Tên nhóm: 7

Thành viên

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và tên | Email | SDT |
| 1 | Nguyễn Đức Lai | nguyenduclai1997@gmail.com | 0338001520 |
| 2 | Nguyễn Đăng Khoa | jeremi220497@gmai.com | 0967286764 |
|  |  |  |  |

Email nhóm: didong3nhom7@gmail.com

Pass: Didong123456

Đề tài:

Tìm hiểu : Kiến thức cơ bản về ES6

1. Biến Let, Var, Const trong ES6 (Nguyễn Đức Lai)

* Biến let có phạm vi sử dụng trong { } ở ngoài không thể sử dụng biến let

Ví dụ:

function foo() {

let x = 10;

if (true) {

var x = 20; *// x ở đây cũng là x ở trên*

console.log(x); *// in ra 20*

}

console.log(x); *// vẫn là 10*

}

Biến let không tạo ra thuộc tính mới cho global scope(this) //global scope là không nằm trong function nào.

let y = 'global';

console.log(this.y); *// undefined*

* Biến var có phạm vi sử dụng trong 1 function chứa nó.

Ví dụ:

function foo() {

var x = 10;

if (true) {

var x = 20; *// x ở đây cũng là x ở trên*

console.log(x); *// in ra 20*

}

console.log(x); *// vẫn là 20*

}

Biến var tạo thuộc tính mới cho global scope (this)

let y = 'global';

console.log(this.y); *// global*

* Biến const sử dụng để khai báo 1 hằng số mà trong quá trình chạy chương trình giá trị không thay đổi.

Ví dụ:

*const pi = 3.14;*

*pi = 1; // Lỗi Uncaught TypeError: Assignment to constant variable*

1. Kiểu dữ liệu Object (Phan Văn Khải)

JavaScript hỗ trợ mở rộng các kiểu dữ liệu.

Một **đối tượng** là một cá thể chứa một tập hợp các cặp giá trị khóa. Không giống như các kiểu dữ liệu nguyên thủy, các đối tượng có thể biểu thị các giá trị nhiều hoặc phức tạp và có thể thay đổi theo thời gian của chúng. Các giá trị có thể là các giá trị vô hướng hoặc các hàm hoặc thậm chí là các mảng của các đối tượng khác.

Các biến thể cú pháp để xác định một đối tượng được thảo luận thêm.

Khởi tạo đối tượng

Giống như các kiểu nguyên thủy, các đối tượng có một cú pháp theo nghĩa đen: **xoăn bracesv** ({và}). Sau đây là cú pháp để xác định một đối tượng.

var identifier = {

Key1:value, Key2: function () {

//functions

},

Key3: [“content1”,” content2”]

}

Nội dung của một đối tượng được gọi là **các thuộc tính** (hoặc các thành viên), và các thuộc tính bao gồm một **tên** (hoặc khóa) và **giá trị** . Tên thuộc tính phải là chuỗi hoặc ký hiệu và giá trị có thể là bất kỳ loại nào (kể cả các đối tượng khác).

Giống như tất cả các biến JavaScript, cả tên đối tượng (có thể là biến bình thường) và tên thuộc tính phân biệt chữ hoa chữ thường. Bạn truy cập các thuộc tính của một đối tượng với một dấu chấm đơn giản.

Sau đây là cú pháp để truy cập Object Properties.

objectName.propertyName

Ví dụ: Trình khởi tạo đối tượng

var person = {

firstname:"Tom",

lastname:"Hanks",

func:function(){return "Hello!!"},

};

//access the object values

console.log(person.firstname)

console.log(person.lastname)

console.log(person.func())

Ví dụ trên, định nghĩa một đối tượng. Đối tượng có ba thuộc tính. Thuộc tính thứ ba đề cập đến một hàm.

Đầu ra sau được hiển thị khi thực hiện thành công mã trên.

Tom

Hanks

Hello!!

Trong ES6, gán giá trị thuộc tính khớp với tên thuộc tính, bạn có thể bỏ qua giá trị thuộc tính.

Thí dụ

var foo = 'bar'

var baz = { foo }

console.log(baz.foo)

Đoạn mã trên định nghĩa một đối tượng **baz** . Đối tượng có một thuộc tính **foo**. Giá trị thuộc tính được bỏ qua ở đây vì ES6 ngầm gán giá trị của biến foo cho khóa foo của đối tượng.

Sau đây là ES5 tương đương với mã trên.

var foo = 'bar'

var baz = { foo:foo }

console.log(baz.foo)

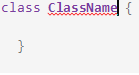
Đầu ra sau được hiển thị khi thực hiện thành công mã trên.

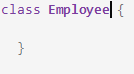
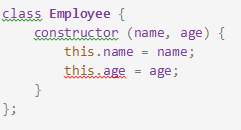
bar

Với cú pháp viết tắt này, công cụ JS nhìn vào phạm vi chứa cho một biến có cùng tên. Nếu nó được tìm thấy, giá trị của biến đó được gán cho thuộc tính. Nếu nó không được tìm thấy, một lỗi tham chiếu được ném.

1. Khai báo Class trong ES6 ( Nguyễn Đăng Khoa)

- Để tạo một đối tượng được khai báo theo chuẩn ES6 thì sử dụng cú pháp như sau:



Trong đó ClassName là tên của đối tượng được khởi tạo.  
VD: Tạo đối tượng Employee:  
  
- Để khai báo constructor trong ES6 thìchỉ cần khai báo một phương thức có tên là constructor :  


- Cách khai báo getter và setter



- Để kế thừa từ đối tượng khác, chúng ta sử dụng từ khóa extends  


- Để thực hiện gọi các phương thức trong lớp cha khi đang ở lớp con, mà phương thức đó đã bị rewritetrong lớp con rồi thì các bạn sử dụng keyword super  


- Để khai báo các thành phần tĩnh cho đối tượng, sử dụng từ khóa static ở trước tên phương thức.  


- Để khởi tạo đối tượng được khai báo theo chuẩn ES6 thì các bạn sử dụng từ khóa new với cú pháp như sau:  
