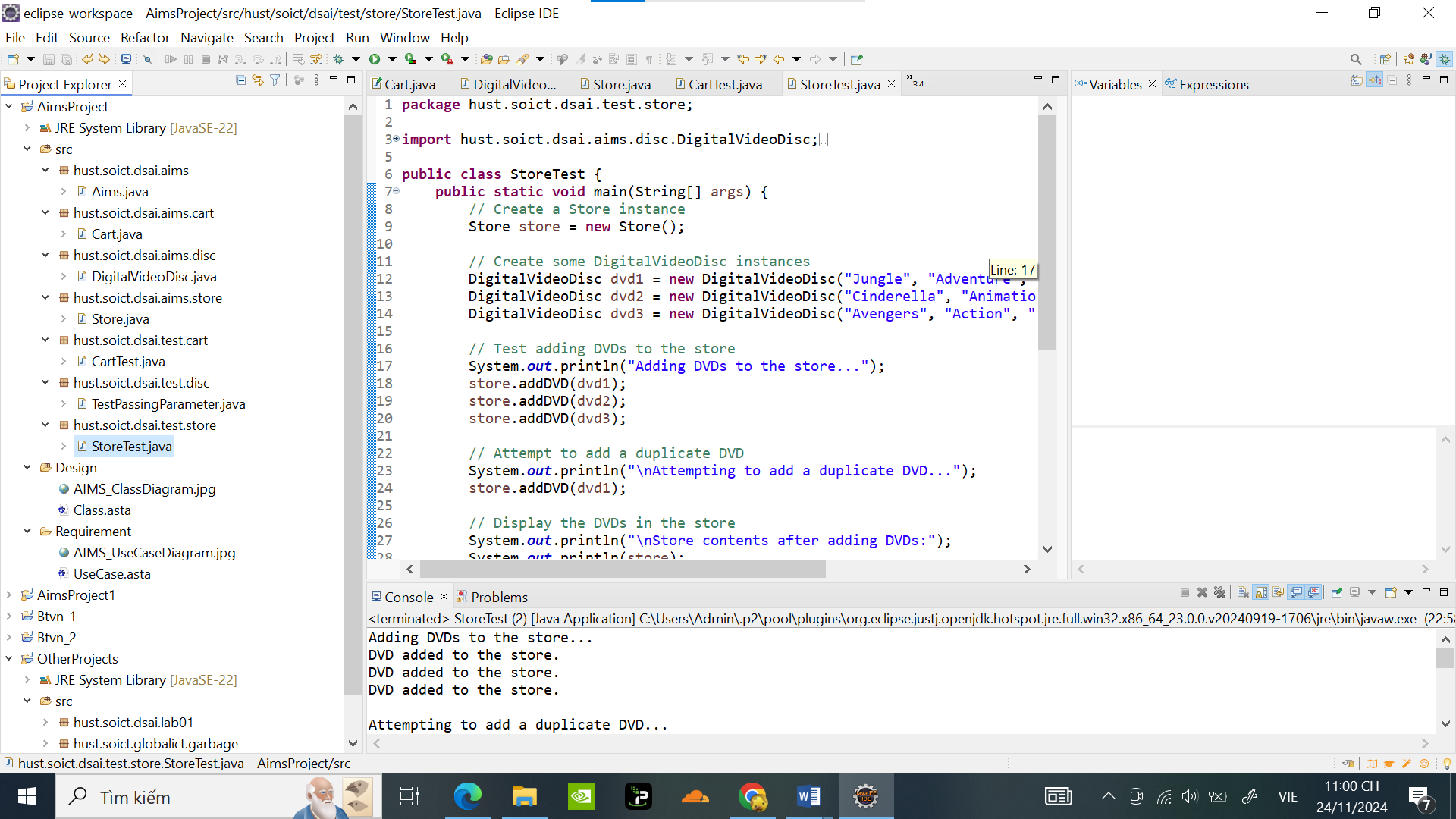
}





**8. Re-organize your projects**

Đã đổi tên như hình ảnh :



**9. . String, StringBuilder and StringBuffer**

**9.1 ConcatenationInLoops:**

**package** hust.soict.globalict.garbage;

**import** java.util.Random;

**public** **class** ConcatenationInLoops {

**public** **static** **void** main(String[] args) {

Random r = **new** Random(123);

**long** start = System.*currentTimeMillis*();

@SuppressWarnings("unused")

String s = "";

**for** (**int** i = 0; i < 65536; i++) {

s += r.nextInt(2);

