



THỰC HỌC – THỰC NGHIỆP

LẬP TRÌNH ECMASCRIPT

GIỚI THIỆU MÔN HỌC



- ⊙ Hiểu được khái niệm tổng quan về ECMAScript
- ⊙ Hiểu ECMAScript 6 và lý do cần sử dụng
- ⊙ Hiểu được cơ bản một vài tính năng mới
- ⊙ Sử dụng các công cụ và thư viện phát triển

- 📖 Tổng quan ECMAScript
- 📖 Giới thiệu về ES6
- 📖 Các công cụ và thư viện hỗ trợ ES6





PHẦN 1: TỔNG QUAN ECMASCRIPT

- ❑ **ECMA** được viết tắt từ European Computer Manufacturer's Association (Hiệp hội các nhà sản xuất máy tính Châu Âu)
- ❑ Tiêu chuẩn ECMA dựa trên một số công nghệ ban đầu, nổi tiếng nhất là JavaScript (Netscape) và JScript (Microsoft).
- ❑ *ECMAScript là một đặc tả chứa một loạt các yêu cầu đòi hỏi các nhà phát hành ngôn ngữ Script phải tuân theo.*

- ❑ Tháng 5 năm 1995, Javascript được phát triển bởi Brendan Eich của Netscape
- ❑ Ban đầu nó được đặt tên là Mocha, sau 4 tháng được đổi tên là LiveScript, và cuối cùng được gọi với tên là JavaScript.
- ❑ Năm 1997, phiên bản ECMAScript đầu tiên phát hành sau khi Netscape chuyển Javascript đến ECMA International để làm công tác chuẩn hóa và viết đặc tả.

Phiên bản	Tên chính thức
1	ECMAScript 1 (1997)
2	ECMAScript 2 (1998)
3	ECMAScript 3 (1999)
4	ECMAScript 4 – Không được phát hành
5	ECMAScript 5 (2009) – ES5
5.1	ECMAScript 5.1 (2011)
6	ECMAScript 2015 – ES6
7	ECMAScript 2016
8	ECMAScript 2017
9	ECMAScript 2018

ECMAScript thường được viết tắt thành ES.

