

TÀI LIỆU LẬP TRÌNH WEB

JAVASCRIPT



October 29, 2022

LÂM VŨ THÀNH TÀI

0787903125 - lamvuthanhtai14042002@gmail.com

**MENU**

[**1.** **Giới Thiệu JS** 2](#_Toc118666716)

[**2.** **Sử dụng JS trong file HTML** 2](#_Toc118666717)

[**3.** **Biến JS** 3](#_Toc118666718)

[**4.** **Comments** 4](#_Toc118666719)

[**5.** **Hàm Built-in** 4](#_Toc118666720)

[**5.1** **Alert** 4](#_Toc118666721)

[**5.2** **Console** 5](#_Toc118666722)

[**5.2.1** **console.log** 5](#_Toc118666723)

[**5.2.2** **console.warn** 5](#_Toc118666724)

[**5.2.3** **console.error** 6](#_Toc118666725)

[**5.3** **Confirm** 6](#_Toc118666726)

[**5.4** **Prompt** 6](#_Toc118666727)

[**5.5** **Set timeout** 7](#_Toc118666728)

[**5.6** **Set interval** 7](#_Toc118666729)

[**6.** **Operator (Toán Tử)** 8](#_Toc118666730)

[**6.1** **Arithmetic (Số học)** 8](#_Toc118666731)

[**6.1.1** **Prefix and Postfix (Tiền tố và hậu tố)** 8](#_Toc118666732)

[**6.2** **Assignment (Gán)** 8](#_Toc118666733)

[**6.3** **Comparison (So sánh)** 8](#_Toc118666734)

[**6.4** **Logic (logical)** 8](#_Toc118666735)

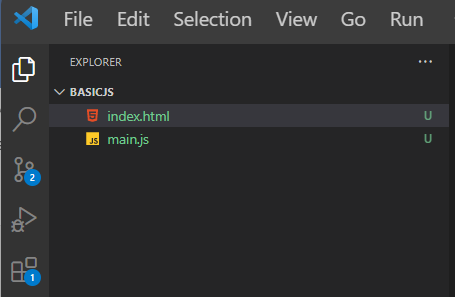
[**6.5** **String (chuỗi)** 8](#_Toc118666736)

1. **Giới Thiệu JS**

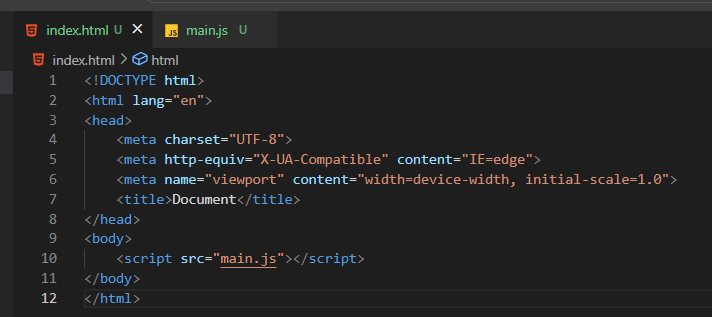
-Giúp tạo những hiệu ứng phức tạp và tạo những sự kiện, trải nghiệm người dùng tốt hơn, bên cạnh đó cũng giúp việc kết hợp giữa font end và back end dễ dàng hơn

1. **Sử dụng JS trong file HTML**

Bước 1: Tạo 1 file.js bên ngoài, có thể cùng cấp với file.html

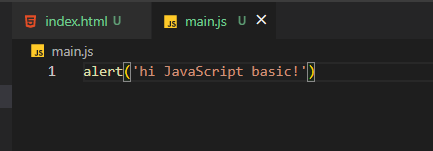


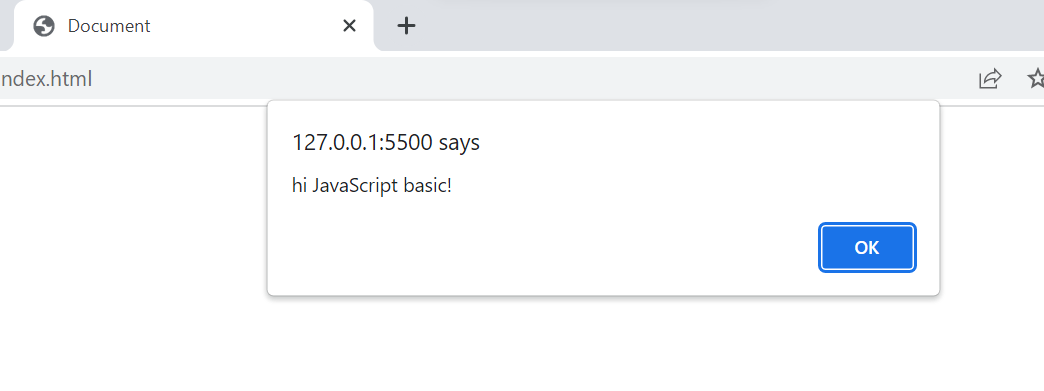
Bước 2: Trong phần body sử dụng thẻ <script></script> và link đến file.js vừa tạo



