JS BaSic

# 1. javascrip là gì?

Javascrip là ngôn ngữ lập trình được sử dùng nhiều trên nền web.

# 2. Nodejs?

Được việt bằng js.

https://topdev.vn/blog/node-js-la-gi/

Tài liệu:

https://codelearn.io/learning/javascript-co-ban

# Đường dẫn js trong html

<script src="https://www.w3schools.com/js/myScript1.js"></script>

<script src="/js/myScript1.js"></script>

# Output print:

console.log(); => in trên cmd console

doccoment.write(); => in trên web

document.getElementById("demo").innerHTML = “”; document.getElementById("demo").style.color = "red";

=> thay đổi dữ liệu trong thẻ có id=”demo”.

alert(); => in ra thông báo

confirm(); => in ra thông báo

prompt(); => in ra thông báo

# Console :

console.log() => in ra nội dung.

console.warn() => in ra cảnh báo.

console.erro() => in ra cảnh báo lỗi.

setTime:

## setTimeout

setTimeout(function() {

Lệnh;

},1000)

setInterval(function() {

Lệnh;

}, 1000)

# Biến - dữ liệu

## Kiểu khai báo biến:

Const: biến không thể thay đổi và chỉ tồn tại trong {}

Let: từ khóa để xác định một biến có phạm vi hạn chế. biến được khởi tạo bằng let sẽ chỉ tồn tại trong {}

Var: dù đặt ở đâu thì tên biến sẽ được khai báo và mang lên trên đầu code.

Không khai báo: dữ liệu và biến sẽ được tính từ lúc bạn khai báo.

## Kiểu dữ liệu của biến:

Kiểu đơn giản:

- kiểu số - number

- kiểu string

- kiểu true flase

- undefined

- null

- symbol

Kiểu phức tạp:

- Function

- Object: Object và array

## Kiểm tra dữ liệu:

Typeof biến;

isNaN(biến);

Chuyển kiểu:

Biến = Biến.toString()

Biến = string(biến)

Biến = number(biến)

## Kiểu true false:

Nhận giá trị false:

False

‘’

Undefined

NaN

Null

## Kiểu string

<https://www.w3schools.com/js/js_string_methods.asp>

https://www.youtube.com/watch?v=b4YivuRmcEw&list=PL\_-VfJajZj0VgpFpEVFzS5Z-lkXtBe-x5&index=23

chuoi = `them bien ${biến} tesst `;

tìm kiếm sửa xóa chuyển đổi kiếm đếm.

Loại bỏ các kí tự đặc biệt.

Chuyển sang dạng mảng

Biến.length()

**Các lệnh thao tác:**

Length chuỗi.length

Find index:

+ chuỗi.indexOf(“chuỗi cần tìm”, ví trí bắt đầu) trả về vị trí đầu tiên tìm được

+ chuỗi.lastIndexOf(“chuỗi cần tìm”) trả về vị trí cuối cần tìm.

Cut string chuỗi.slice(vị trí đầu, vị trí cuối)

Replace chuỗi.replace.(“chuỗi cần thay”, “chuỗi thay”)

Thay tất cả sử dụng biểu thức chính quy chuỗi.replace.(/chuỗi cần thay/g, chuỗi thay”)

Convert to upper case chuỗi.toUpperCase() => in thường thành in hoa

Convert to lower case chuỗi.toLowerperCase() => in thường thành in hoa

Trim chuỗi .trim() loại bỏ khoảng trắng ở đầu cuối.

Split chuỗi.split(“kí tự phân tách thành mảng”) chuyển chuỗi thành array

Get a character by index chuỗi.charAt(vị trí) lấy ra kí tự

## Kiểu number

- làm tròn.

Biến.toFixed(số đằng sau dấu phẩy)

## Kiểu array

<https://www.w3schools.com/js/js_arrays.asp>

https://www.youtube.com/watch?v=KrYacXScNQk&list=PL\_-VfJajZj0VgpFpEVFzS5Z-lkXtBe-x5&index=26

Array.isArray(biến) kiểm tra xem có phải kiểu array không?

Làm việc với mảng:

- Chuyển sang kiểu chuỗi. array.toString()

- join array.join(“ kí tự phân tách”) => chuyển sang chuỗi

- pop array.pop() => xóa phần tử cuối mảng. và trả về phần tử đã xóa

- push array.push(phần tử 1, phần tử 2, …) => thêm phần tử cuối mảng và trả về độ dài của mảng

- shift array.pop() xóa phần tử đầu mảng. và chả về phần tử đã xóa.

- unshift array.unshif(phần tử 1, phần tử 2, …) => thêm phần tử vào đầu mảng và trả về độ dài mảng.

- splicing array.splice(truyền vị chí, số phần từ muốn xóa, phần tử 1 muốn thêm, phần tử 2 muốn thêm, …)

- concat array.concat(array2). Hợp nhất 2 mảng.

- slicing array.slice(truyền vị trí, xóa đến ví trí nào).

# Một số từ khóa:

|  |  |
| --- | --- |
| **Từ khóa** | **Sự miêu tả** |
| break | Kết thúc một công tắc hoặc một vòng lặp |
| continue | Nhảy ra khỏi vòng lặp và bắt đầu ở trên cùng |
| debugger | Ngừng thực thi JavaScript và gọi (nếu có) hàm gỡ lỗi |
| try ... catch | Triển khai xử lý lỗi cho một khối câu lệnh |

# Events

|  |  |
| --- | --- |
| onchange | Một phần tử HTML đã được thay đổi |
| onclick | Người dùng nhấp vào một phần tử HTML |
| onmouseover | Người dùng di chuyển chuột qua một phần tử HTML |
| onmouseout | Người dùng di chuyển chuột khỏi phần tử HTML |
| onkeydown | Người dùng nhấn một phím bàn phím |
| onload | Trình duyệt đã tải xong trang |

object: đối tượng