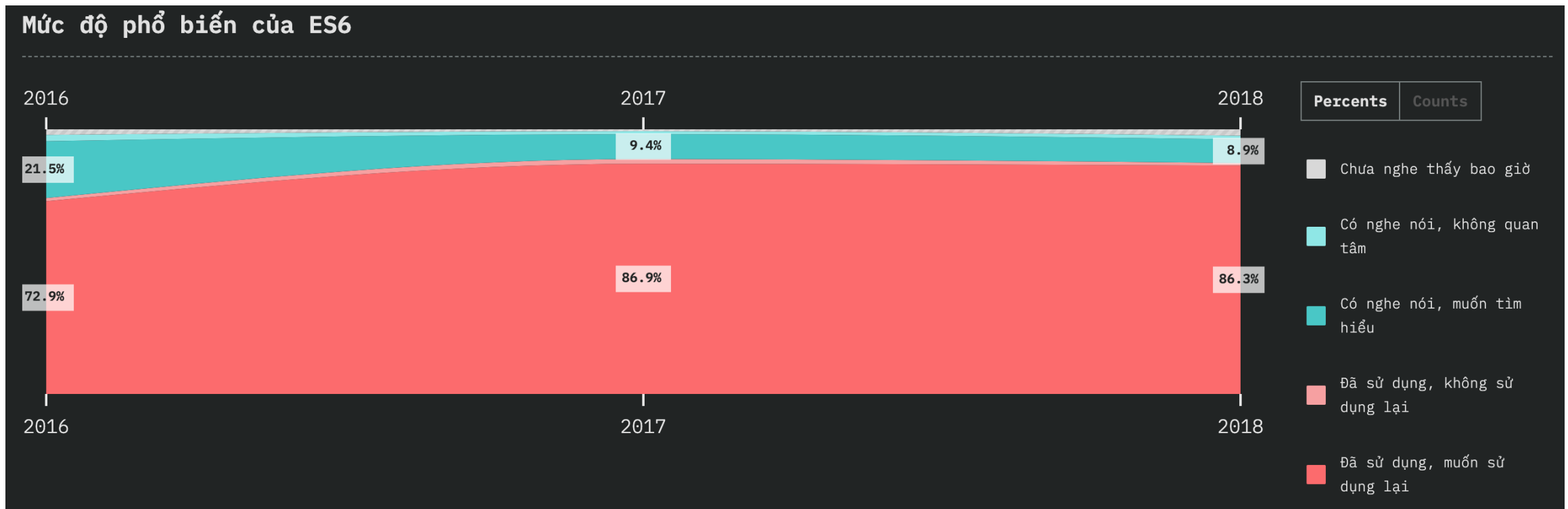


PHẦN 2: GIỚI THIỆU ES6

- ❖ ECMAScript 6 còn được gọi là ES6
- ❖ ES6 ra đời vào năm 2015 nên cái tên ES2015 được lấy làm tên chính thức với nhiều tính năng mới học hỏi các ngôn ngữ cấp cao khác
- ❖ ES6 là một phiên bản có rất nhiều cải tiến mạnh mẽ về cú pháp cũng như hiệu năng xử lý JS so với các phiên bản trước đó.



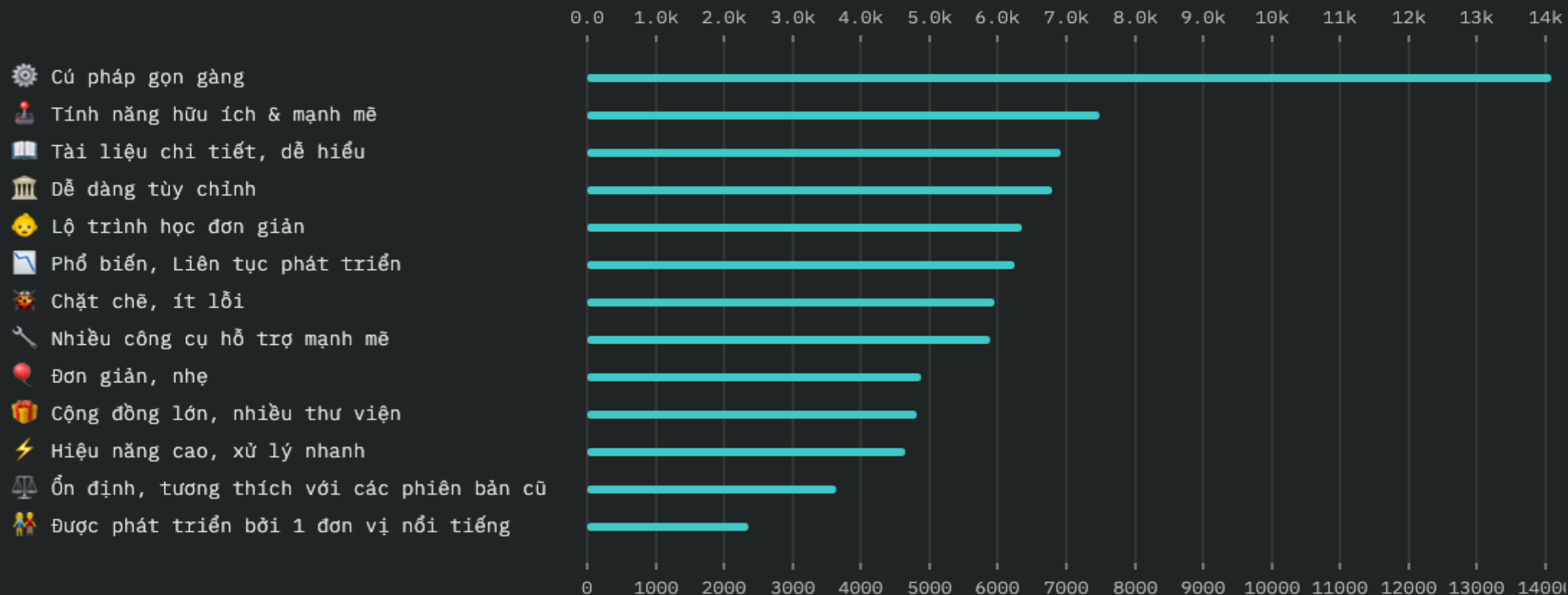
❑ Mức độ phổ biến của ES6 trong lập trình Javascript



