

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

BÁO CÁO THỰC HÀNH
IT3103-744528-2024.1
BÀI THỰC HÀNH -LAB02

Họ và tên sv: **Nguyễn Đăng**
Phúc Hưng

MSSV: **20226084**

Lớp: **IT-EP 01 K67**

GVHD: Lê Thị Hoa

HTGD: Đặng Mạnh Cường

BÁO CÁO THỰC HÀNH LAB 2

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Table of Contents

Table of Figures.....	1
1. Bài toán đặt ra.....	2
2. Yêu cầu hệ thống.....	2
2.1. Đối với Customer:.....	2
2.2. Đối với Store Manager:.....	2
3. Use Case Diagram	3
4. Class Diagram.....	4
5. Source Code	5
5.1. Aims Class.....	5
5.2. DigitalVideoDisc Class.....	6
5.3. Cart Class	8
6. Kết quả demo.....	10
7. Reading Assignment.....	12
8. Answer the question.....	12

Table of Figures

Figure 1: Use Case Diagram.....	3
Figure 2: Class Diagram.....	4
Figure 3: Aims Class	5
Figure 4: DigitalVideoDisc Class 1	6
Figure 5: DigitalVideoDisc Class 2	7
Figure 6: Cart Class 1.....	8
Figure 7: Cart Class 2.....	9
Figure 8: Code Demo	10
Figure 9: Result.....	11
Figure 10: Getter and Setter Methods.....	12

1. Bài toán đặt ra

Tiki và Sendo hợp tác sáp nhập trở thành công ty Ti-do. Họ yêu cầu xây dựng hệ thống mới cho dự án AIMS (An Internet Media Store), hiện tại chỉ cung cấp sản phẩm là đĩa DVD.

2. Yêu cầu hệ thống

2.1. Đối với Customer:

- Xem danh sách các đĩa DVD, hiển thị theo thứ tự từ mới nhất đến cũ nhất.
- Tra cứu đĩa DVD:
 - o Theo tiêu đề: người dùng nhập một chuỗi các từ khóa (không phân biệt chữ hoa thường), DVD được hiển thị nếu tiêu đề có chứa chuỗi kí tự con trùng với ít nhất 1 trong các từ khóa.
 - o Theo thể loại: người dùng nhập tên thể loại (không phân biệt chữ hoa thường), DVD được hiển thị nếu có chứa thể loại đó.
 - o Theo giá tiền: người dùng nhập giá tối đa, có thể nhập hoặc không nhập giá tối thiểu.
- Xem thông tin DVD và thêm vào giỏ hàng.
 - o Nghe demo: nếu độ dài của DVD nhỏ hơn hoặc bằng 0 thì thông báo không thể chạy DVD.
- Xem giỏ hàng:
 - o Hiển thị thông tin DVD.
 - o Hiển thị tổng giá.
 - o Nghe demo.
 - o Sắp xếp DVD:
 - Theo tiêu đề: theo thứ tự bảng chữ cái, sau đó theo giá từ cao xuống thấp.
 - Theo giá: theo thứ tự từ cao xuống thấp, sau đó theo thứ tự bảng chữ cái.
 - o Thêm, bớt số lượng DVD, xóa DVD khỏi giỏ hàng.
 - o Nhận 1 DVD miễn phí ngẫu nhiên do hệ thống chọn từ giỏ hàng.
 - o Lọc giỏ hàng: Nếu có thì hiển thị thông tin DVD, không thì hiển thị thông báo không tìm thấy sản phẩm trong giỏ hàng.
 - Theo ID
 - Theo tiêu đề
 - o Đặt hàng:
 - Người dùng nhập thông tin vận chuyển.
 - Hệ thống tính toán phí vận chuyển dựa trên khối lượng đơn hàng và địa điểm vận chuyển.
 - Hệ thống hiển thị hóa đơn:
 - Danh sách các đĩa DVD
 - Tổng giá trước VAT
 - Tổng giá sau VAT
 - Phí vận chuyển
 - Thanh toán: bằng thẻ tín dụng
 - Nhận thông tin đơn hàng và thanh toán qua email

2.2. Đối với Customer:

- Đăng nhập vào hệ thống
- Xem danh sách đơn hàng chờ duyệt.
 - o Xem thông tin đơn hàng.
 - o Duyệt hoặc hủy bỏ đơn hàng.
- Thêm DVD vào cửa hàng: nhập thông tin DVD:
 - o ID.
 - o Tiêu đề.
 - o Đạo diễn.
 - o Độ dài.
 - o Giá tiền.
- Xóa DVD khỏi cửa hàng.

