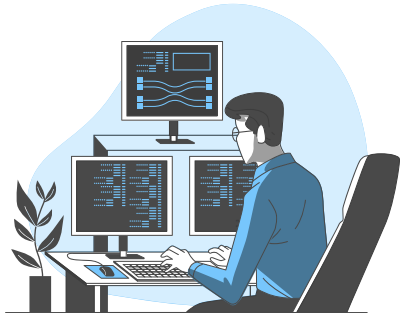
Advanced

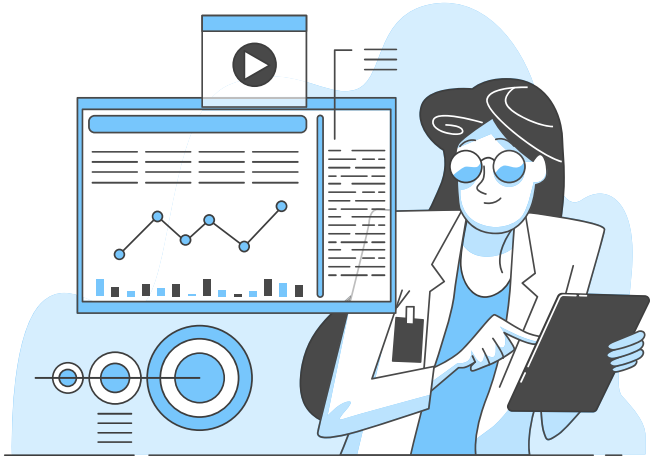
JS

Khóa học Backend

Bài 11: Javascript nâng cao (Tiết 1)



Nội dung



01

Scope

01. Scope

- **Scope** là **phạm vi truy cập**, nó đề cập đến ngữ cảnh của đoạn code.
- Có **2 kiểu** phạm vi là:
  - Phạm vi **toàn cục** (global): Một biến nằm ở phạm vi toàn cục thì biến đó được **sử dụng ở đâu cũng được**.
  - Phạm vi **cục bộ** (local): Một biến được khai báo **trong một hàm** thì biến này là biến cục bộ, và **chỉ sử dụng được ở trong hàm đó**.

02

Hoisting

2.1. Biến trong hoisting

- **Hoisting** nghĩa là có thể **sử dụng 1 biến xong** sau đó **mới cần khai báo** biến đó.
- Javascript sẽ di chuyển toàn bộ các khai báo biến lên đầu chương trình.
- Vì vậy, những dòng code có sử dụng biến mà chưa khai báo sẽ không bị lỗi.

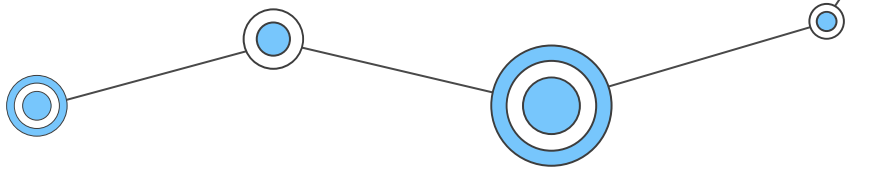
2.2. Từ khóa let, const và var trong hoisting

- Sử dụng **var**:
  - Nếu biến **chưa gán giá trị** thì sẽ **trả về undefined**.

- Sử dụng **let** hoặc **const**:
  - Nếu biến **chưa gán giá trị** thì sẽ xuất hiện thông báo **lỗi "a is not defined"**.
  - Lỗi này muốn nói rằng biến a chưa được định nghĩa.

2.3. Hàm trong hoisting

- **Declaration** Function
  - **Có** tính **hosting**
  - *Cú pháp: function tenHam(){ // Code }*
- **Expression** Function
  - **Không** có tính **hosting**
  - *Cú pháp: var tenBien = function(){ // Code }*
- **Arrow** Function
  - **Không** có tính **hosting**
  - *Cú pháp: var tenBien = () => { // Code }*



03

Từ khóa "this"

3.1. This trong Javascript

- Từ khóa **this** sẽ **trỏ về đối tượng** mà **nó đang thuộc về**.
- *Lưu ý: Nếu đặt **this** ở **ngoài cùng** chương trình thì **this** sẽ **trả về** đối tượng **window**.*

3.2. This trong các sự kiện javascript

- Khi gán sự kiện cho một phần tử HTML, thì **this** chính là phần tử HTML đó.