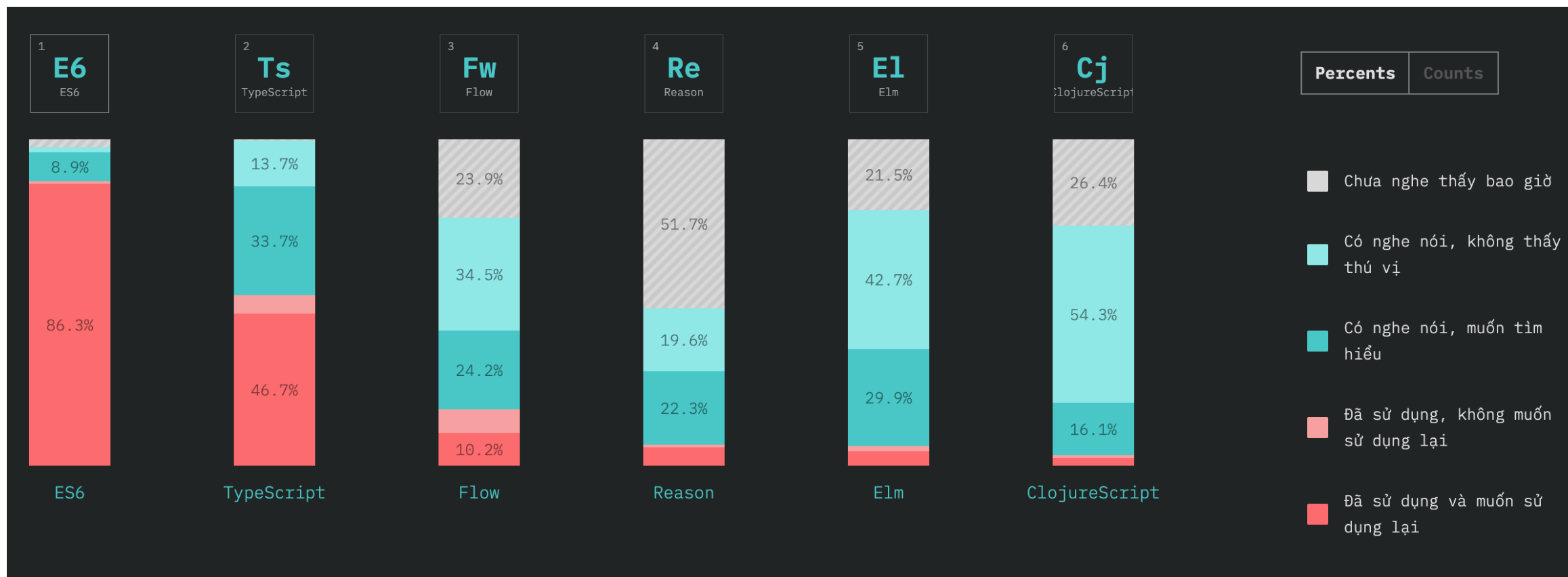
Điểm nổi bật của ES6

[Share](#)

Dưới đây là những điểm nổi bật được cộng đồng developer yêu thích của ES6



So sánh ES6 với các ngôn ngữ Script khác



Trình duyệt	
Safari 10	100%
Safari 9	54%
Chrome 57	97%
Firefox 52	94%
Edge 14	93%
Internet Explorer 11	11%
iOS WKWebView 10	100%
Android WebView 5+	97%