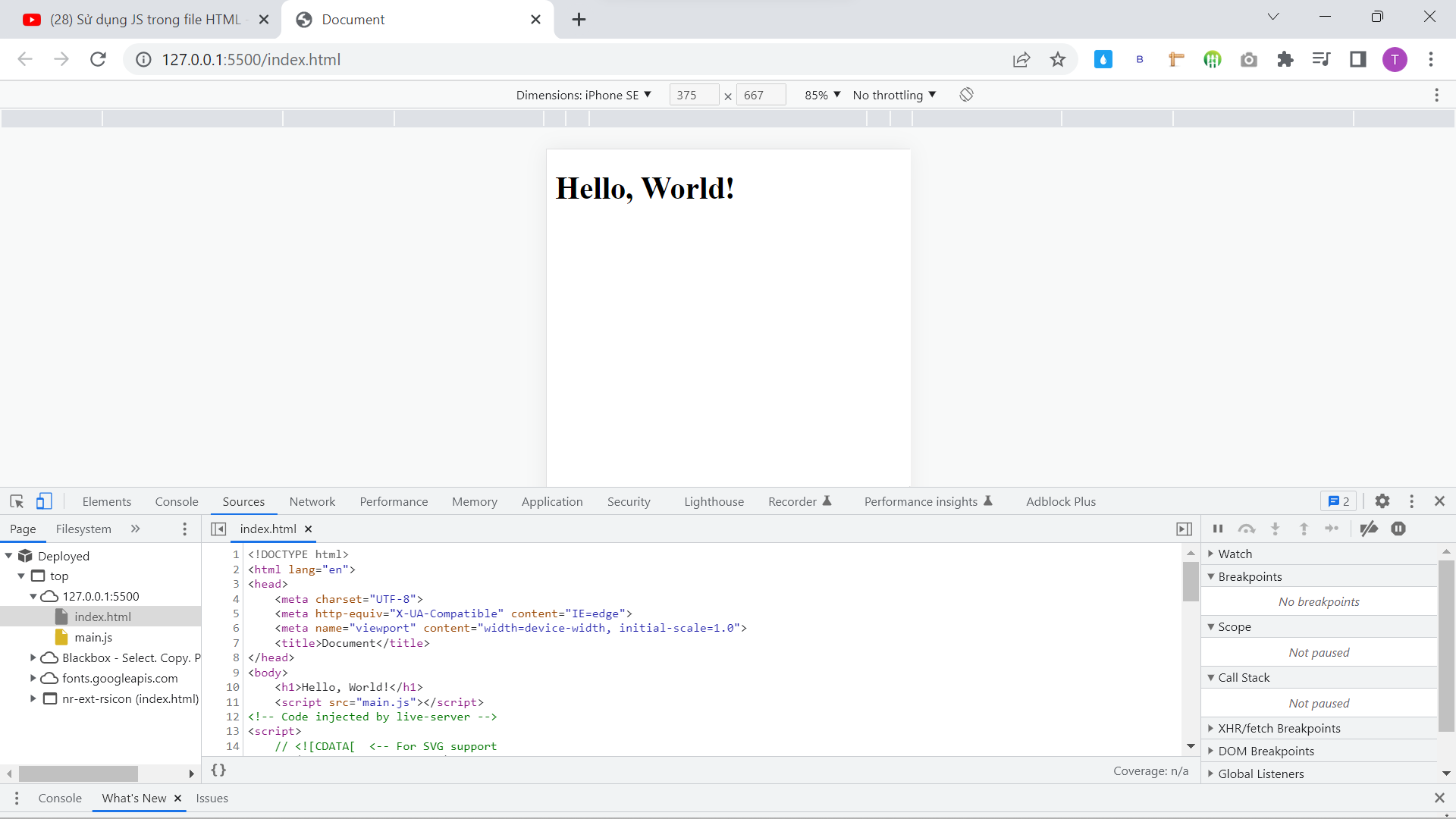
Bước 3: kiểm tra liên kết có thành công hay không

+Code vài dòng bên file.js sau đó chạy live sever check kết quả





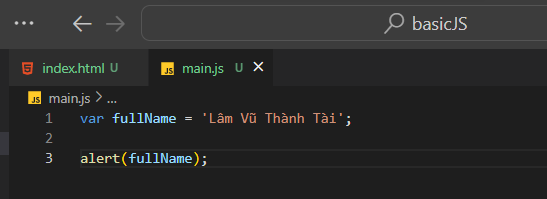
+Kiểm tra bằng Devtool



1. **Biến JS**

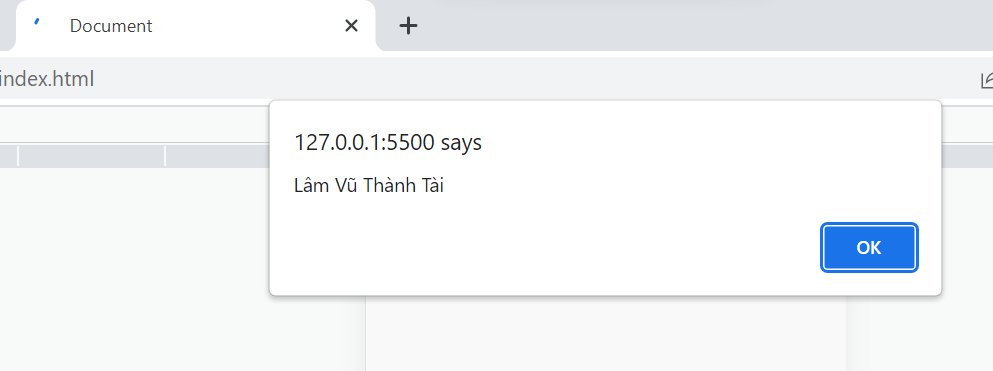
-Dùng **var** để khai báo biến:

var tênbiến = ‘Giá trị gán vào’



