

 Cập nhật tháng 8 năm 2024

## [Bài Đọc] Infinite loop và Nested loop trong javascript

### 1. Infinite Loop (Vòng lặp vô hạn)

- Một **infinite loop** trong JavaScript là một vòng lặp không bao giờ kết thúc vì điều kiện thoát khỏi vòng lặp không bao giờ được đáp ứng. Đây thường là lỗi do lập trình viên vô tình gây ra, nhưng đôi khi nó được sử dụng một cách có mục đích trong các ứng dụng đặc biệt.
- Cách xảy ra vòng lặp vô hạn
  - Không có điều kiện thoát hợp lệ:

```
while (true) {  
  console.log(...data: "Vòng lặp này sẽ chạy mãi mãi");  
}
```

- Điều kiện thoát luôn đúng (hoặc luôn sai)

```
let i = 0;  
while (i >= 0) {  
  console.log(...data: i); // Không bao giờ thoát khỏi vòng lặp  
  i++; // Giá trị của i luôn tăng  
}
```

- Cách tránh vòng lặp vô hạn
  - Đảm bảo điều kiện thoát **có thể đạt được**:

```
let i = 0;  
while (i < 10) {  
  console.log(...data: i);  
  i++; // Điều kiện thoát là khi i >= 10  
}
```

- Tránh thay đổi giá trị điều kiện không mong muốn:

```
let i = 0;
while (i < 5) {
  console.log(...data: i);
  i--; // Lỗi: Điều kiện i < 5 không bao giờ đúng
}
```

- Ứng dụng của vòng lặp vô hạn
  - Server hoặc trình lắng nghe sự kiện:

```
while (true) {
  // Xử lý sự kiện hoặc yêu cầu liên tục
}
```

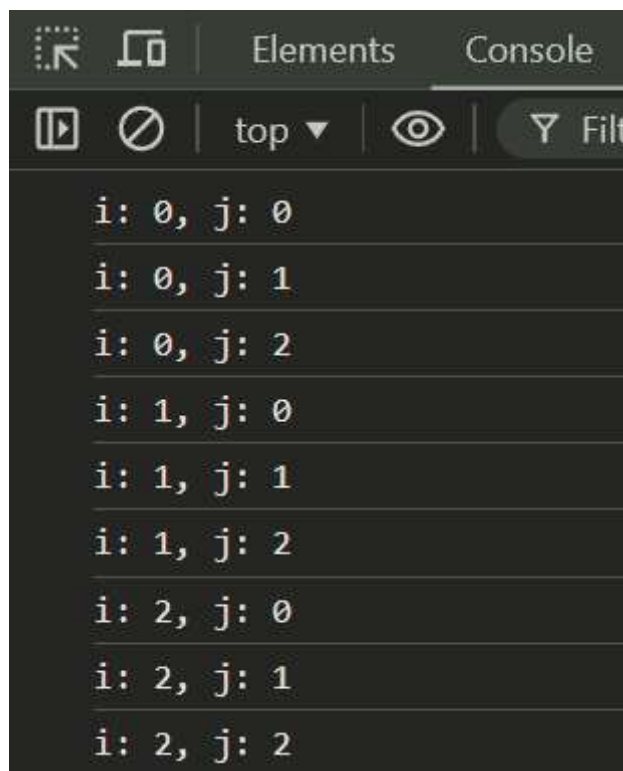
## 2. Nested Loop (Vòng lặp lồng nhau)

- Một **nested loop** là vòng lặp nằm trong một vòng lặp khác. Trong JavaScript, bạn có thể sử dụng vòng lặp lồng nhau để xử lý các cấu trúc dữ liệu phức tạp, như mảng nhiều chiều.

Ví dụ:

```
for (let i = 0; i < 3; i++) {
  for (let j = 0; j < 3; j++) {
    console.log(...data: `i: ${i}, j: ${j}`);
  }
}
```

Kết quả:



### 3. Tài liệu tham khảo

<https://flexiple.com/javascript/infinite-loops-javascript>

<https://www.tutorialspoint.com/how-to-use-nested-for-loop-in-javascript>

#### Danh sách các bài học