3. Use Case Diagram

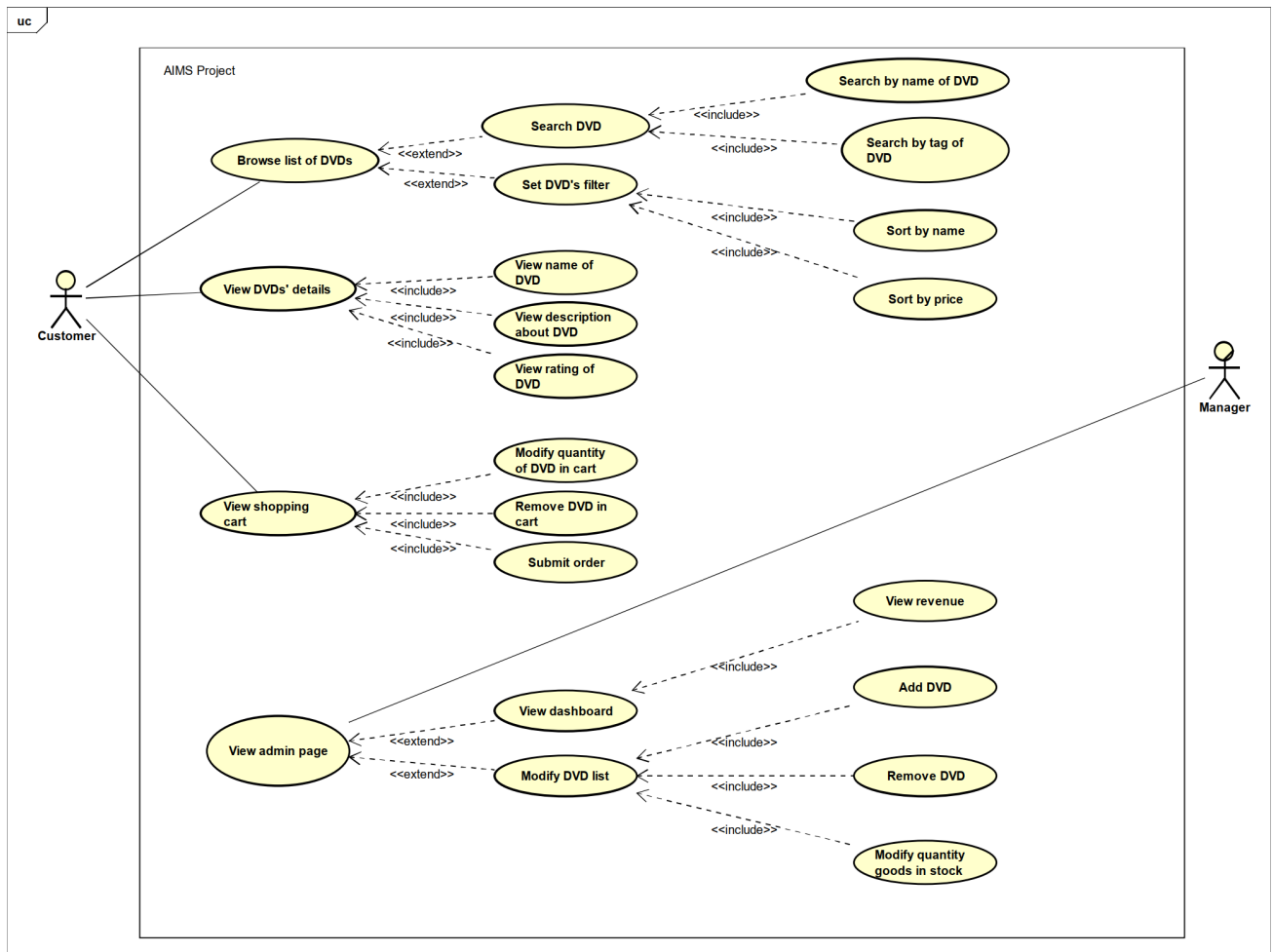


Figure 1: Use Case Diagram

4. Class Diagram

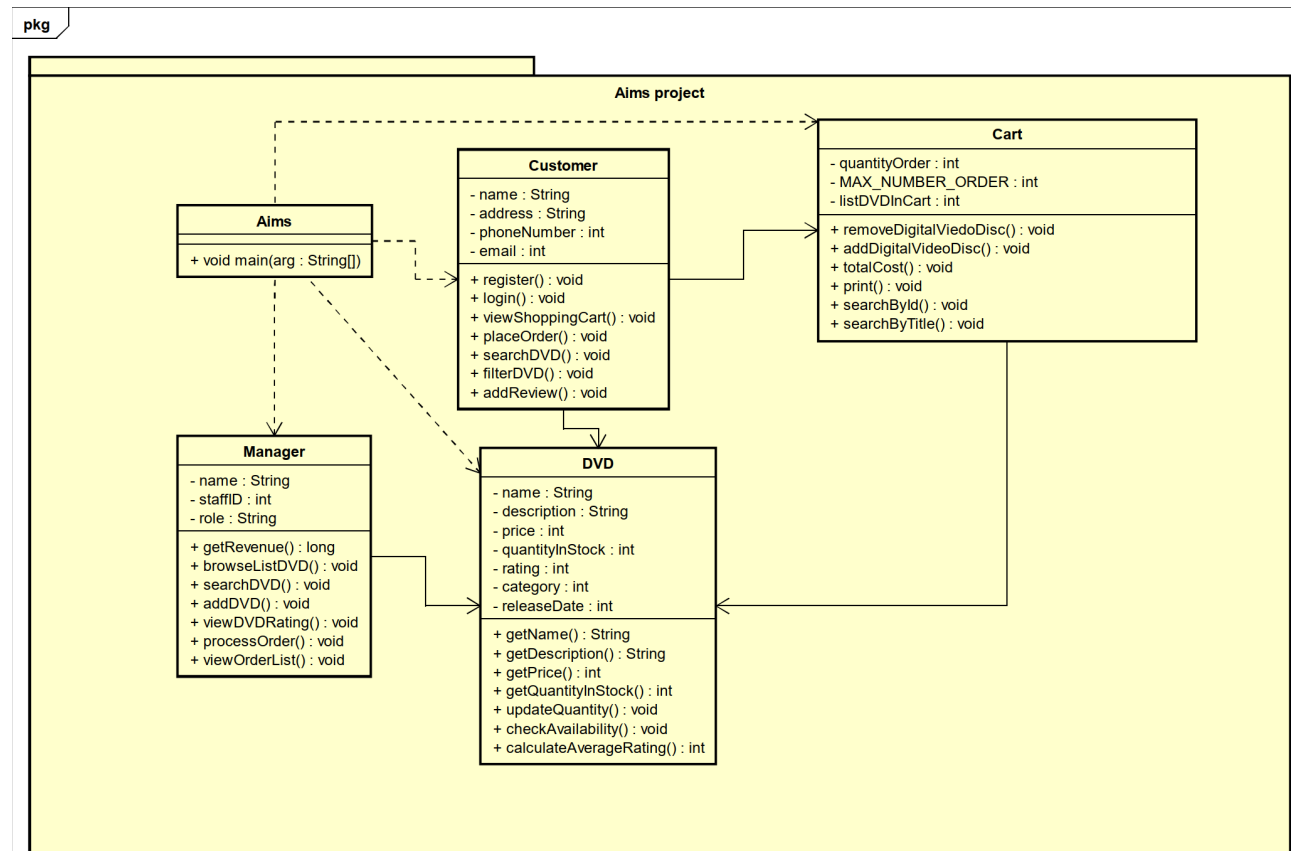


Figure 2: Class Diagram

5. Source Code

5.1. Aims Class

```
package Lab02.Code;

public class Aims {
    public static void main(String[] args) {
        Cart anOrder = new Cart();

        DVD dvd1 = new DVD("The Lion King", "Animation", "Roger Allers", 87, 19.95f);
        anOrder.addDigitalVideoDisc(dvd1);

        DVD dvd2 = new DVD("Star Wars", "Science Fiction", "George Lucas", 87, 24.95f);
        anOrder.addDigitalVideoDisc(dvd2);

        DVD dvd3 = new DVD("Aladdin", "Animation", 18.99f);
        anOrder.addDigitalVideoDisc(dvd3);

        System.out.println("Total Cost is: ");
        System.out.println(anOrder.totalCost());

        System.out.println("List before remove:");
        anOrder.print();

        anOrder.removeDigitalVideoDisc("Star Wars");

        System.out.println("\nList after remove:");
        anOrder.print();
    }
}
```