}

System.***out***.println(System.*currentTimeMillis*() - start); // This prints roughly 500.

r = **new** Random(123);

start = System.*currentTimeMillis*();

StringBuilder sb = **new** StringBuilder();

**for** (**int** i = 0; i < 65536; i++) {

sb.append(r.nextInt(2));

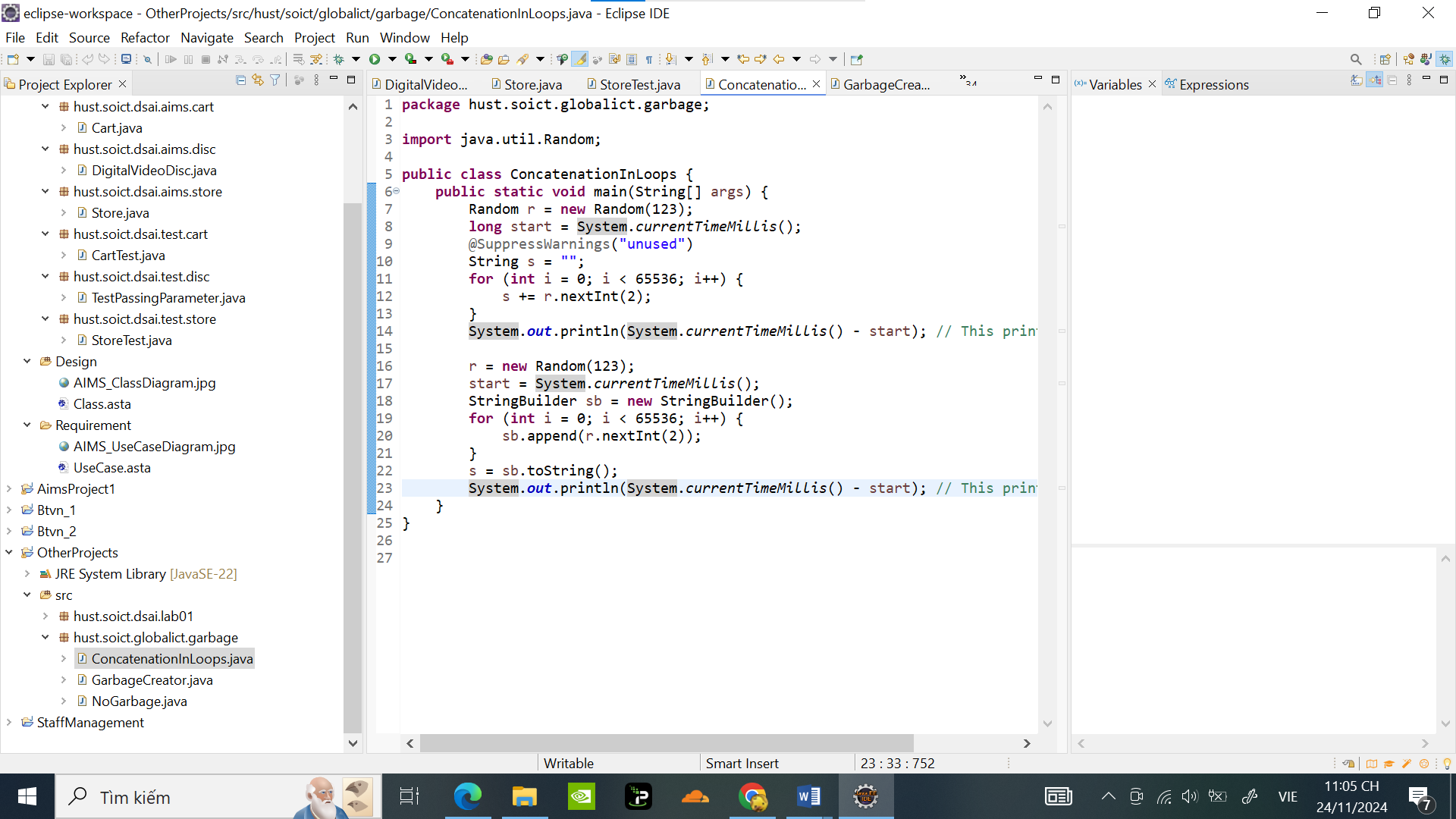
}

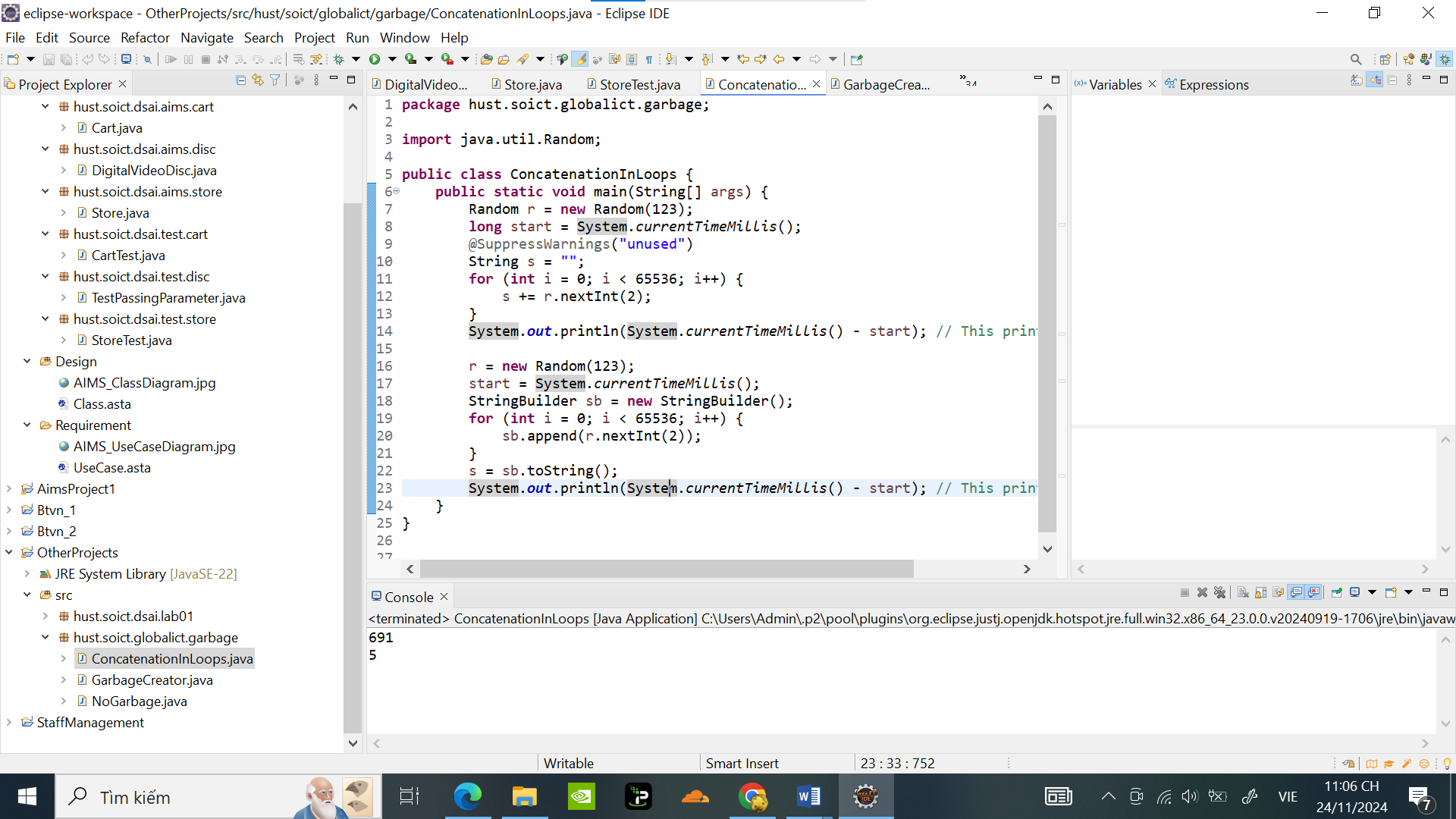
s = sb.toString();

System.***out***.println(System.*currentTimeMillis*() - start); // This prints 5.

