

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



BÁO CÁO THỰC HÀNH
IT3103-744529-2024.1

BÀI THỰC HÀNH 3

Họ và tên sinh viên: Đoàn Thanh Tùng

MSSV: 20225946

Lớp: Việt Nhật 02 – K67

GVHD: Lê Thị Hoa

HTGD: Bùi Trọng Dũng

Hà Nội 10/2024

BÁO CÁO THỰC HÀNH LAB 03

THỰC HÀNH LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Contents

1. Working with method overloading.....	2
1.1. Overloading by differing types of parameter	2
Code	2
Result.....	2
1.2. Overloading by differing the number of parameters	3
Code	3
Result.....	5
2. Passing parameter	6
Code	6
Result.....	6
3. Class Member and Instance Member.....	7
Code	7
Result.....	9
4. Open the Cart class.....	9
Code	9
Result.....	11
5. Implement the Store class.....	12
Code	12
Result.....	13
6. String, StringBuilder and StringBuffer.....	14
Code	14
Result.....	14
7 Class Diagram	16
Table of Figures	17

1. Working with method overloading

1.1. Overloading by differing types of parameter

Code

```
// Doan Thanh Tung 20225946
public void addDigitalVideoDisc(DigitalVideoDisc[] dvdList) {
    for (int i = 0; i < dvdList.length; i++) {
        if (qtyOrdered < MAX_NUMBERS_ORDERED) {
            itemsOrdered[qtyOrdered++] = dvdList[i];
            System.out.println(dvdList[i].getTitle() + " is added sucessfully");
        } else {
            System.out.println("the cart is full");
            break;
        }
    }
    return;
}
```

Figure 1 Method addDigitalVideoDisc(DigitalVideoDisc [] dvdList)

```
3 // Doan Thanh Tung - 20225946
4 public class Aims {
5     public static void main(String args[]) {
6         Cart order = new Cart();
7         DigitalVideoDisc dvd1 = new DigitalVideoDisc("Doraemon the movie", "Anime", "Zen Soichiro", 90, 25.50f);
8         DigitalVideoDisc dvd2 = new DigitalVideoDisc("Spiderman no way home", "Science Fiction", "Jon Watts", 90,
9             30.15f);
10        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Naruto", "Anime", 21.30f);
11        DigitalVideoDisc dvd4 = new DigitalVideoDisc("Diary of a Wimpy Kid", "Comedy", "Thor Freudenthal", 40.50f);
12        DigitalVideoDisc[] dvdList = { dvd1, dvd2, dvd3, dvd4 };
13        order.addDigitalVideoDisc(dvdList);
14    }
15 }
16 }
17 }
```

Figure 2 Aims Class

Result

```
Doraemon the movie is added sucessfully
Spiderman no way home is added sucessfully
Naruto is added sucessfully
Diary of a Wimpy Kid is added sucessfully
```

Figure 3 Result

1.2. Overloading by differing the number of parameters

Code

```
//Doan Thanh Tung - 20225946
public void addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {
    if (qtyOrdered < MAX_NUMBERS_ORDERED - 2) {
        itemsOrdered[qtyOrdered++] = dvd1;
        itemsOrdered[qtyOrdered++] = dvd2;
        System.out.println(dvd1.getTitle() + " and " + dvd2.getTitle() + " has been added!");
        qtyOrdered++;
    } else
        System.out.println("The cart is full.");
}
```

Figure 4 Method addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2)

```
3 // Doan Thanh Tung - 20225946
4 public class Aims {
5     public static void main(String args[]) {
6         Cart order = new Cart();
7         DigitalVideoDisc dvd1 = new DigitalVideoDisc("Doraemon the movie", "Anime", "Zen Soichiro", 90, 25.50f);
8         DigitalVideoDisc dvd2 = new DigitalVideoDisc("Spiderman no way home", "Science Fiction", "Jon Watts", 90,
9             30.15f);
10        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Naruto", "Anime", 21.30f);
11        DigitalVideoDisc dvd4 = new DigitalVideoDisc("Diary of a Wimpy Kid", "Comedy", "Thor Freudenthal", 40.50f);
12        order.addDigitalVideoDisc(dvd1, dvd2);
13    }
14 }
15 }
16 }
```

