

🕒 Cập nhật tháng 8 năm 2024

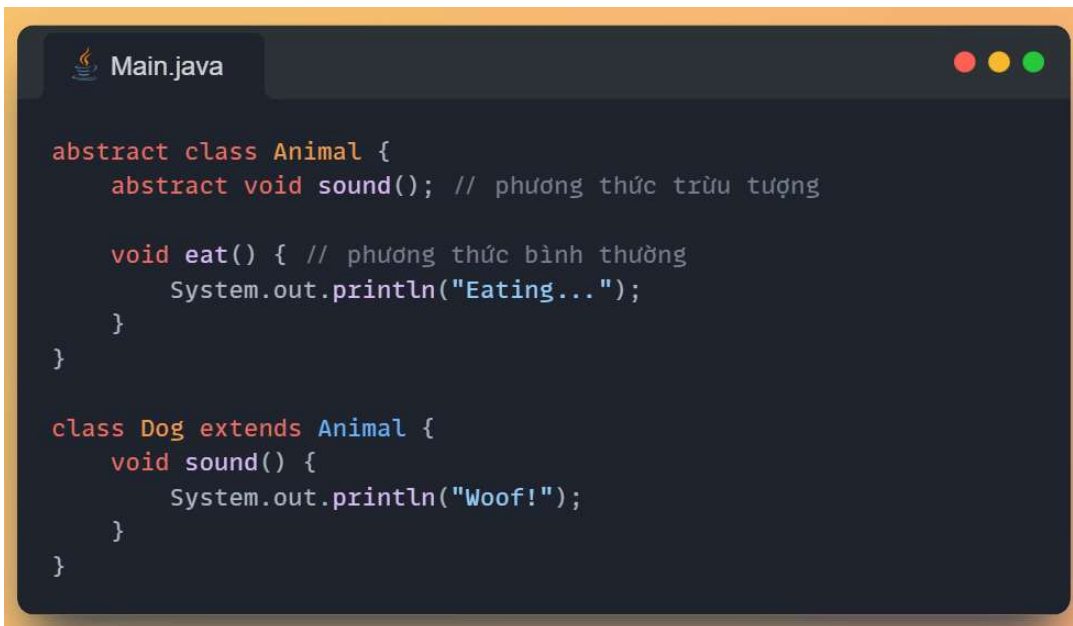
[Bài đọc] Tính trừu tượng

1. Khái niệm:

- Tính trừu tượng là quá trình **ẩn đi chi tiết cài đặt** và chỉ **hiển thị những gì cần thiết** cho người dùng. Nó giúp giảm độ phức tạp của chương trình bằng cách che giấu các phần không quan trọng, chỉ cung cấp các thông tin cốt lõi
- Ví dụ thực tế: Khi bạn lái xe, bạn chỉ cần biết cách điều khiển vô lăng, phanh, ga,... mà không cần hiểu động cơ hoạt động thế nào. Đó là trừu tượng

2. Tính trừu tượng trong Java hoạt động như thế nào

- Trong Java, tính trừu tượng được thực hiện bằng hai cách:
- **Abstract Class (Lớp trừu tượng)**
 - Được khai báo với từ khóa **abstract**
 - Có thể chứa cả phương thức **có cài đặt** và **không có cài đặt (abstract method)**
 - Không thể tạo đối tượng từ lớp trừu tượng
 - Ví dụ:



```

Main.java

abstract class Animal {
    abstract void sound(); // phương thức trừu tượng

    void eat() { // phương thức bình thường
        System.out.println("Eating...");
    }
}

class Dog extends Animal {
    void sound() {
        System.out.println("Woof!");
    }
}
    
```

- **Interface (Giao diện)**
 - Chỉ chứa các phương thức **không có cài đặt** (từ Java 8 có thể có phương thức **default** và **static**)
 - Dùng từ khóa **interface**
 - Một lớp có thể **implement** nhiều **interface**

- Ví dụ:

```
Main.java

interface Drawable {
    void draw();
}

class Circle implements Drawable {
    public void draw() {
        System.out.println("Drawing circle...");
    }
}
```

3. Lợi ích của tính trừu tượng

- **Giảm độ phức tạp:** Chỉ tập trung vào hành vi chính
- **Tăng tính bảo trì:** Dễ thay đổi hoặc mở rộng hệ thống
- **Tăng tính bảo mật:** Ẩn đi logic xử lý bên trong

4. Khi nào nên dùng trừu tượng?

- Khi bạn muốn xác định **hành vi chung** cho nhiều đối tượng con
- Khi bạn muốn xây dựng một **khuôn mẫu (template)** cho các lớp con

5. So sánh Abstract Class vs Interface

Tiêu chí	abstract	interface
Từ khóa	abstract	interface
Kế thừa	1 lớp	nhiều interface
constructor	Có thể có	Không có
Loại phương thức	Có thể có cả 2 loại	chủ yếu là abstract

Tài nguyên đọc thêm: <https://www.geeksforgeeks.org/abstraction-in-java-2>

Danh sách các bài học