Link kiểm tra đầy đủ : <https://kangax.github.io/compat-table/es6/>

□ Template Strings

- ❖ Trích dẫn bằng `backtick`
- ❖ Có thể sử dụng biểu thức ngay bên trong string

```
// JS  
let name = 'Darryl';  
console.log(`Hello ${name}`);
```

- ❖ Có thể sử dụng nhiều dòng

```
// JS  
document.body.innerHTML = `

<h1>Intro to <span>ES6</span></h1>  
</div>`;


```

□ Let và Const

❖ Thêm 2 cách mới để khai báo biến trong Javascript ngoài **var**

```
// JS
function foo(count) {
  var i; // HOISTING
  for (i = 0; i < count; i++) {
    // Làm việc gì đó với biến i
  }

  console.log(i);
}

foo(5);
```

```
// JS
function foo(count) {
  for (let i = 0; i < count; i++) {
    // Làm việc gì đó với biến i
  }

  console.log(i);
}

foo(5);
```

❑ Default parameters (Tham số mặc định)

- ❖ Cung cấp giá trị mặc định cho một tham số nếu nó không được chỉ định

```
let counter = 0;
```

```
function increment(amount = 1) {  
    return counter += amount;  
}
```

```
increment();           //=> 1  
increment(2);          //=> 3  
increment();           //=> 4
```


❑ Destructuring

- ❖ Đơn giản hóa việc giải nén các giá trị khỏi một mảng hoặc một đối tượng

```
const months = ['January', 'February', 'March', 'April'];
```

```
let [jan, feb, mar, apr] = months;
```

```
console.log(jan); //=> "January"
```

```
let { first_name } = {first_name: 'Darryl', last_name: 'Pogue' };
```

```
console.log(first_name); //=> "Darryl"
```

□ Arrow function

- ❖ Hàm mũi tên là hàm ẩn danh (expression function) sử dụng "mũi tên" =>

```
let arr = [1,6,2,8,4,9,3];
```

```
// ES5
```

```
arr.sort(function sorter(a, b) {  
    return a - b;  
});
```

```
// ES6
```

```
arr.sort((a, b) => { return a - b });
```

❑ Cú pháp Classes mới

- ❖ Về mặt chức năng, class không khác gì cú pháp đã làm trong nhiều năm với ES5
- ❖ Cú pháp mới giúp code “nhìn ổn hơn”

ES6

```
class Person {
  constructor() {
    this.first_name = 'Darryl';
    this.last_name = 'Pogue';
  }

  get name() {
    return `${this.first_name} ${this.last_name}`;
  }

  greet() {
    console.log(`Hello ${this.name}`);
  }
}
```

ES5

```
function Person() {
  this.first_name = 'Darryl';
  this.last_name = 'Pogue';
}

Object.defineProperty(Person.prototype, 'name', {
  get: function() {
    return `${this.first_name} ${this.last_name}`;
  }
});

Person.prototype.greet = function() {
  console.log(`Hello ${this.name}`);
};
```

Ngoài ra ES6 còn cung cấp rất nhiều tính năng thú vị khác :

- ☐ Modules : Import & Export
- ☐ Spread operator
- ☐ Maps and Sets
- ☐ Async & Await
- ☐ Generator Functions
- ☐ WeakMap / WeakSet
- ☐ Reflect API
- ☐ Getter & Setter