Figure 3: Aims Class

5.2. DigitalVideoDisc Class

```
package Lab02.Code;

import java.util.List;

public class DVD {
    String name;
    String description;
    double price;
    int quantityInStock;
    int rating;
    String genre;
    int releaseDate;
    String category;
    String author;

    public DVD(String name, String genre, String author, int releaseDate, double price) {
        this.name = name;
        this.genre = genre;
        this.author = author;
        this.releaseDate = releaseDate;
        this.price = price;
    }

    public DVD(String name, String description, int price, int quantityInStock, int rating, String genre,
        int releaseDate) {
        this.name = name;
        this.description = description;
        this.price = price;
        this.quantityInStock = quantityInStock;
        this.rating = rating;
        this.genre = genre;
        this.releaseDate = releaseDate;
    }

    public DVD(String name, int price) {
        this.name = name;
        this.price = price;
    }

    public DVD(String name, String genre, double price) {
        this.name = name;
        this.price = price;
        this.genre = genre;
    }

    public DVD(String name, String genre, int releaseDate) {
        this.name = name;
        this.genre = genre;
        this.releaseDate = releaseDate;
    }

    public String getName() {
        return name;
    }

    public String getDescription() {
        return description;
    }

    public double getPrice() {
        return price;
    }

    public int getQuantityInStock() {
        return quantityInStock;
    }
}
```



```
public void updateQuantity(int quantity) {
    quantityInStock += quantity;
    System.out.println("Quantity of DVD '" + name + "' updated. Current quantity: " + quantityInStock);
}

public void checkAvailability() {
    if (quantityInStock > 0) {
        System.out.println("DVD '" + name + "' is currently available.");
    } else {
        System.out.println("DVD '" + name + "' is currently out of stock.");
    }
}

public void calculateAverageRating(List<Integer> ratings) {
    if (ratings.isEmpty()) {
        rating = 0;
        return;
    }
    int sum = 0;
    for (int r : ratings) {
        sum += r;
    }
    rating = sum / ratings.size();
    System.out.println("Average rating of DVD '" + name + "': " + rating);
}

public void displayDVDInfo() {
    System.out.println("DVD Information:");
    System.out.println("- Title: " + name);
    System.out.println("- Description: " + description);
    System.out.println("- Price: " + price);
    System.out.println("- Quantity in stock: " + quantityInStock);
    System.out.println("- Rating: " + rating);
    System.out.println("- Genre: " + genre);
    System.out.println("- Release year: " + releaseDate);
}

public String getGenre() {
    return genre;
}
}
```