Figure 5 Aims Class

Result

```
Doraemon the movie and Spiderman no way homehas been added!
```

Figure 6 Result

2. Passing parameter

Code

```
3 //Doan Thanh Tung 20225946
4 public class TestPassingParameter {
5
6     public static void main(String[] args) {
7         DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
8         DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderella");
9
10        swap(jungleDVD, cinderellaDVD);
11        System.out.println("Jungle DVD's title: " + jungleDVD.getTitle());
12        System.out.println("Cinderella DVD's title: " + cinderellaDVD.getTitle());
13
14        changeTitle(jungleDVD, cinderellaDVD.getTitle());
15        System.out.println("Jungle DVD's title: " + jungleDVD.getTitle());
16    }
17
18    public static void swap(DigitalVideoDisc o1, DigitalVideoDisc o2) {
19        String tmp = o1.getTitle();
20        o1.setTitle(o2.getTitle());
21        ;
22        o2.setTitle(tmp);
23    }
24
25    public static void changeTitle(DigitalVideoDisc dvd, String title) {
26        String oldTitle = dvd.getTitle();
27        dvd.setTitle(title);
28        dvd = new DigitalVideoDisc(oldTitle);
29    }
30
31 }
32
```

Figure 7 Passing parameter code

Result

```
Jungle DVD's title: Cinderella
Cinderella DVD's title: Jungle
Jungle DVD's title: Jungle
```

Figure 8 Passing parameter result

3. Class Member and Instance Member

Code

```
3 // Doan Thanh Tung - 20225946
4 public class DigitalVideoDisc {
5     private String title;
6     private String category;
7     private String director;
8     private float cost;
9     private int length;
10    private int id;
11    private static int nbDigitalVideoDiscs = 0;
12
13    public DigitalVideoDisc(String title) {
14        super();
15        this.title = title;
16        this.id = nbDigitalVideoDiscs++;
17    }
18
19    public DigitalVideoDisc(String title, String category, float cost) {
20        super();
21        this.title = title;
22        this.category = category;
23        this.cost = cost;
24        this.id = nbDigitalVideoDiscs++;
25    }
26}
```

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

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

Figure 9 Class member and Instance member

Result

```
3 // Doan Thanh Tung - 20225946
4 public class Aims {
5     public static void main(String args[]) {
6         Cart order = new Cart();
7         DigitalVideoDisc dvd1 = new DigitalVideoDisc("Doraemon the movie", "Anime", "Zen Soichiro", 90, 25.50f);
8         DigitalVideoDisc dvd2 = new DigitalVideoDisc("Spiderman no way home", "Science Fiction", "Jon Watts", 90,
9             30.15f);
10        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Naruto", "Anime", 21.30f);
11        DigitalVideoDisc dvd4 = new DigitalVideoDisc("Diary of a Wimpy Kid", "Comedy", "Thor Freudenthal", 40.50f);
12        order.addDigitalVideoDisc(dvd1);
13        System.out.println(dvd1.getId() + " (id)");
14        order.addDigitalVideoDisc(dvd2);
15        System.out.println(dvd2.getId() + " (id)");
16        order.addDigitalVideoDisc(dvd3);
17        System.out.println(dvd3.getId() + " (id)");
18        order.addDigitalVideoDisc(dvd4);
19        System.out.println(dvd4.getId() + " (id)");
20    }
21 }
```

Console ×

<terminated> Aims [Java Application] D:\cac loai files\sts-4.21.0.RELEASE\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.9.v20231028-0858\jre\bin\javaw.exe (Nov 24, 2024, 10:47:10 AM – 10:47:10 AM) [pid: 3028]

