

LOGIC BOOLEAN CƠ BẢN: TOÁN TỬ AND, OR & NOT

A AND B

“Sarah has a driver’s license
AND good vision”

		A	
		TRUE	FALSE
B	TRUE	TRUE	FALSE
	FALSE	FALSE	FALSE

Giá trị dương

Kết quả của toán tử phụ thuộc vào 2 biến

👉
true khi **TẤT CẢ** true

Không quan trọng có bao nhiêu biến

A OR B

“Sarah has a driver’s license
OR good vision”

		A	
		TRUE	FALSE
B	TRUE	TRUE	TRUE
	FALSE	TRUE	FALSE

👉
true khi có **MỘT CÁI** true

NOT A, NOT B



Đảo giá trị **true/false**

👉 **VÍ DỤ:**
A: Sarah has a driver’s license
B: Sarah has good vision

Biến Boolean có thể là TRUE hoặc FALSE



BIẾN BOOLEAN

👉 A: Age is greater or equal 20 false

👉 B: Age is less than 30 true

age = 16

HÃY DÙNG TOÁN TỬ!

👉 !A true
false

👉 A AND B false
false true

👉 A OR B true
false true

👉 !A AND B true
true true

👉 A OR !B false
false false

		A	
B	AND	TRUE	FALSE
	TRUE	TRUE	FALSE
	FALSE	FALSE	FALSE

		A	
B	OR	TRUE	FALSE
	TRUE	TRUE	TRUE
	FALSE	TRUE	FALSE

TÓM TẮT LỊCH SỬ JAVASCRIPT

1995

👉 Brendan Eich tạo ra **phiên bản JavaScript đầu tiên chỉ trong 10 ngày**. Nó được gọi là Mocha, nhưng đã có nhiều tính năng cơ bản của JavaScript hiện đại!



1996

👉 Mocha đổi thành LiveScript và sau đó là JavaScript, để thu hút các lập trình viên Java. Tuy nhiên, **JavaScript gần như không liên quan gì đến Java**. 🙅

👉 Microsoft khởi chạy IE, sao chép **JavaScript từ Netscape** và gọi nó là JScript;



1997

👉 Với nhu cầu chuẩn hóa ngôn ngữ, ECMA phát hành ECMAScript 1 (ES1), **tiêu chuẩn chính thức đầu tiên cho JavaScript** (ECMAScript là tiêu chuẩn, JavaScript là ngôn ngữ trong thực tế);



2009

👉 ES5 (ECMAScript 5) được phát hành với rất nhiều tính năng mới tuyệt vời;

2015

👉 ES6/ES2015 (ECMAScript 2015) đã được phát hành: **bản cập nhật lớn nhất cho ngôn ngữ này từ trước đến nay!**

👉 ECMAScript thay đổi **chu kỳ phát hành hàng năm** để chuyển ít tính năng hơn cho mỗi bản cập nhật

2016 – ∞

👉 Phát hành ES2016/ES2017/ES2018/ES2019/ES2020/ES2021/ ... /ES2089



TƯƠNG THÍCH NGƯỢC: KHÔNG PHÁ VỠ WEB!

```
// ES1 Code  
function add(n) {  
  var x = 5 + add.arguments[0];  
  return x;  
}
```

1997



TƯƠNG
THÍCH
NGƯỢC

Công cụ JavaScript
hiện đại

2020

ĐỪNG PHÁ VỠ WEB!

- 👉 Các tính năng cũ **không** bao giờ bị xóa;
- 👉 Không thực sự là phiên bản mới, chỉ là **bản cập nhật tĩnh tiến** (bản phát hành)
- 👉 Các trang web hoạt động **vĩnh viễn!**

Công cụ JavaScript
hiện đại

2020



~~KHÔNG
TƯƠNG THÍCH~~

```
// ES2089 Code 🤔  
c int add n <=> int 5 + n
```

2089

CÁCH SỬ DỤNG JAVASCRIPT HIỆN ĐẠI



Trong quá trình phát triển: Chỉ cần sử dụng Google Chrome mới nhất!



Trong quá trình sử dụng: Sử dụng Babel để transpile và polyfill code của bạn (chuyển đổi lại ES5 để đảm bảo khả năng tương thích với trình duyệt cho tất cả người dùng).

[illegible]

<http://kangax.github.io/compat-table>

ES5

👉 Được hỗ trợ đầy đủ trong tất cả các trình duyệt (xuống đến IE 9 từ năm 2011);

👉 **Sẵn sàng sử dụng ngay hôm nay**

BABEL

ES6/ES2015



ES2020

👉 **ES6+:** Được hỗ trợ trong tất cả trình duyệt hiện đại;

👉 Không hỗ trợ trong các trình duyệt cũ hơn :

👉 Có thể sử dụng **hầu hết** các tính năng trong sản xuất với quá trình chuyển đổi và polyfill 😊

ES2021 – ∞

👉 **ESNext:** Các phiên bản trong tương lai (các đề xuất tính năng mới đạt đến Giai đoạn 4);

👉 Có thể sử dụng **một số** tính năng trong khi sử dụng với quá trình transpiling và polyfilling.

Sẽ thêm video mới

(Tính đến năm 2020)

JAVASCRIPT HIỆN ĐẠI NGAY TỪ ĐẦU

🔥 Học **JavaScript hiện đại ngay từ đầu!**

👉 Tuy nhiên, hãy tìm hiểu một số điều từng được thực hiện **trước** JavaScript hiện đại (ví dụ `const` & `let` với `var` và hàm khởi tạo với ES6 `class`).

3 lý do chúng ta không nên quên **Good Ol' JavaScript**:

- 👉 Bạn sẽ hiểu rõ hơn JavaScript thực sự hoạt động như thế nào;
- 👉 Nhiều hướng dẫn và code bạn tìm thấy trực tuyến ngày nay vẫn còn trong ES5;
- 👉 Khi làm việc trên codebases cũ, chúng sẽ được viết bằng ES5.