Figure 4: DVD

5.3. Cart Class

```

import java.util.ArrayList;
import java.util.List;

public class Cart {
    int quantityOrder;
    final int MAX_NUMBER_ORDER = 20;
    List<DVD> listDVDInCart = new ArrayList<>();

    public void removeDigitalVideoDisc(String dvdTitle) {
        for (int i = 0; i < listDVDInCart.size(); i++) {
            DVD dvd = listDVDInCart.get(i);
            if (dvd.getName().equals(dvdTitle)) {
                listDVDInCart.remove(i);
                quantityOrder--;
                System.out.println("Removed DVD '" + dvdTitle + "' from cart.");
                return;
            }
        }
        System.out.println("DVD '" + dvdTitle + "' not found in cart.");
    }

    public void addDigitalVideoDisc(DVD dvd) {
        if (quantityOrder >= MAX_NUMBER_ORDER) {
            System.out.println("Cart is full! Cannot add more DVDs.");
            return;
        }
        listDVDInCart.add(dvd);
        quantityOrder++;
        System.out.println("Added DVD '" + dvd.getName() + "' to cart.");
    }

    public double totalCost() {
        double total = 0;
        for (DVD dvd : listDVDInCart) {
            total += dvd.getPrice();
        }
        return total;
    }

    public void print() {
        if (listDVDInCart.isEmpty()) {
            System.out.println("Cart is empty.");
            return;
        }
        System.out.println("List of DVDs in cart:");
        for (DVD dvd : listDVDInCart) {
            System.out.println("- " + dvd.getName() + " - Price: " + dvd.getPrice());
        }
        System.out.println("Total cost: " + totalCost());
    }

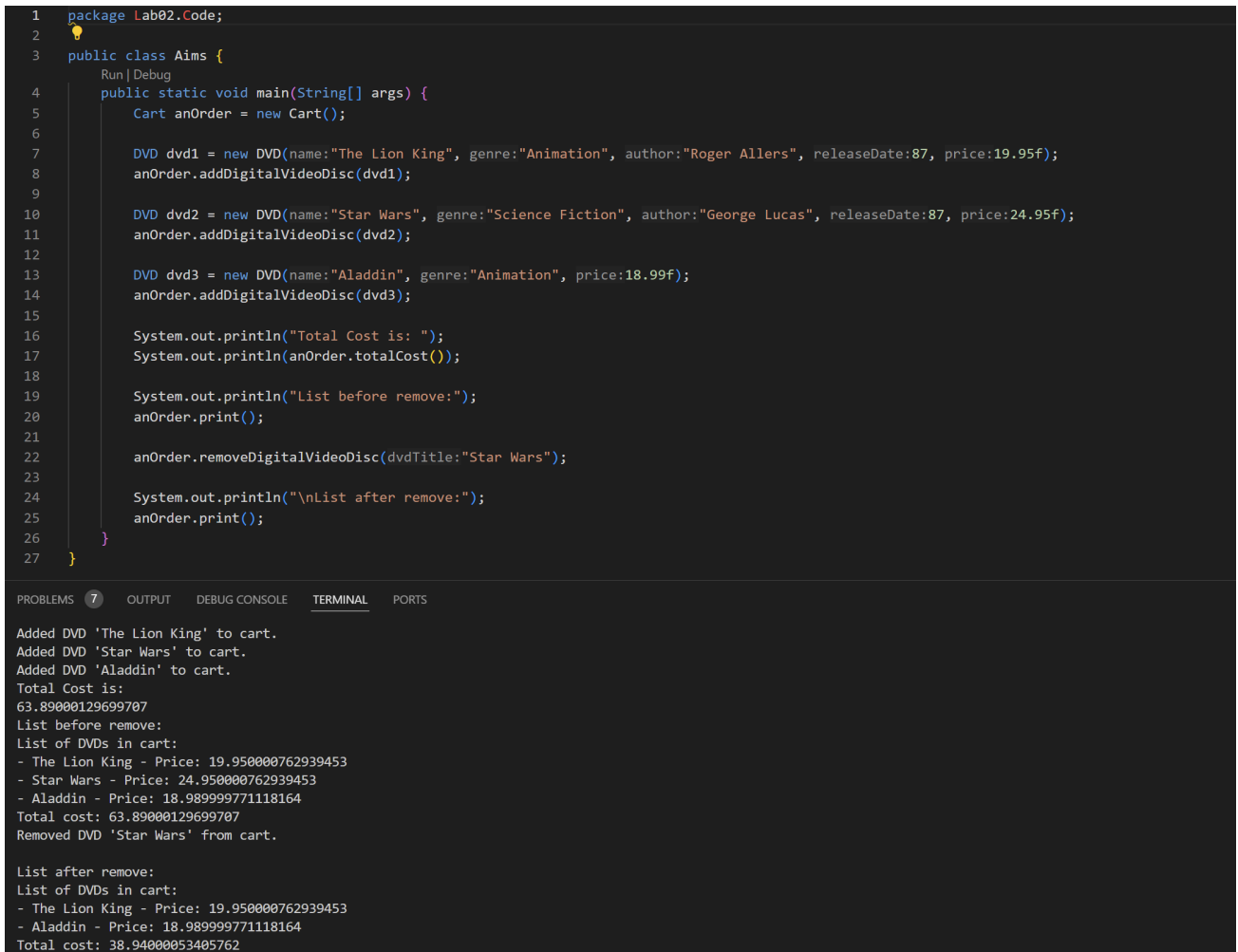
    public void searchById(int dvdId) {
        System.out.println("Search by ID function not yet implemented.");
    }

    public void searchByTitle(String dvdTitle) {
        for (DVD dvd : listDVDInCart) {
            if (dvd.getName().equals(dvdTitle)) {
                System.out.println("DVD Information:");
                dvd.displayDVDInfo();
                return;
            }
        }
        System.out.println("DVD '" + dvdTitle + "' not found in cart.");
    }
}

