

 Cập nhật tháng 8 năm 2024

[Bài đọc] Từ khóa super và phương thức super()

1. Từ Khóa super trong Java

Từ khóa super được sử dụng để tham chiếu đến đối tượng của lớp cha (superclass). Nó giúp truy cập các thành viên (biến và phương thức) của lớp cha.

- Sử dụng super để gọi phương thức của lớp cha:

```
class Animal {  
    void sound() {  
        System.out.println("Animal makes a sound");  
    }  
}  
  
class Dog extends Animal {  
    void sound() {  
        super.sound(); // Gọi phương thức sound() của lớp cha  
        System.out.println("Dog barks");  
    }  
}  
  
public class Test {  
    public static void main(String[] args) {  
        Dog dog = new Dog();  
        dog.sound();  
    }  
}
```

Trong ví dụ trên, `super.sound()` gọi phương thức `sound()` của lớp `Animal` trước khi in ra "Dog barks" từ lớp `Dog`.

- Sử dụng super để truy cập biến của lớp cha:

```

class Animal {
    // Khai báo biến name trong lớp Animal
    String name = "Animal";
}

class Dog extends Animal {
    // Khai báo biến name trong lớp Dog
    String name = "Dog";

    void display() {
        // Truy cập biến name của lớp con (Dog)
        System.out.println("Name from subclass: " + name);

        // Truy cập biến name của lớp cha (Animal) thông qua super
        System.out.println("Name from superclass: " + super.name);
    }
}

public class Test {
    public static void main(String[] args) {
        // Tạo đối tượng dog từ lớp Dog
        Dog dog = new Dog();
        // Gọi phương thức display để in ra giá trị biến name của lớp con và lớp cha
        dog.display();
    }
}

```

Kết quả sẽ là:

```

Name from subclass: Dog
Name from superclass: Animal

```

Dùng `super.name` để truy cập biến name từ lớp cha (Animal).

2. Phương Thức `super()` trong Java

Phương thức `super()` là một phương thức đặc biệt trong Java được sử dụng để gọi đến constructor của lớp cha. Phương thức này phải được gọi trong constructor của lớp con và phải là câu lệnh đầu tiên trong constructor.

- Sử dụng `super()` để gọi constructor của lớp cha:

```
class Animal {
    Animal() {
        System.out.println("Animal constructor");
    }
}

class Dog extends Animal {
    Dog() {
        super(); // Gọi constructor của lớp cha
        System.out.println("Dog constructor");
    }
}

public class Test {
    public static void main(String[] args) {
        Dog dog = new Dog();
    }
}
```

Kết quả sẽ là:

```
Animal constructor
Dog constructor
```

Phương thức `super()` gọi constructor của lớp cha (Animal) trước khi thực thi constructor của lớp con (Dog).

- **Khi không gọi `super()`:** Nếu bạn không gọi `super()` một cách rõ ràng, Java sẽ tự động gọi constructor mặc định của lớp cha (nếu có). Nếu lớp cha không có constructor mặc định (không có tham số), thì bạn phải gọi constructor của lớp cha một cách rõ ràng.

```
class Animal {
    Animal(String name) {
        System.out.println("Animal constructor with name: " + name);
    }
}

class Dog extends Animal {
    Dog() {
        super("Dog"); // Phải gọi constructor của lớp cha có tham số
        System.out.println("Dog constructor");
    }
}

public class Test {
    public static void main(String[] args) {
        Dog dog = new Dog();
    }
}
```

Kết quả sẽ là:

```
Animal constructor with name: Dog
Dog constructor
```

Link tài nguyên đọc thêm:

- super: <https://www.geeksforgeeks.org/super-keyword/>

Danh sách các bài học