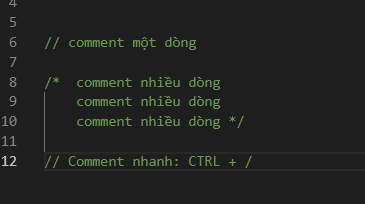
Dấu = dùng để gán giá trị cho biến

Kết quả:



1. **Comments**

-Giúp ghi chú, vô hiệu hóa hoặc giải thích code



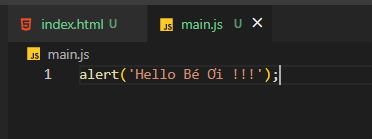
1. **Hàm Built-in**

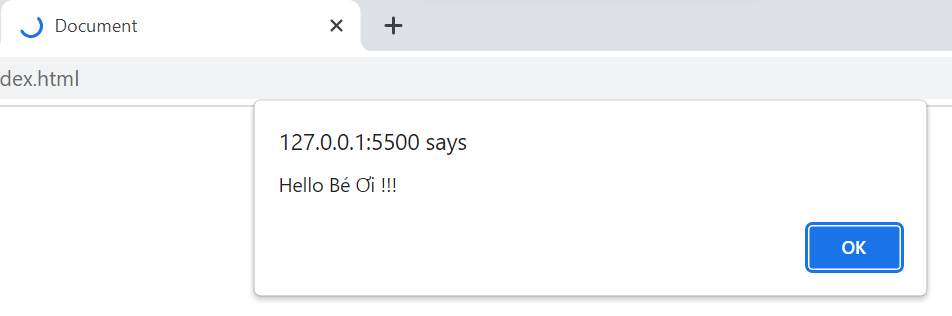
-Là các hàm có sẵn trong JS, chỉ việc nhớ và gọi ra sài khi cần thôi !

* 1. **Alert**

-Dùng để hiển thị message (thông báo) trên web site.

-Cú pháp: alert(‘Nội dung message’);



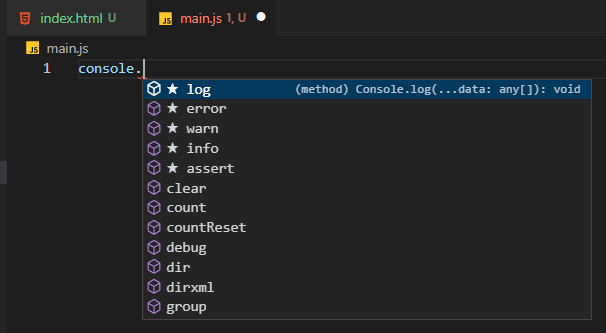


-Thông báo sẽ hiện cho đến khi chúng ta tắt (bấm OK).

* 1. **Console**

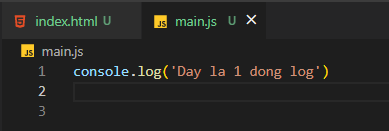
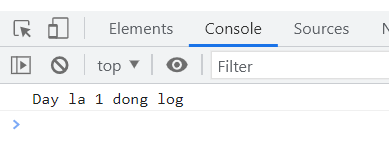
-Là một đối tượng có chứa rất nhiều các methods (phương thức).

-Cách gọi các phương thức của console ta dùng cú pháp: console + “dấu chấm”

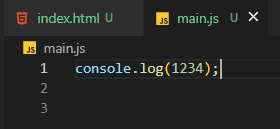
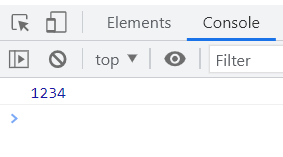


* + 1. **console.log**

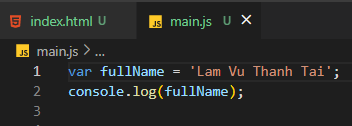
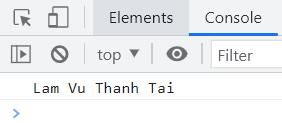
-Giúp hiển thị cảnh báo trên màn hình console (trong dev tool)

-Ngoài giá trị chuỗi ta có thể gán giá trị số

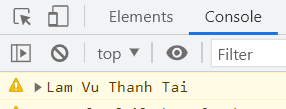
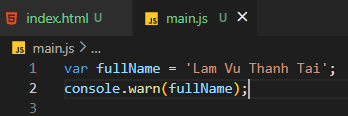
 

-Gán giá trị là biến JS

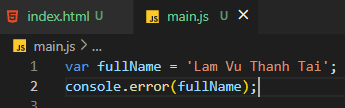
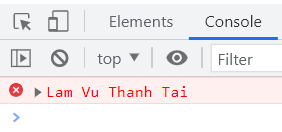
* + 1. **console.warn**