```

Figure 6: Cart

6. Kết quả demo



```
1 package Lab02.Code;
2
3 public class Aims {
4     Run | Debug
5     public static void main(String[] args) {
6         Cart anOrder = new Cart();
7
8         DVD dvd1 = new DVD(name:"The Lion King", genre:"Animation", author:"Roger Allers", releaseDate:87, price:19.95f);
9         anOrder.addDigitalVideoDisc(dvd1);
10
11        DVD dvd2 = new DVD(name:"Star Wars", genre:"Science Fiction", author:"George Lucas", releaseDate:87, price:24.95f);
12        anOrder.addDigitalVideoDisc(dvd2);
13
14        DVD dvd3 = new DVD(name:"Aladdin", genre:"Animation", price:18.99f);
15        anOrder.addDigitalVideoDisc(dvd3);
16
17        System.out.println("Total Cost is: ");
18        System.out.println(anOrder.totalCost());
19
20        System.out.println("List before remove:");
21        anOrder.print();
22
23        anOrder.removeDigitalVideoDisc(dvdTitle:"Star Wars");
24
25        System.out.println("\nList after remove:");
26        anOrder.print();
27    }
28 }
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

Added DVD 'The Lion King' to cart.
Added DVD 'Star Wars' to cart.
Added DVD 'Aladdin' to cart.
Total Cost is:
63.89000129699707
List before remove:
List of DVDs in cart:
- The Lion King - Price: 19.950000762939453
- Star Wars - Price: 24.950000762939453
- Aladdin - Price: 18.989999771118164
Total cost: 63.89000129699707
Removed DVD 'Star Wars' from cart.

List after remove:
List of DVDs in cart:
- The Lion King - Price: 19.950000762939453
- Aladdin - Price: 18.989999771118164
Total cost: 38.94000053405762

Figure 8: Code Demo
và kết quả

7. Reading Assignment

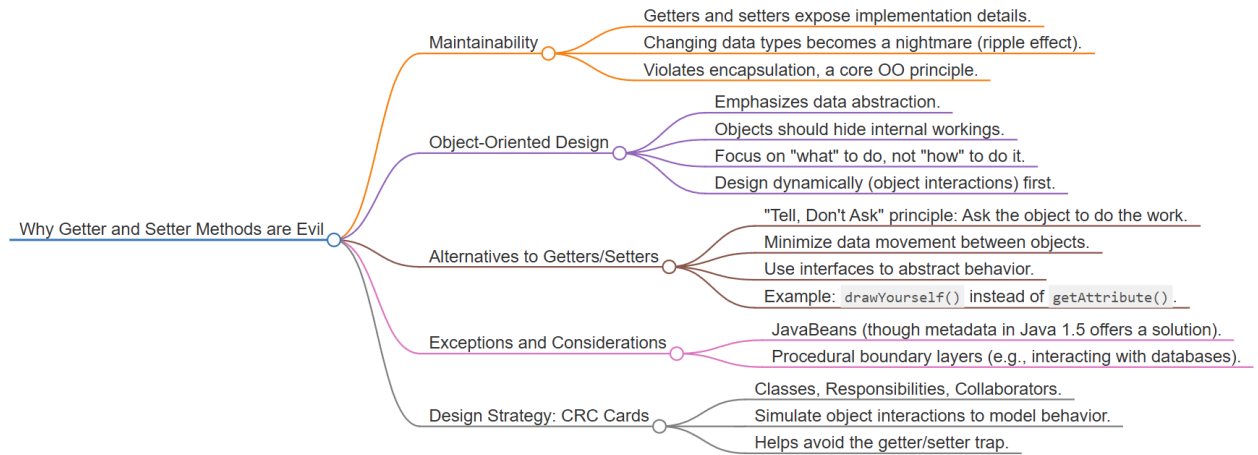


Figure 10: Getter and Setter Methods

8. Answer the question

Question:

- If you create a constructor method to build a DVD by title then create a constructor method to build a DVD by category. Does JAVA allow you to do this?

Answer:

No, it will cause to Constructor Overloading with the same parameters lists and Ambiguity with the same String data type input.

To avoid this problem, we should use different number of parameters and different data types.