Doraemon the movie is added sucessfully
0 -id
Spiderman no way home is added sucessfully
1 -id
Naruto is added sucessfully
2 -id
Diary of a Wimpy Kid is added sucessfully
3 -id

Figure 10 Result

4. Open the Cart class

Code

```
//Doan Thanh Tung - 20225946
public void print() {
    System.out.println("\n*****CART*****");
    System.out.println("Ordered Items:");
    for (int i = 0; i < qtyOrdered; i++) {
        System.out.println(itemsOrdered[i].toString());
    }
    System.out.printf("Total cost: $%.2f\n", totalcost());
    System.out.println("*****");
}
```

Figure 11 Code method to print the content in Cart

```
//Doan Thanh Tung - 20225946
public void search(int id) {
    for (int i = 0; i < qtyOrdered; i++) {
        if (itemsOrdered[i].isMatch(id)) {
            System.out.println("\nSuccessfull");
            System.out.println(itemsOrdered[i].toString());
            return;
        }
    }
    System.out.println("\nNot Found!");
}
```

Figure 12 Code method search by id


```
//Doan Thanh Tung - 20225946|
public void search(String title) {
    for (int i = 0; i < qtyOrdered; i++) {
        if (itemsOrdered[i].isMatch(title)) {
            System.out.println("\nSuccessfull");
            System.out.println(itemsOrdered[i].toString());
            return;
        }
    }
    System.out.println("\nNot Found!");
}
}
```

Figure 13 Code method search by title

```
//Doan Thanh Tung - 20225946|
public boolean isMatch(int id) {
    return this.id == id;
}

public boolean isMatch(String title) {
    return this.title.toLowerCase().equals(title.toLowerCase());
}
}
```

Figure 14 IsMatch() in DigitalVideoDisc class

```
//Doan Thanh Tung - 20225946|
public class CartTest {
    public static void main(String args[]) {
        Cart order = new Cart();
        DigitalVideoDisc dvd1 = new DigitalVideoDisc("Doraemon the movie", "Anime", "Zen Soichiro", 90, 25.50f);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Spiderman no way home", "Science Fiction", "Jon Watts", 90, 30.15f);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Naruto", "Anime", "Hayato-date", 60, 21.30f);
        DigitalVideoDisc dvd4 = new DigitalVideoDisc("Diary of a Wimpy Kid", "Comedy", "Thor Freudenthal", 85, 40.15f);
        order.addDigitalVideoDisc(dvd1);
        order.addDigitalVideoDisc(dvd2);
        order.addDigitalVideoDisc(dvd3);
        order.addDigitalVideoDisc(dvd4);
        order.search("Doraemon the movie");
    }
}
```

Figure 15 CartTest class

Result

```
Doraemon the movie is added sucessfully
Spiderman no way home is added sucessfully
Naruto is added sucessfully
Diary of a Wimpy Kid is added sucessfully

*****CART*****
Ordered Items:
Doraemon the movie - Anime - Zen Soichiro - 90: 25.5$
Spiderman no way home - Science Fiction - Jon Watts - 90: 30.15$
Naruto - Anime - Hayato-date - 60: 21.3$
Diary of a Wimpy Kid - Comedy - Thor Freudenthal - 85: 40.5$
Total cost: $117.45
*****
```

Figure 16 Result

5. Implement the Store class

Code

```
//Doan Thanh Tung - 20225946
public class Store {
    private DigitalVideoDisc itemsInStore[];
    private int numberOfItems;

    • public Store(int maxSize) {
        itemsInStore = new DigitalVideoDisc[maxSize];
        numberOfItems = 0;
    }
    • public void addDVD(DigitalVideoDisc dvd) {
        if (numberOfItems < itemsInStore.length) {
            itemsInStore[numberOfItems++] = dvd;
            System.out.println(dvd.getTitle() + " is added successfully");
        } else {
            System.out.println("Max capacity!");
        }
    }
    • public void removeDVD(DigitalVideoDisc dvd) {
        for (int i = 0; i < numberOfItems; i++) {
            if (itemsInStore[i].equals(dvd)) {
                for (int j = i; j < numberOfItems - 1; j++) {
                    itemsInStore[j] = itemsInStore[j + 1];
                }
                numberOfItems--;
                System.out.println(dvd.getTitle() + " is removed successfully");
                return;
            }
        }
        System.out.println("Not found");
    }
}
```