-Dùng để cảnh báo trên màng hình console



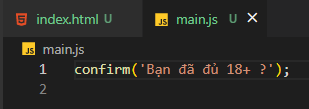
* + 1. **console.error**

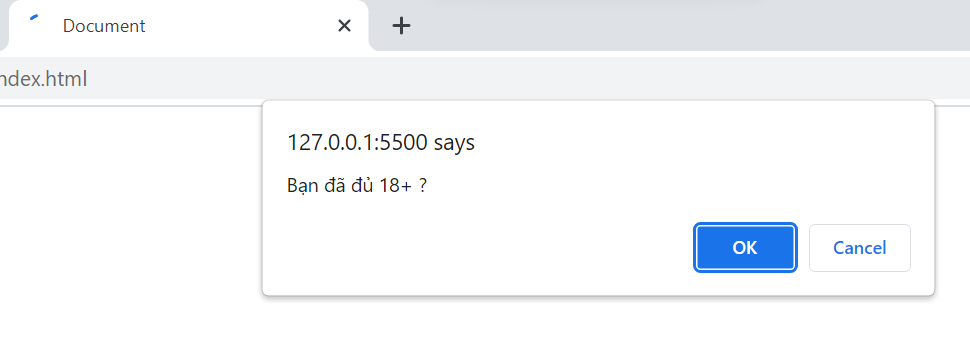
-Dùng để báo lỗi trên màng hình console

* 1. **Confirm**

-Dùng để thông báo (message) và yêu cầu user xác nhận.

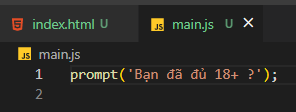


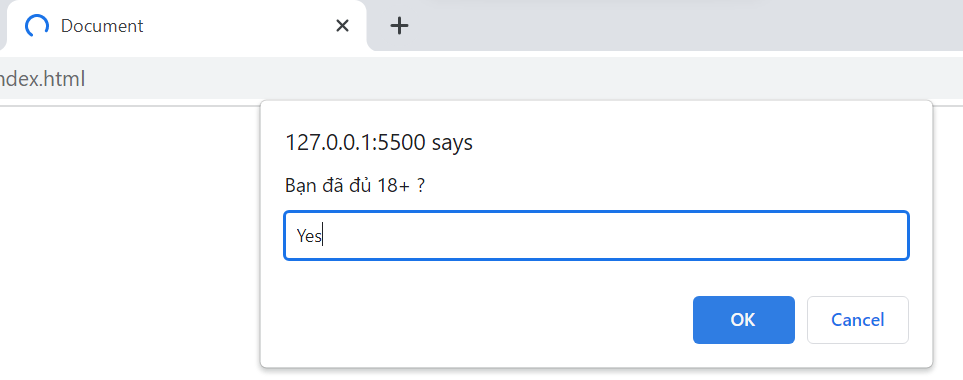
****

-Sẽ hiện hộp thông báo đến khi người dùng xác nhận.

* 1. **Prompt**

-Kết hợp Alert và Confirm, ngoài ra yêu cầu người dùng nhập input để xác nhận.





-Hộp thoại sẽ hiện đến khi user nhập input và click xác nhận.

* 1. **Set timeout**

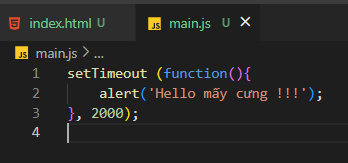
-Dùng để chạy một đoạn code trong một khoản thời gian (được tính là ms)

-Cú pháp:

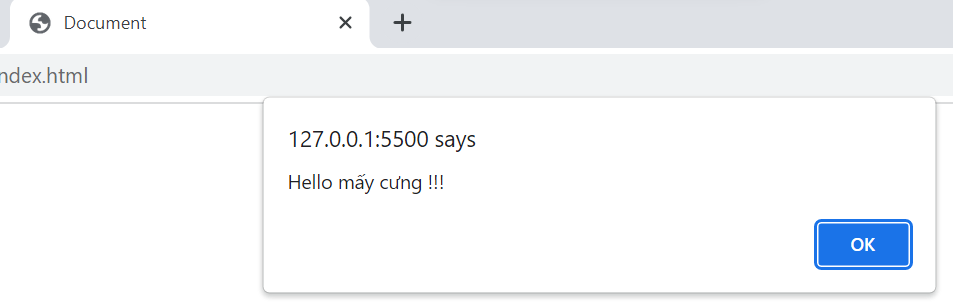
setTimeout (function(){

//Code thực thi;

}, thời gian thực hiện);



Sau 2s thì sẽ hiện lên thống báo:



* 1. **Set interval**

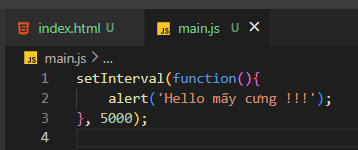
-Dùng để chạy một đoạn code trong một khoản thời gian (được tính là ms) nhưng khác với timout là nó sẽ lặp lại vô tận

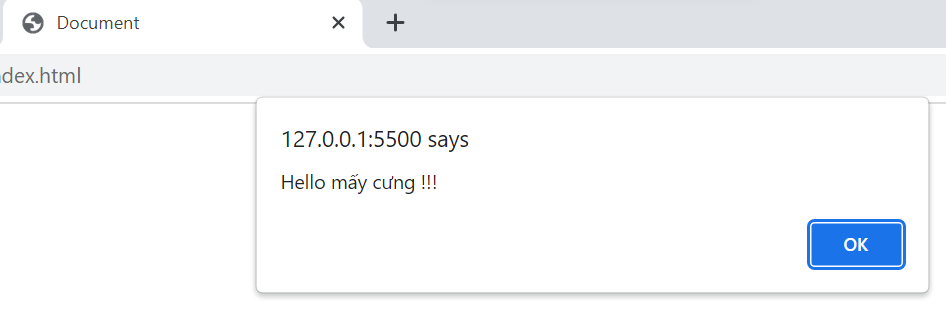
-Cú pháp:

setInterval (function(){

//Code thực thi;