}

}





**9.2 GarbageCreator khi sử dụng đọc file large\_text\_file.txt:**

package hust.soict.globalict.garbage;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

public class GarbageCreator {

public static void main(String[] args) throws IOException {

String filePath = "D:\\Downloads\\large\_text\_file.txt"; // Replace with the path to your large file

byte[] inputBytes = { 0 };

long startTime, endTime;

inputBytes = Files.readAllBytes(Paths.get(filePath));

startTime = System.currentTimeMillis();

@SuppressWarnings("unused")

String outputString = "";

for (byte b : inputBytes) {

outputString += (char) b;

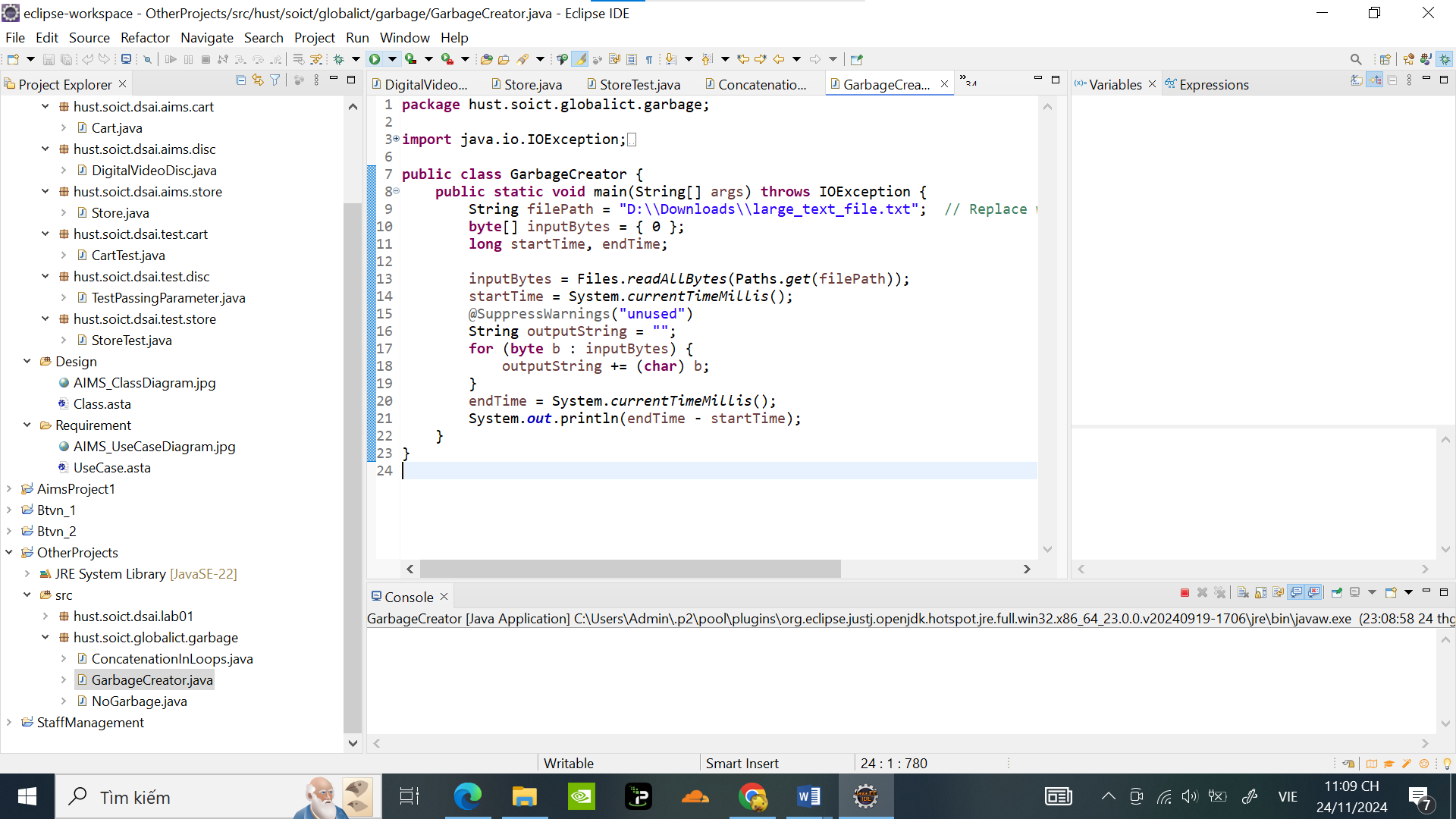
}

endTime = System.currentTimeMillis();

System.out.println(endTime - startTime);

}

}



Tốc độ đọc file rất là chậm nên gần như bị đơ

**9.3 NoGarbage khi sử dụng đọc file large\_text\_file.txt:**

package hust.soict.globalict.garbage;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

public class NoGarbage {

public static void main(String[] args) throws IOException {

String filePath = "D:\\Downloads\\large\_text\_file.txt"; // Replace with the path to your large file

byte[] inputBytes = { 0 };

long startTime = 0, endTime;

inputBytes = Files.readAllBytes(Paths.get(filePath));