Figure 17 Store Class

```
//Doan Thanh Tung - 20225946
public class StoreTest {
    public static void main(String[] args) {

        Store store = new Store(5);

        DigitalVideoDisc dvd1 = new DigitalVideoDisc("123", "Music", "thanh", 3, 15.21f);
        DigitalVideoDisc dvd2 = new DigitalVideoDisc("456", "Music", "quan", 5, 10.14f);
        DigitalVideoDisc dvd3 = new DigitalVideoDisc("789", "Music", "nguyen", 4, 20.51f);

        store.addDVD(dvd1);
        store.addDVD(dvd2);
        store.addDVD(dvd3);
        store.removeDVD(dvd2);
    }
}
```

Figure 18 StoreTest class

Result

```
123 is added successfully
456 is added successfully
789 is added successfully
456 is removed successfully
```

Figure 19 Result

6. String, StringBuilder and StringBuffer

Code

```
import java.util.Random;

public class ConcatenationInLoops {
    public static void main(String[] args) {
        Random r = new Random(123);
        long start = System.currentTimeMillis();
        String s = "";
        for (int i = 0; i < 65536; i++)
            s += r.nextInt(2);
        System.out.println(System.currentTimeMillis() - start);

        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);
    }
}
```

Figure 20 So sánh String và StringBuilder

Result

```
491
1
```

Figure 21 Result

```
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

//Doan Thanh Tung - 20225946
public class GarbageCreator {
    public static void main(String[] args) {
        String filePath = "src/hust/soict/globalict/garbage/test.txt"; // 166KB
        String garbage = "";
        long start = System.currentTimeMillis();
        try (FileReader fileReader = new FileReader(new File(filePath))) {
            int ch;
            while ((ch = fileReader.read()) != -1) {
                garbage += (char) ch;
            }
        } catch (FileNotFoundException e) {
            System.out.println(e.getMessage());
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }

        System.out.println("Time taken: " + (System.currentTimeMillis() - start) + " ms");
    }
}
```

Figure 22 GarbageCreator class

```
Time taken: 2091 ms
```

Figure 23 Result: program hangs

```
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

//Doan Thanh Tung - 20225946
public class NoGarbage {
    public static void main(String[] args) {
        String filePath = "src/hust/soict/globalict/garbage/test.txt"; // 166KB
        StringBuilder garbage = new StringBuilder();
        long start = System.currentTimeMillis();

        try (FileReader fileReader = new FileReader(new File(filePath))) {
            int ch;
            while ((ch = fileReader.read()) != -1) {
                garbage.append((char) ch);
            }
        } catch (FileNotFoundException e) {
            System.out.println("File not found: " + e.getMessage());
        } catch (IOException e) {
            System.out.println("Error reading file: " + e.getMessage());
        }
        System.out.println("Time taken: " + (System.currentTimeMillis() - start) + " ms");
    }
}
```

