

 Cập nhật tháng 8 năm 2024

[Bài đọc] Tính kế thừa và từ khóa extends

1. Khái niệm

- **Kế thừa** là một trong bốn đặc trưng cơ bản của lập trình hướng đối tượng. Nó cho phép một **lớp con (subclass)** kế thừa các thuộc tính (biến) và phương thức (hàm) từ một **lớp cha (superclass)**
- Giúp tái sử dụng mã, dễ bảo trì, mở rộng chương trình

2. Từ khóa extends

- Công dụng:
 - Từ khóa **extends** được dùng để khai báo rằng một lớp kế thừa từ một lớp khác
- Cú pháp:



```
MathUtil.java

class Parent {
    // thuộc tính và phương thức
}

class Children extends Parent {
    // kế thừa LopCha và có thể mở rộng thêm
}
```

- Sau khi kế thừa, lớp con có thể:
 - Dùng lại các biến và phương thức công khai (public/protected) từ lớp cha
 - Viết thêm phương thức riêng
 - **Ghi đè (override)** phương thức lớp cha nếu cần
- Ví dụ:

```
class Person {  
    void about() {  
        System.out.println("Tôi là một người.");  
    }  
}  
  
class Student extends Person {  
    void study() {  
        System.out.println("Tôi đang học Java.");  
    }  
}  
  
public class Main {  
    public static void main(String[] args) {  
        Student sv = new Student();  
        sv.about(); // gọi từ lớp cha  
        sv.study(); // gọi từ lớp con  
    }  
}
```

```
Tôi là một người.  
Tôi đang học Java.
```

3. Ghi nhớ

- **extends** dùng để kế thừa từ một lớp khác
- Java chỉ cho phép kế thừa đơn (một lớp con chỉ kế thừa từ một lớp cha)
- Tính kế thừa giúp tái sử dụng mã và tổ chức chương trình tốt hơn

Tài nguyên đọc thêm: https://www.w3schools.com/java/ref_keyword_extends.asp