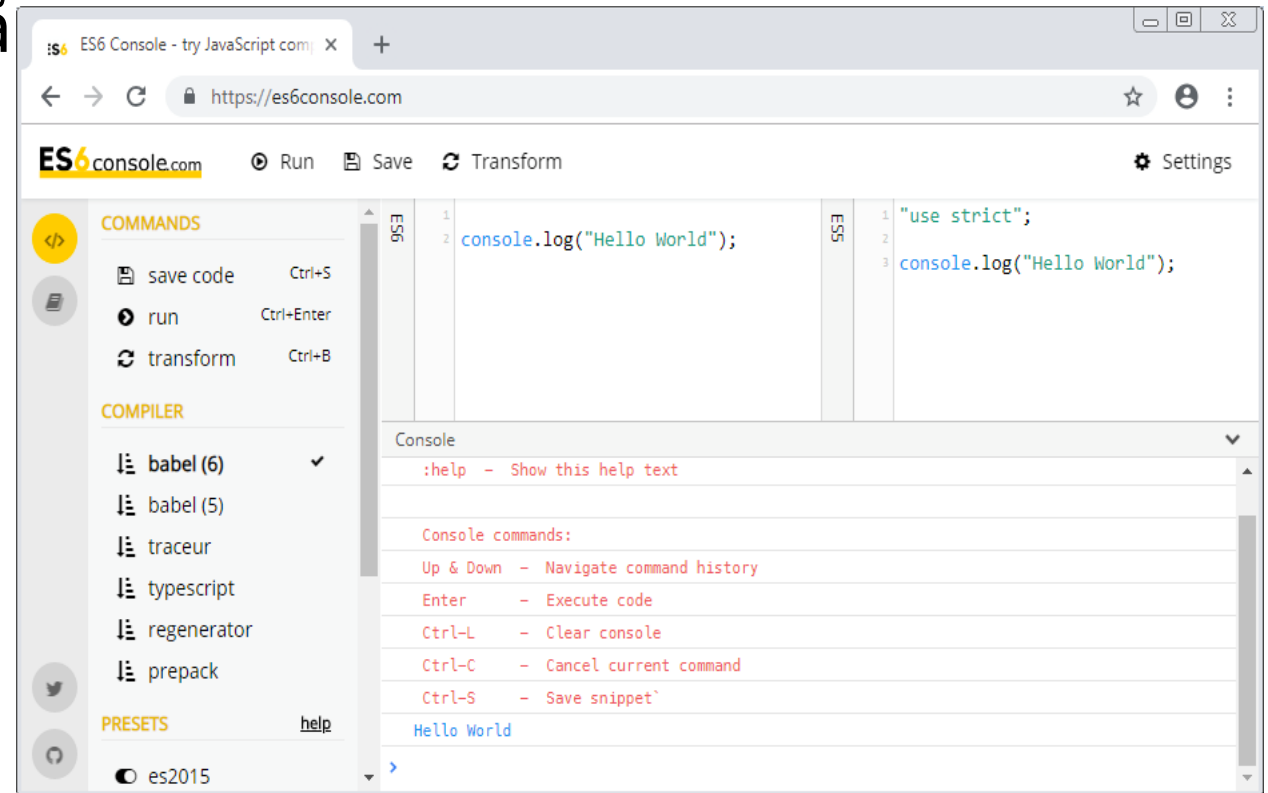
PHẦN 4:
CÔNG CỤ VÀ THƯ VIỆN HỖ TRỢ

1. Online Tool

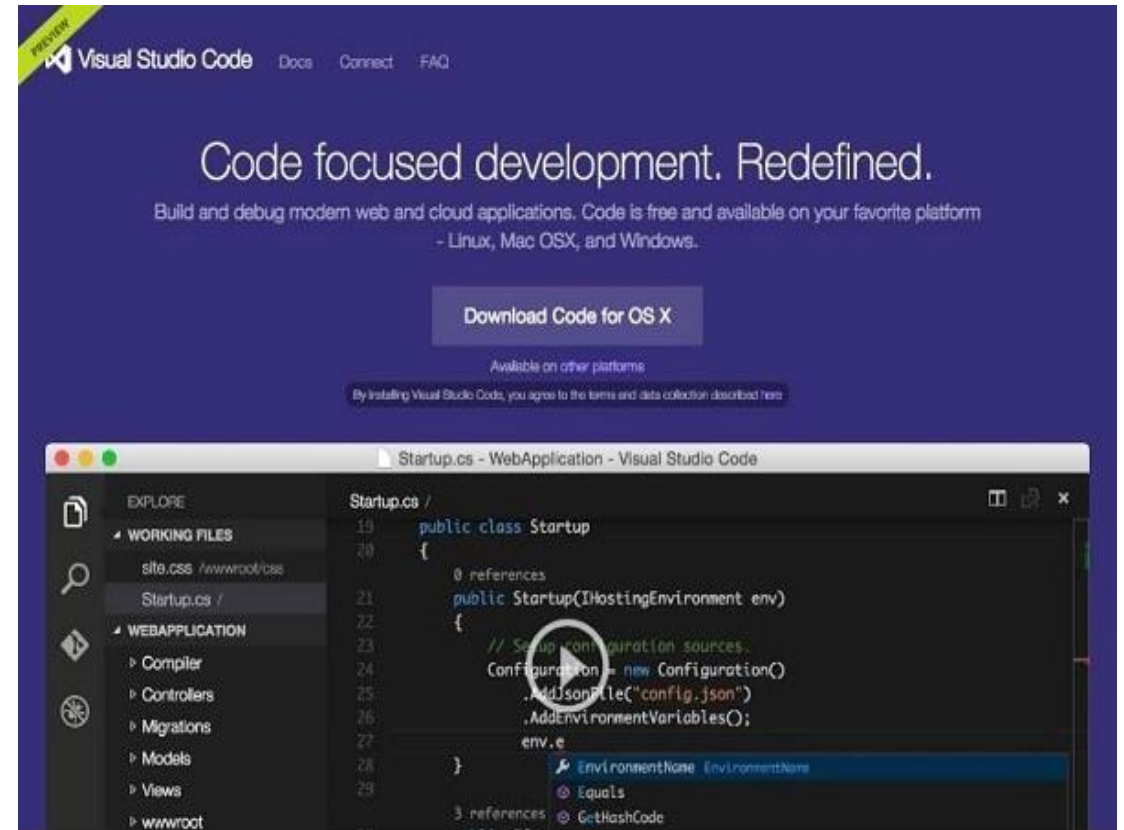
❖ Cách đơn giản nhất để viết mã ES6 là sử dụng một công cụ trực tuyến

❖ ví dụ :

➤ <https://es6console.com/>



- ❖ Hoặc bạn có thể dùng một Text Editor bất kỳ để viết mã **ECMAScript**, khuyến khích bạn sử dụng trình soạn thảo **Visual Studio Code**, đây là trình soạn thảo được cung cấp miễn phí.
- ❖ truy cập đường liên kết dưới để tải.
<https://code.visualstudio.com>



Khái niệm **Transpiler** (*transforming + compiler*) là một cách gọi các công cụ có tác dụng chuyển đổi code từ một phiên bản A về một phiên bản B.

- ❖ Babel là một **transpiler** có tác dụng chuyển đổi code ES6 về code ES5. vì phiên bản ES5 đã được hỗ trợ trên hầu hết tất cả các trình duyệt.
- ❖ Babel dành cho các developer muốn đưa các tính năng mới nhất của ES6 vào ứng dụng của mình nhưng vẫn cần chạy các tính năng này trên tất hầu hết các trình duyệt.



Ví dụ bạn có đoạn code ES6 như sau:

```
var sum = (a, b) => {  
  return a + b;  
}
```

Sau khi babel.js dịch thành ES5 để có thể chạy tốt trên trình duyệt cũ

```
var sum = function sum(a, b) {  
  return a + b;  
};
```

- ☑ Tổng quan ECMAScript
- ☑ Giới thiệu về ES6 và các tính năng mới trong ES6
- ☑ Các công cụ và thư viện hỗ trợ ES6





thank
you!