Figure 24 NoGarbage class

```
Time taken: 30 ms
```

Figure 25 Result: Program finishes quickly

7 Class Diagram

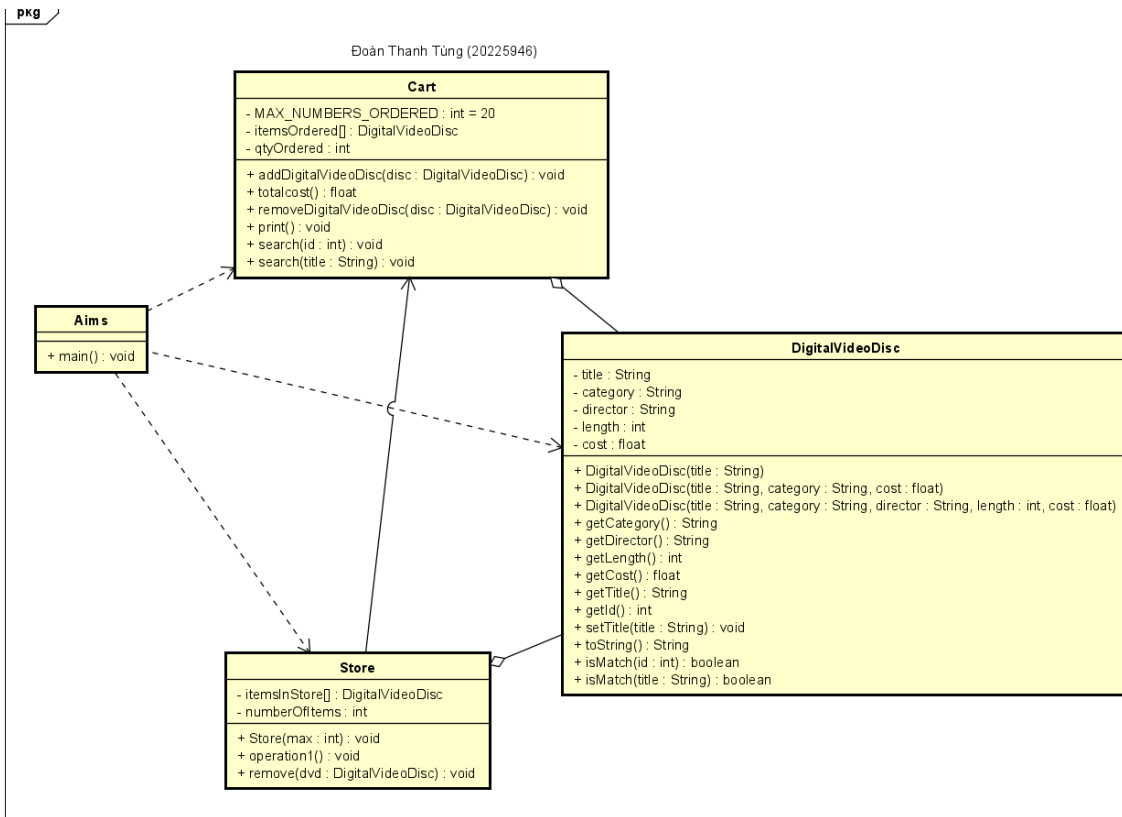


Figure 26 Class Diagram

Table of Figures

Figure 1 Method addDigitalVideoDisc(DigitalVideoDisc [] dvdList).....	2
Figure 2 Aims Class	2
Figure 3 Result.....	3
Figure 4 Method addDigitalVideoDisc(DigitalVideoDisc [] dvdList).....	4
Figure 5 Aims Class	4
Figure 6 Result.....	5
Figure 7 Passing parameter code	6
Figure 8 Passing parameter result.....	6
Figure 9 Class member and Instance member	8
Figure 10 Result.....	9
Figure 11 Code method to print the content in Cart.....	9
Figure 12 Code method search by id.....	10
Figure 13 Code method search by title	10
Figure 14 IsMatch() in DigitalVideoDisc class	10
Figure 15 CartTest class.....	11
Figure 16 Result.....	11
Figure 17 Store Class	12
Figure 18 StoreTest class	13
Figure 19 Result.....	13
Figure 20 So sánh String và StringBuilder	14
Figure 21 GarbageCreator class.....	14
Figure 22 Result: program hangs.....	15
Figure 23 NoGarbage class.....	15
Figure 24 Result: Program finishes quickly.....	15
Figure 25 Class Diagram	16