3.3. This trong javascript object

- *(Không có lý thuyết, sẽ có ví dụ trong lúc học)*

3.4. This trong arrow function

- Arrow function **không tồn tại** đối tượng **this**.
- Khi sử dụng **this** sẽ lấy đối tượng **window**.

04

Modules

4.1. Module trong Javascript là gì?

- Một **module** là một **file javascript bình thường**.
- File đó đặt tên có ý nghĩa dựa trên các dòng code bên trong.
- Ví dụ: file **sum.js** sẽ chứa **hàm tính tổng bên trong**.
- Công dụng: giúp code nhanh hơn, mạch lạc hơn, tái sử dụng ở nhiều nơi.

4.2. Khai báo và sử dụng module

- **Khai báo** module
  - Dùng từ khóa **export**.
  - Cú pháp:

```
export function tenHam() {
  // Code
}
```

- **Sử dụng** module
  - Dùng từ khóa **import**.
  - Cú pháp:

```
import { tenHam } from "duong_dan_file.js";
```

4.3. Đổi tên module

- Có một số trường hợp bạn phải **đổi tên module**:
  - Muốn **rút gọn tên** cho đỡ dài.
  - Trong trường hợp source code của bạn **đã tồn tại tên đó**, thì phải thay **đổi tên để tránh bị trùng tên**.

- **Cách 1**: Đổi tên trong file module.

```
// Trong file module:
export { oldName as newName };

// Trong file cần dùng đến module đó:
import { newName } from "duong_dan_file.js";
```

- **Cách 2**: Đổi tên trong file import.

```
import { oldName as newName } from "duong_dan_file.js";
```

4.4. Default export module

- Dùng để export mặc định.
- Mỗi file chỉ có 1 hàm (hoặc 1 biến) được export default.
- Cú pháp:

```
// Trong file module:
export default function tenHam() {
  // Code
}

// Trong file cần dùng đến module đó:
import tenHam from "duong_dan_file.js";
```

05

JSON

5.1. Khái niệm

- **JSON** viết tắt của **J**ava**S**cript **O**bject **N**otation.
- Là một định dạng dữ liệu được lưu dưới **dạng chuỗi**.

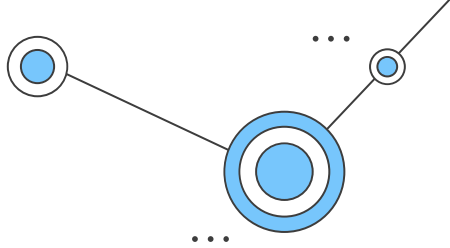
- **Chỉ cho phép** các kiểu dữ liệu cơ bản: number, string, boolean, array, object, null.
- **Không cho phép**: function, date, undefined.

- Trường hợp giá trị của JSON là dạng Object thì:
  - Có các **cặp key/value**.
  - **key**: đặt **trong dấu nháy kép**.
  - Không có dấu phẩy ở cặp key/value cuối cùng.

5.2. Các ví dụ

- **Ví dụ 1**: JSON có giá trị là Object
- **Ví dụ 2**: JSON có giá trị là Number
- **Ví dụ 3**: JSON có giá trị là String
- **Ví dụ 4**: JSON có giá trị là Boolean
- **Ví dụ 5**: JSON có giá trị là Array
- **Ví dụ 6**: JSON có giá trị là Null

Bài tập



- **Link bài tập**: https://dacavn.notion.site/B-i-t-p-b-i-11-Javascript-n-ng-cao-Ti-t-1-3b1f975d1c6a48a4ada8bb1cb029bcdf?pvs=4