}, thời gian thực hiện);

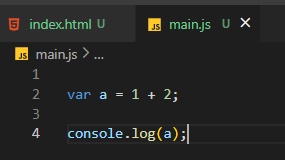
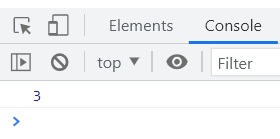
****



-Dù user có tắt thông báo thì cứ mỗi 5s sẽ hiển thị lại !

1. **Operator (Toán Tử)**
   1. **Arithmetic (Số học)**

- **Các toán tử số học** (toán tử đại số) này thực hiện trên các số - dữ liệu dạng số (cụ thể hoặc là biến). Chúng cũng gần tương tự như trong toán học thôi.

** **

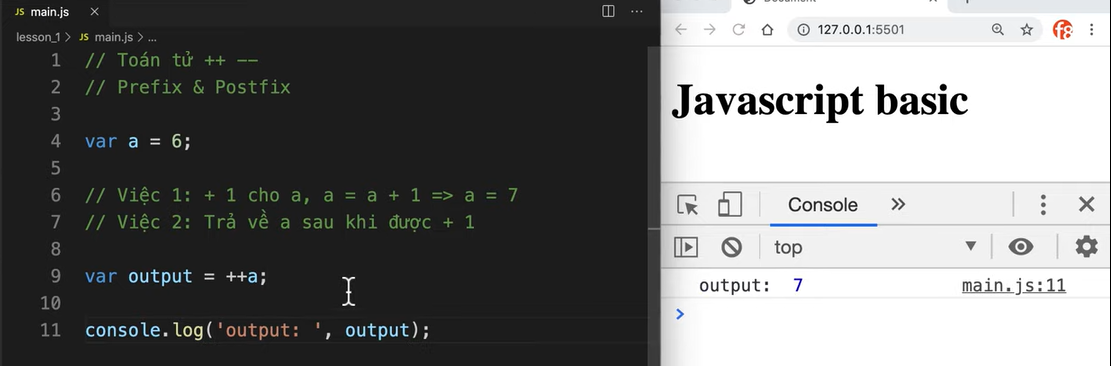
-Ngoài ra còn có các toán tử:



* + 1. **Prefix and Postfix (Tiền tố và hậu tố)**

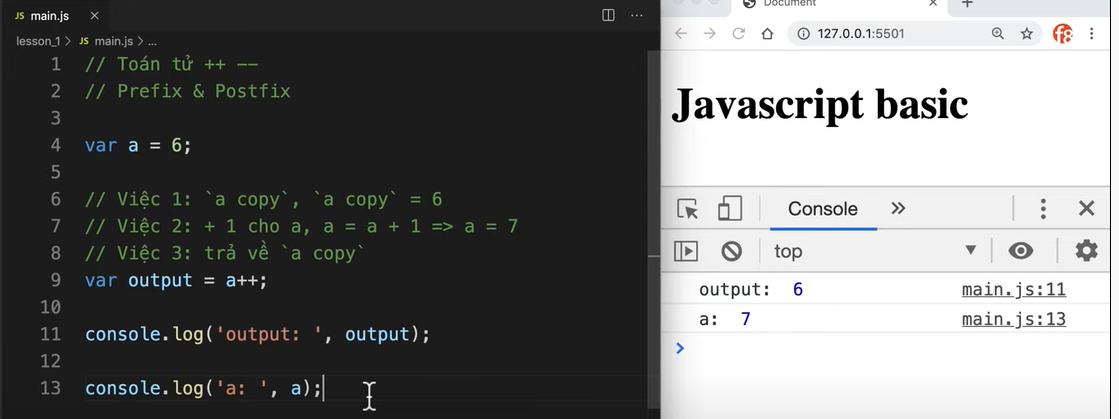
-Trong toán tử **++,** --

+Tiền tố:



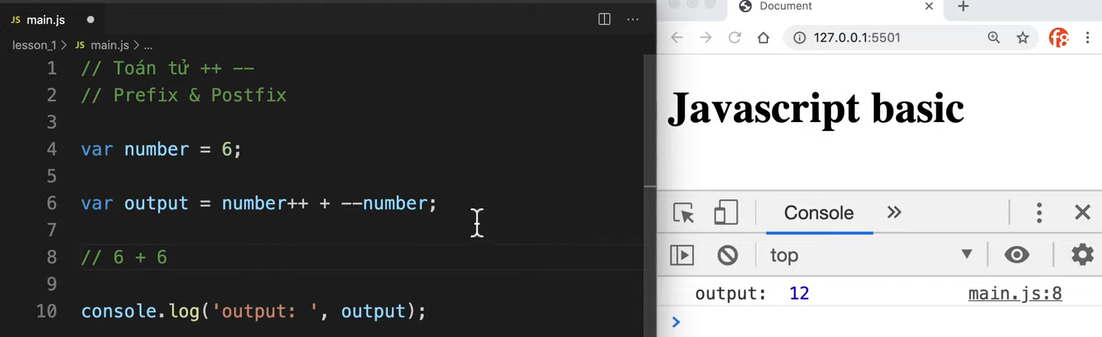
-Đối với --a; cũng giống v

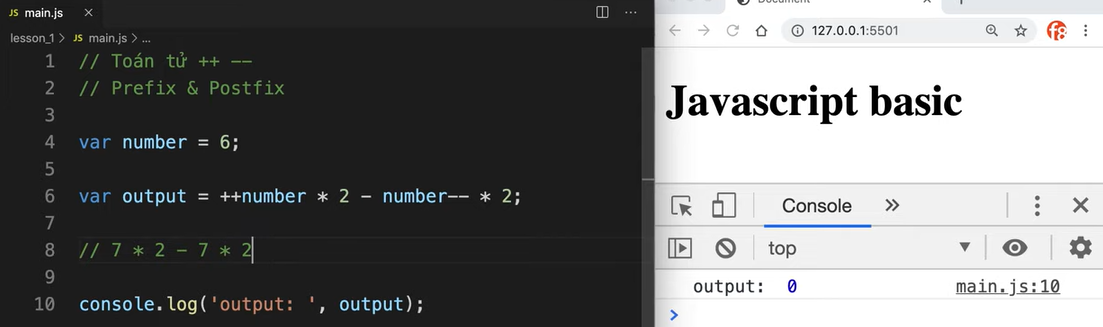
+Hậu tố:



-Đối với --a; cũng giống v

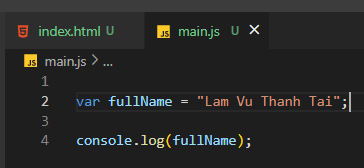
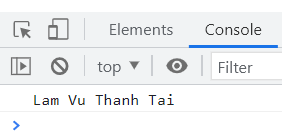
+VD:



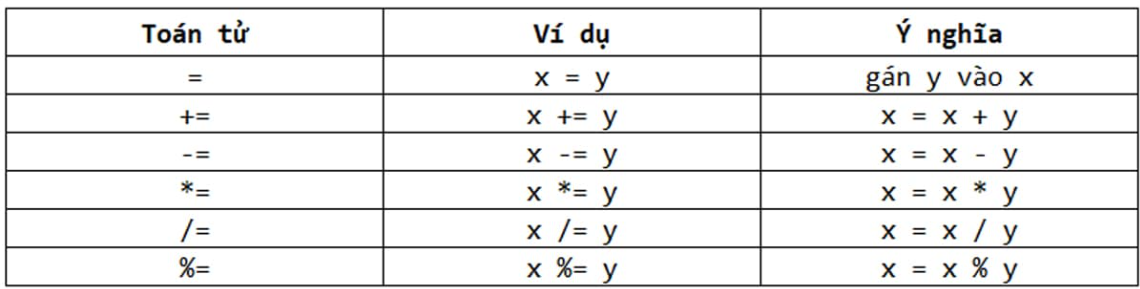


* 1. **Assignment (Gán)**

**- Toán tử gán** được dùng để **gán giá trị** ở bên phải toán tử vào biến ở bên trái toán tử.

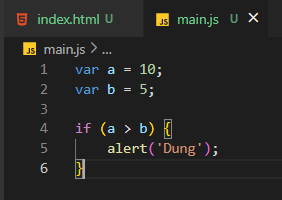
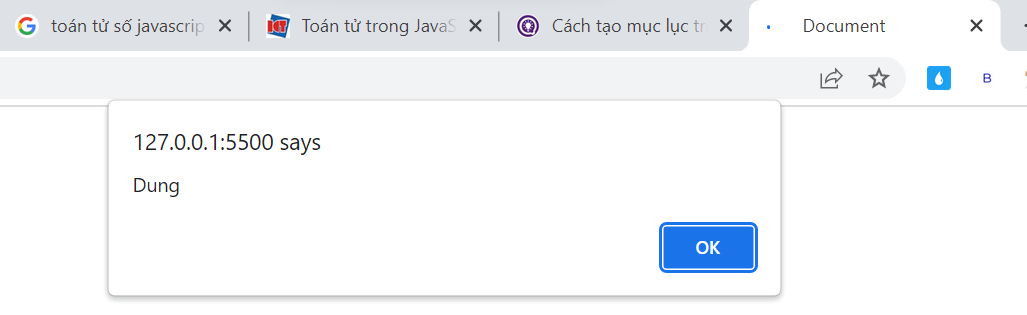
** **

-Dấu “ = ” là một trong những toán tử gán ! Ngoài ra còn:

