Báo cáo Lab 04 (OOP\_TN)

Họ và tên: Phan Sỹ Hùng

MSSV: 20225631

1. Source code
2. Cart class

A computer screen shot of a program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated

1. Book class

A screen shot of a computer program

Description automatically generated

1. CompactDisc class

A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated

1. DigitalVideoDisc class

A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated

1. Disc class

A screen shot of a computer program

Description automatically generated

1. Media class

A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated

1. MediaComparatorByCostTitle class

A screen shot of a computer program

Description automatically generated

1. MediaComparatorByTitleCost class

A screen shot of a computer program

Description automatically generated

1. Playable class

A screen shot of a computer

Description automatically generated

1. Track class

A screen shot of a computer program

Description automatically generated

1. Store class

A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated

1. Aims class

A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated

1. Answer

When overriding the equals() method of the Object class, you will have to cast the Object parameter obj to the type of Object that you are dealing with. For example, in the Media class, you must cast the Object obj to a Media, and then check the equality of the two objects’ attributes as the above requirements (i.e. title for Media; title and length for Track). If the passing object is not an instance of Media, what happens?

* **instanceof Check Fails**:
* The if (!(obj instanceof Media)) or if (!(obj instanceof Track)) condition evaluates to true.
* **equals() Returns false**:
* The method immediately returns false, indicating that the objects are not equal.

# Polymorphism with toString() method

* **Tính đa hình (Polymorphism)**: Cho phép các đối tượng của các lớp con khác nhau (CD, DVD, Book) được lưu trữ trong cùng một danh sách kiểu lớp cha (Media), nhưng vẫn thể hiện hành vi khác nhau dựa trên lớp thực sự.
* **Ghi đè phương thức (Method Overriding)**: Phương thức toString() được ghi đè trong mỗi lớp con để tùy chỉnh cách hiển thị thông tin.

**Question**: Alternatively, to compare items in the cart, instead of using Comparator, we can use the Comparable interface and override the compareTo()method. You can refer to the Java docs to see the information of this interface.

Suppose we are taking this Comparable interface approach.

* What class should implement the Comparable interface?
* In those classes, how should you implement the compareTo()method be to reflect the ordering that we want?
* Can we have two ordering rules of the item (by title then cost and by cost then title) if we use this Comparable interface approach?
* Suppose the DVDs has a different ordering rule from the other media types, that is by title, then decreasing length, then cost. How would you modify your code to allow this?

1. What class should implement the Comparable interface?

Lớp Media nên implement Comparable<Media> để xác định quy tắc sắp xếp mặc định (natural ordering) cho tất cả các đối tượng thuộc lớp Media và lớp con của nó.

1. In those classes, how should you implement the compareTo()method be to reflect the ordering that we want?

Sắp xếp theo title (tăng dần), nếu bằng thì so tiếp theo cost (tăng dần):

1. Can we have two ordering rules of the item (by title then cost and by cost then title) if we use this Comparable interface approach?

* Không, vì Comparable chỉ cho phép một quy tắc sắp xếp tự nhiên duy nhất.
* Để hỗ trợ nhiều quy tắc sắp xếp, cần sử dụng Comparator.

1. Suppose the DVDs has a different ordering rule from the other media types, that is by title, then decreasing length, then cost. How would you modify your code to allow this?

Ghi đè compareTo() trong lớp DigitalVideoDisc để tùy chỉnh quy tắc sắp xếp riêng