StringBuilder outputStringBuilder = new StringBuilder();

for (byte b : inputBytes) {

outputStringBuilder.append((char)b);

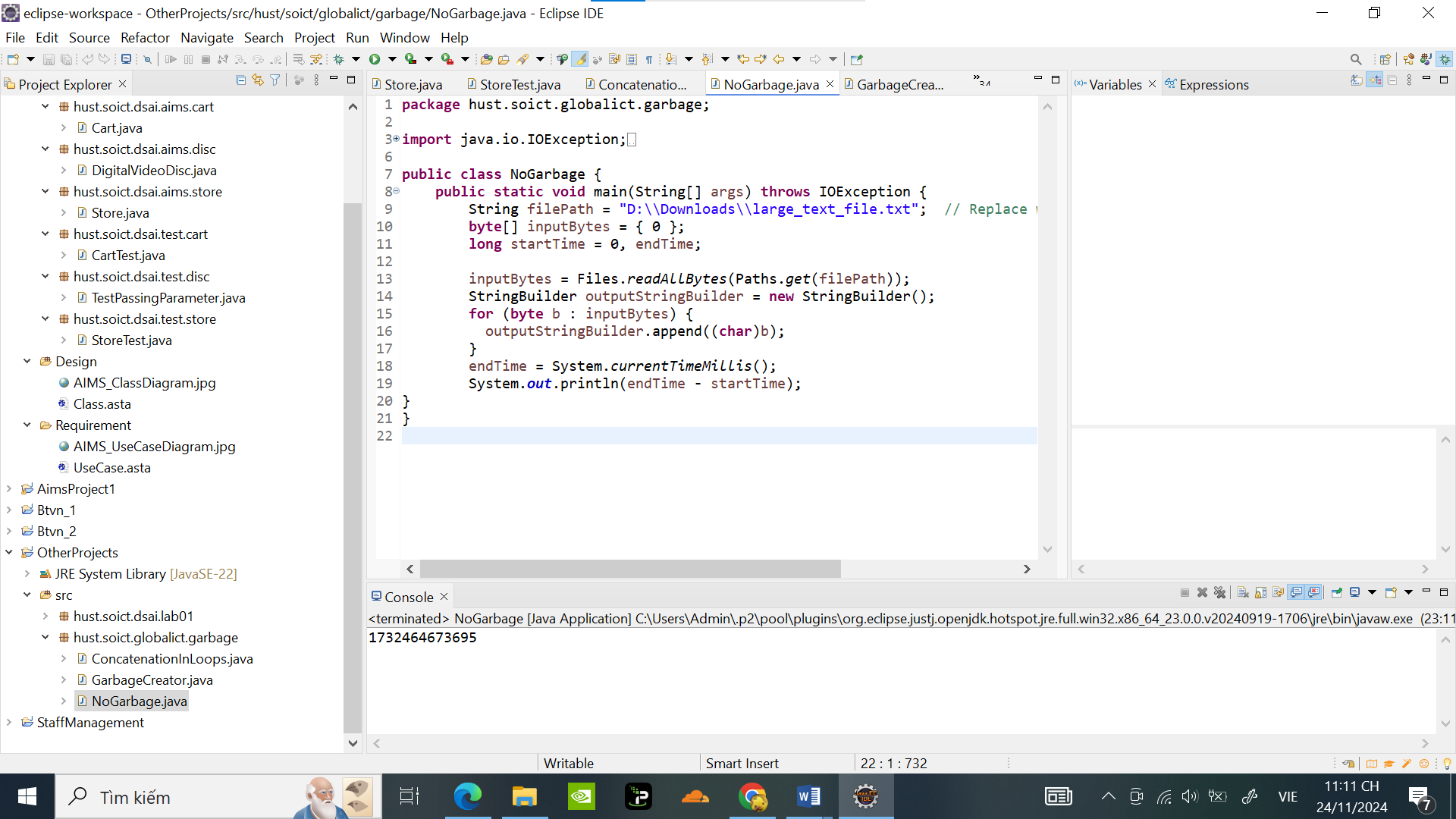
}

endTime = System.currentTimeMillis();

System.out.println(endTime - startTime);

}

}



Ta có thể thấy tốc độ đọc file của NoGarbage chỉ trong 1 giây khi thay đổi + thành String và StringBuilder khiêns cho quá trình đọc file hiệu quả hơn

**10. Release flow demonstration**

Đã tạo trên Github