

🕒 Cập nhật tháng 8 năm 2024

[Bài đọc] Upcasting và Downcasting

1. Upcasting (Ép kiểu lên)

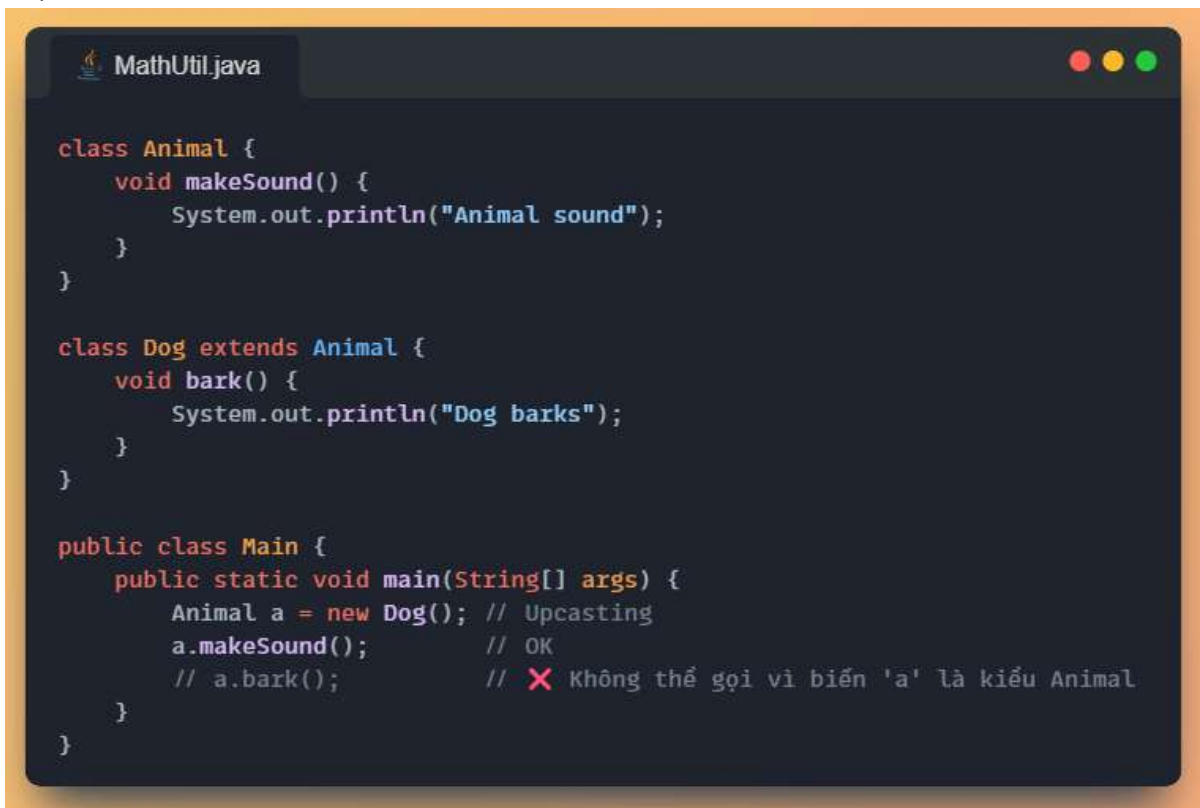
- Upcasting là khi bạn gán đối tượng của lớp con cho một biến kiểu lớp cha
- Java tự động thực hiện việc này mà không cần ép kiểu tường minh



```
MathUtil.java

Parent obj = new Children(); // Upcasting
```

- Ví dụ:



```
MathUtil.java

class Animal {
    void makeSound() {
        System.out.println("Animal sound");
    }
}

class Dog extends Animal {
    void bark() {
        System.out.println("Dog barks");
    }
}

public class Main {
    public static void main(String[] args) {
        Animal a = new Dog(); // Upcasting
        a.makeSound();         // OK
        // a.bark();            // ✗ Không thể gọi vì biến 'a' là kiểu Animal
    }
}
```

- Ưu điểm của upcasting:
 - Cho phép sử dụng tính đa hình (polymorphism)

- Giúp code linh hoạt và dễ mở rộng

2. Downcasting (Ép kiểu xuống)

- **Downcasting** là khi bạn gán **đối tượng lớp cha** (thực chất đang tham chiếu đến lớp con) cho một **biến kiểu lớp con**
 - Java **không tự động** thực hiện - bạn cần ép kiểu **tường minh (explicit casting)**
 - Nếu đối tượng không thực sự thuộc lớp con, sẽ **gây lỗi tại runtime** (**ClassCastException**)
- **Cú pháp:**

```
MathUtil.java

Children obj = (Children) new LopCha(); // Downcasting
```

- **Ví dụ:**

```
MathUtil.java

Animal a = new Dog();           // Upcasting
Dog d = (Dog) a;                 // Downcasting hợp lệ
d.bark();                       // OK
```

- **Downcasting không hợp lệ**

```
MathUtil.java

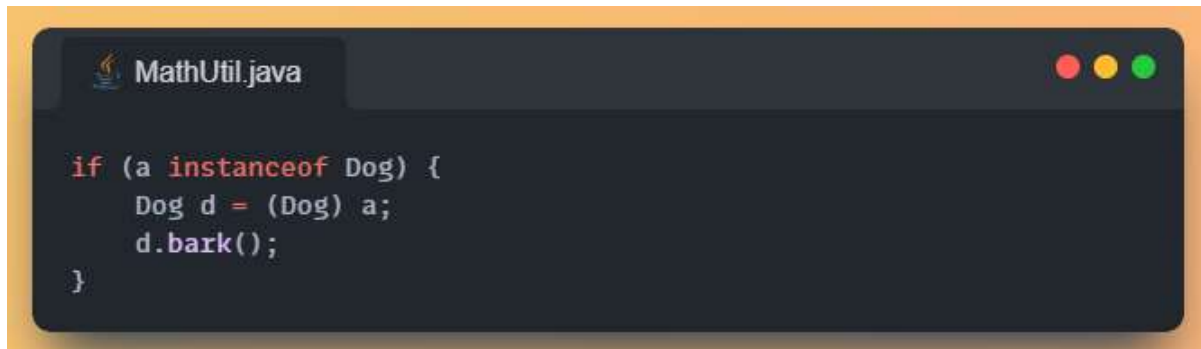
Animal a = new Animal();
Dog d = (Dog) a; // Lỗi tại runtime! Vì 'a' không thật sự là Dog
```

3. Tóm tắt

Đặc điểm	UpCasting	DownCasting
Hướng ép kiểu	Lớp con → lớp cha	Lớp cha → lớp con
Tự động thực hiện	Có	Ko - Cần ép kiểu tường minh
An toàn	Rất an toàn	Có thể gây lỗi nếu không kiểm tra
Sử dụng	Đa hình, tính tổng quát	Truy cập chức năng cụ thể của lớp con

4. Cách an toàn khi Downcasting

- Dùng toán tử `instanceof` để kiểm tra kiểu thực tế trước khi ép:

A screenshot of a code editor window titled "MathUtil.java". The code inside is a Java snippet demonstrating safe downcasting. It uses the `instanceof` operator to check if a variable `a` is an instance of the `Dog` class before casting it to `Dog` and calling the `bark()` method.

```
if (a instanceof Dog) {  
    Dog d = (Dog) a;  
    d.bark();  
}
```

Tài nguyên đọc thêm: <https://www.scientecheasy.com/2020/07/upcasting-and-downcasting-in-java.html>

Danh sách các bài học