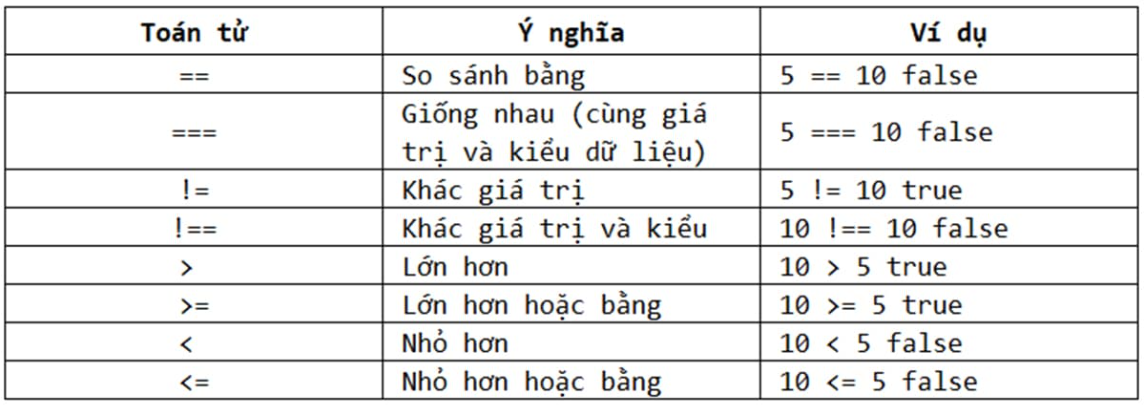


* 1. **Comparison (So sánh)**

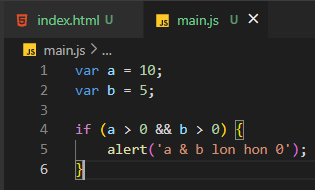
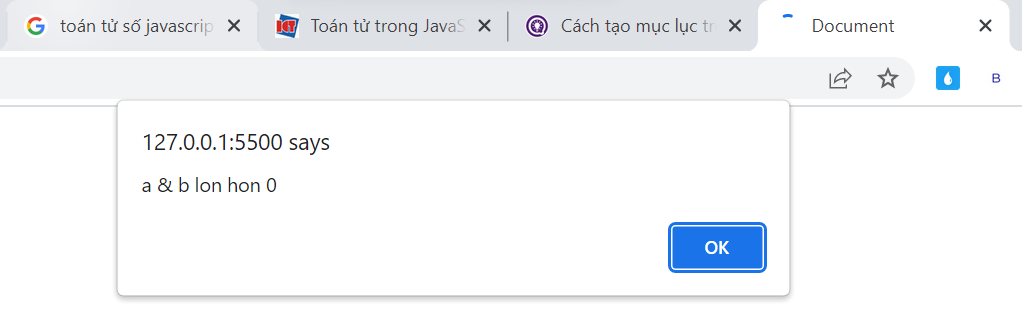
**- Toán tử so sánh trong JavaScript** sử dụng trong các biểu thức về logic để so sánh bằng, khác nhau, lớn hơn, nhỏ hơn.... Nó sẽ trả về giá trị true hoặc false.

**** ****

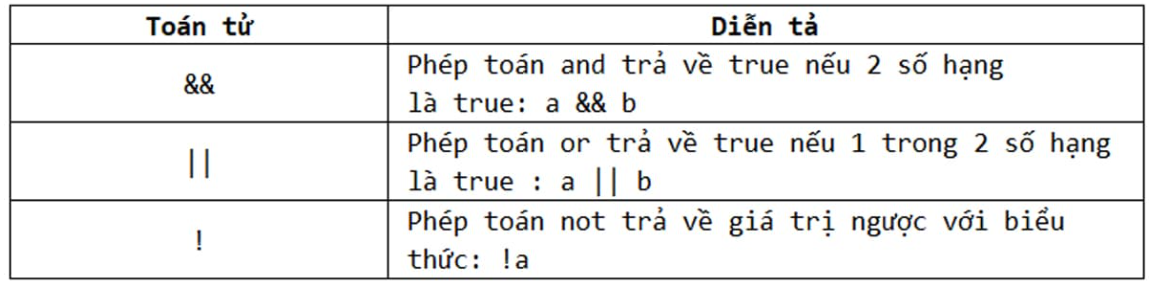
-Ngoài ra còn có các toán tử so sánh khác:

****

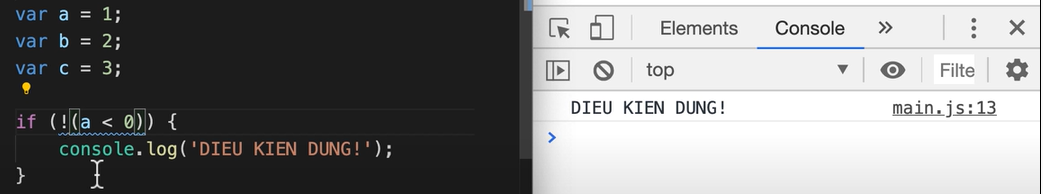
* 1. **Logic (logical)**

**** ****

**-** JavaScript cung cấp 3 loại **toán tử logic** gồm các phép toán: and, or, not

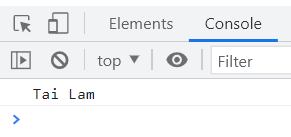
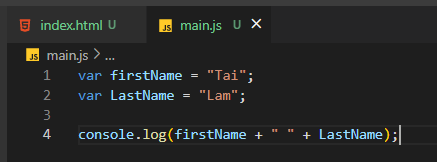
****

-Toán tử not: phủ định lại giá trị ban đầu

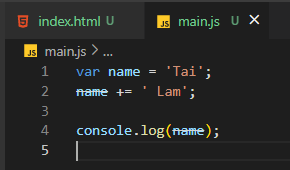
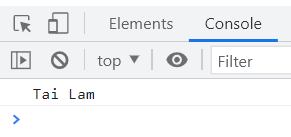
****

* 1. **String (chuỗi)**

**-**Trong JavaScript, chúng ta có thể sử dụng toán tử + để **nối hai chuỗi** lại với nhau.Nó có thể nối cả số vào chuỗi.

