

 Cập nhật tháng 8 năm 2024

[Bài đọc] Vòng lặp for và ứng dụng

1. Tại sao lại cần vòng lặp for

- Trong lập trình, có nhiều lúc bạn cần **lặp lại một hành động nhiều lần**, ví dụ:
 - In ra 100 số tự nhiên đầu tiên
 - Tính tổng từ 1 đến n
 - Duyệt qua từng phần tử của mảng
- Nếu viết từng dòng lệnh cho mỗi lần lặp thì rất mất công và dài dòng.
- Vì vậy, Java cung cấp vòng lặp **for** để **tự động lặp lại một khối lệnh**.

2. Cấu trúc cơ bản của vòng lặp for

Vòng lặp for được sử dụng khi bạn biết số lần cần lặp trước. Cấu trúc của vòng lặp for trong Java như sau:

```
for (initialization; condition; increment/decrement) {  
    // Khối lệnh cần thực thi  
}
```

Giải thích:

- **Initialization (Khởi tạo):** Biến điều kiện được khởi tạo trước khi vòng lặp bắt đầu. Đây thường là biến đếm
- **Condition (Điều kiện):** Điều kiện được kiểm tra trước mỗi lần lặp. Nếu điều kiện này là true, vòng lặp sẽ tiếp tục; nếu là false, vòng lặp dừng
- **Increment/Decrement (Tăng/Giảm):** Sau mỗi lần lặp, giá trị của biến điều kiện được thay đổi theo hướng tăng hoặc giảm

Ví dụ về vòng lặp for:

```

public static void main(String[] args) new *
{
    // Vòng lặp in ra các số từ 1 đến 5
    for (int i = 1; i <= 5; i++) {
        System.out.println("Số hiện tại: " + i);
    }
}

```

Giải thích:

- `int i = 1`: Khởi tạo biến `i` với giá trị 1.
- `i <= 5`: Kiểm tra điều kiện, vòng lặp sẽ tiếp tục nếu `i` nhỏ hơn hoặc bằng 5.
- `i++`: Sau mỗi lần lặp, giá trị của `i` sẽ được tăng thêm 1.

Kết quả:

```

Số hiện tại: 1
Số hiện tại: 2
Số hiện tại: 3
Số hiện tại: 4
Số hiện tại: 5

```

3. Ứng dụng phổ biến

Ứng dụng thực tế	Mô tả
In ra dãy số	in ra từ 1 đến 100
Tính tổng, Tính tích	Dùng lặp để cộng hoặc nhân
Duyệt mảng	Xem từng phần tử
Vẽ hình, hiệu ứng	Tạo mô hình bằng ký tự
Kiểm tra điều kiện lặp	Kết hợp <code>for</code> với <code>if</code>

4. Lợi ích của vòng lặp for

- **Tối ưu code**: Ngắn gọn, dễ đọc
- **Tăng hiệu suất**: Lặp nhanh và tự động
- **Linh hoạt**: Kết hợp với điều kiện để xử lý logic phức tạp
- **Nền tảng của các thuật toán**: Như sắp xếp, tìm kiếm, xử lý dữ liệu lớn

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

