

Reading Assignment:

Polymorphism có những lợi ích gì?

Polymorphism giúp tái sử dụng mã nguồn hiệu quả, giảm thiểu sự lặp lại trong code. Nó cũng mang lại sự linh hoạt và khả năng mở rộng, vì có thể dễ dàng thêm các lớp mới mà không thay đổi mã hiện tại. Điều này giúp hệ thống dễ bảo trì và mở rộng.

Inheritance có ích như thế nào trong việc đạt được Polymorphism trong Java?

Inheritance cho phép các lớp con kế thừa các thuộc tính và phương thức từ lớp cha, tạo điều kiện cho việc sử dụng phương thức chung từ lớp cha, nhưng hành vi có thể khác nhau dựa vào loại đối tượng. Điều này giúp đạt được polymorphism khi các lớp con có thể thay đổi hoặc mở rộng hành vi của lớp cha.

Sự khác biệt giữa Polymorphism và Inheritance trong Java là gì?

Polymorphism tập trung vào việc thay đổi hành vi của đối tượng dựa trên loại đối tượng thực tế, trong khi inheritance giúp tái sử dụng mã nguồn và tạo ra các lớp con từ lớp cha. Polymorphism cho phép sử dụng một phương thức với nhiều đối tượng khác nhau, còn inheritance tạo ra mối quan hệ giữa các lớp.

17.Question:

Lớp nào nên triển khai Comparable

Lớp Media nên triển khai Comparable<Media> vì đây là thứ tự sắp xếp tự nhiên cho các media item

Có thể có nhiều quy tắc sắp xếp với Comparable không?

Không, Comparable chỉ cho phép 1 thứ tự sắp xếp tự nhiên (phương thức compareTo). Để có nhiều cách sắp xếp, cần dùng các Comparator riêng (như giải pháp trước đó)

Thứ tự sắp xếp DVD (title → độ dài giảm dần → cost): Đầu tiên, so sánh tiêu đề (title) của hai DVD bằng cách sử dụng phương thức compareToIgnoreCase để đảm bảo so sánh không phân biệt chữ hoa và chữ thường. Nếu tiêu đề khác nhau, kết quả so sánh sẽ được trả về ngay lập tức. Nếu tiêu đề giống nhau, tiếp tục so sánh độ dài (length) của hai DVD. Cuối cùng, nếu cả tiêu đề và độ dài của hai DVD đều giống nhau, so sánh chi phí (cost) của chúng theo thứ tự giảm dần

```
public class DVDTitleLengthCostComparator implements Comparator<DigitalVideoDisc> {  
    @Override  
    public int compare(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {  
        int titleCompare = dvd1.getTitle().compareToIgnoreCase(dvd2.getTitle());  
        if (titleCompare != 0) return titleCompare;  
        int lengthCompare = Integer.compare(dvd2.getLength(), dvd1.getLength());  
        if (lengthCompare != 0) return lengthCompare;  
        return Double.compare(dvd2.getCost(), dvd1.getCost());  
    }  
}
```