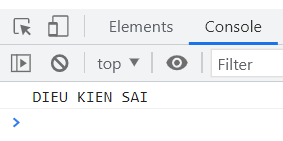
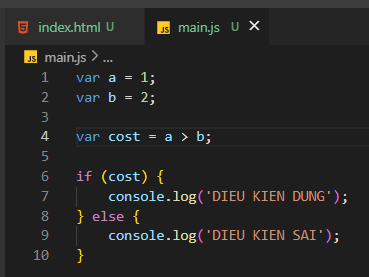
****

**-**Ngoài ra có thể kết hợp với toán tử gán

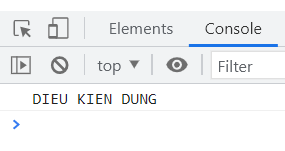
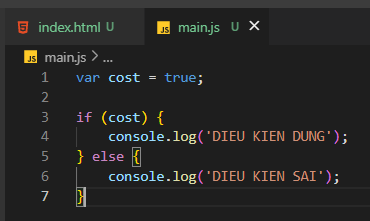
****

1. **Câu lệnh điều kiện If- else**

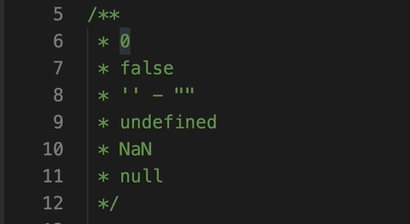
-Câu điều kiện Nếu – Thì, cấu trúc câu giống như ở các ngôn ngữ lập trình khác

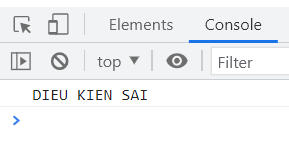
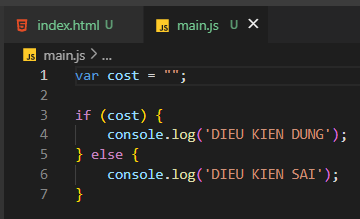
****

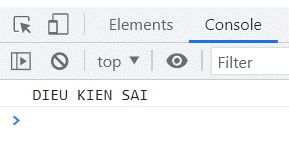
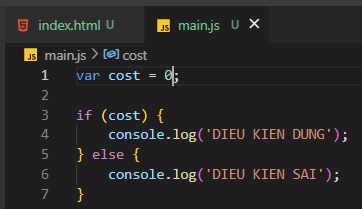
**-**Ngoài điều kiện là các toán tử ra thì còn nhận giá trị true, false



-6 giá trị khi sử dụng JS sẽ chuyển đổi thành false

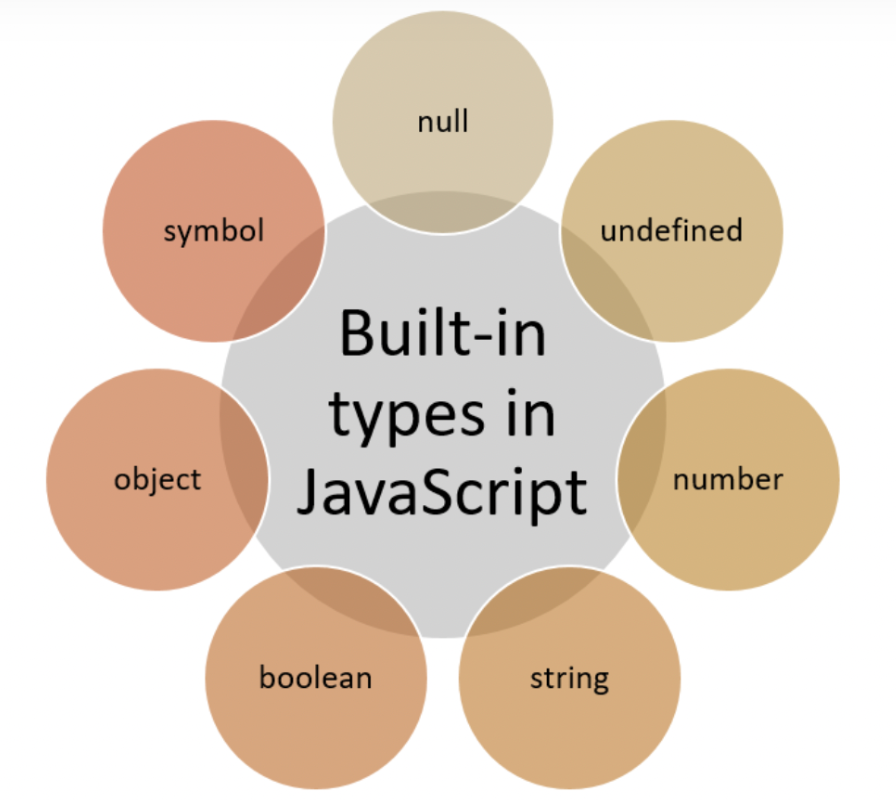






-Ngoài những giá trị này ra thì javascript sẽ convert (chuyển đổi) thành giá trị true

1. **Kiểu dữ liệu trong JS**

****

* 1. **Kiểu dữ liệu nguyên thủy (Primary Data)**
  2. **Kiểu dữ liệu phức tạp (Complex Data)**