JavaScript

JavaScript web HTML CSS JavaScript JavaScript Java ECMA ECMA-262 ECMAScript 6 2015 JavaScript 1. JavaScript JavaScript JavaScript JavaScript

- JavaScript
  - JS call apply bind
  - JavaScript

JavaScript

JavaScript JavaScript JavaScript
1998 ECMAScript 2 1999 ECMAScript 3 try/catch ECMAScript 4 2009 ECMAS
mode" JSON 2011 ECMAScript 5.1 2015 ECMAScript 6 2016 ECMAScript 7
Array.prototype.includes

JavaScript

JS call apply bind

> JS call apply bind

this this

```
box.onclick = function(){
  function fn(){
    alert(this);
  }
  fn();
};
```

this box, Window

```
box.onclick = function(){
  var _this = this;
  function fn(){
    alert(_this);
  }
  fn();
};
```

this

call apply bind

this

```
box.onclick = function(){
  function fn(){
    alert(this);
  }
  fn();
};
```

```
box.onclick = function(){
  function fn(){
    console.log(this);
  }
  fn.call(this);
};
```

call this

```
fn.call(this);
```

this fn this box javscript box fn this

```
box.onclick = function(){
  function fn(){
    console.log(this);
  }
  fn.call(this);
};
```

```
box.onclick = function(){
  var fn = function(){
    console.log(this); //box
  }.call(this);
};
```

```
box.onclick = function(){
  (function(){
    console.log(this);
  }.call(this)); //box
};
```

```
var objName = {name:'JS2016'};
var obj = {
  name:'0 _ 0',
  sayHello:function(){
    console.log(this.name);
  }.bind(objName)
};
obj.sayHello();//JS2016
```

call apply bind this

```
function fn(a,b,c,d){
  console.log(a,b,c,d);
}

//call
fn.call(null,1,2,3);

//apply
fn.apply(null,[1,2,3]);

//bind
var f = fn.bind(null,1,2,3);
f(4);
```

```
1 2 3 undefined
1 2 3 undefined
1 2 3 4
```

null

call apply bind call apply call apply bind this

```
var objName = {name:'JS2016'};
var obj = {
  name:'0 _ 0',
  sayHello:function(){
    console.log(this.name);
  }.bind(objName)
```

```
};
obj.sayHello();//JS2016
```

bind call sayHello obj sayHello

Math.max

```
Math.max(10,6)
```

apply

```
var arr = [1,2,30,4,5];
console.log(Math.max.apply(null,arr));
```

```
function fn(){
  [].push.call(arguments,3);
  console.log(arguments); //[1, 2, 3]
}
fn(1,2);
```

```
var arr = ['aaabc'];
console.log(''.indexOf.call(arr,'b')); //3
```

```
var str = 'aaabc';
console.log(str.indexOf('b'));
```

```
var arr = ['aaabc'];
''.indexOf.call(arr);
```

arr indexOf 100%

```
''.indexOf.call(arr,'b')
```

arr ['aaabc'] 'aaabc'

```
'aaabc'.indexOf('b');
```

bind

```

Function.prototype.$bind = function(obj){
```

```
    //    this
  var _this = this;
    //
  var a = [].slice.call(arguments,1);
    //
  return function(){
    //          this  obj            concat                          f(4)           a  argument
    _this.apply(obj,a.concat([].slice.call(arguments)));
  };
};
```

bind

```
var arr = ['JSS'];

var index = ''.indexOf.$bind(arr,'S');
console.log(index())

------------------

function fff(){
  [].push.$bind(arguments,1);
  console.log(arguments);
}

fff();
```

bind

JavaScript

> JavaScript

- [Basic]
- [Basic]
- [Basic]
- [Basic]
- [Basic] ES6

JavaScript JavaScript number string boolean undefined null object symbol (ES6)

number string boolean symbol primiti object undefined null ype

object :

- object
- array
- function

- typeof
- instanceof
- Object.prototype.toString

typeof

```
typeof 123 // "number"
typeof NaN // "number"

typeof "abc" // "string"

typeof true // "boolean"

typeof function foo() {} // "function"

typeof undefined // "undefined"
```

```
typeof Symbol('abc') // "symbol"
```

```
typeof {a: 1} // "object"
typeof [1, 2] // "object"
typeof null // "object"
```

js 1-3

- 000
- 010
- 100
- 110
- 1

`null` `typeof` `object` `undefined` −2^30

instanceof

`instanceof` prototype

```
function instance_of(L, R) {//L      R
 var O = R.prototype;//   R
 L = L.__proto__;//   L
 while (true) {
   if (L === null)
     return false;
   if (O === L)//       O     L     true
     return true;
   L = L.__proto__;
 }
}
```

> “ https://www.ibm.com/developerworks/cn/web/1306_jiangjj_jsinstanceof/

```
'abc' instanceof String; // false,
123 instanceof Number; // false
true instanceof Boolean; // false

//
```

```
(new String('abc')) instanceof String; // true
(new String('abc')) instanceof Object; // true
String instanceof Object; // true

let person = function () {};
let Amy = new person();
Amy instanceof person; // true

[1,2] instanceof Array; // true
[1,2] instanceof Object; // true
```

`instanceof` `new` js __proto__ prototype

> “ https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/instanceof

Object.prototype.toString

`Object.prototype.toString.call(arguments)` `[object xxx]` `xxx`

```
Object.prototype.toString.call(123) // "[object Number]"

Object.prototype.toString.call('1bc') // "[object String]"

Object.prototype.toString.call({a:'a'}) // "[object Object]"

Object.prototype.toString.call([1,'a', true]) // "[object Array]"

Object.prototype.toString.call(true) // "[object Boolean]"

Object.prototype.toString.call(() => {}) // "[object Function]"

Object.prototype.toString.call(null) // "[object Null]"

Object.prototype.toString.call(undefined) // "[object Undefined]"

Object.prototype.toString.call(Symbol(1)) // "[object Symbol]"

Object.prototype.toString.call(new Date) // "[object Date]"
```

```
Object.prototype.toString.call(/\w+/ig) // "[object RegExp]"

Object.prototype.toString.call(new Error) // "[object Error]"
```

`JavaScript`

-

```
//
var name = "John";

//
function sayhi () {
    return console.log("Hi!");
}
```

```
function sayhi () {
    var saying = "Hi, John.";
    console.log(saying);
}
console.log(saying); //
```

ES6`let``const`

```
let x = 1;
{
  let x = 2, y=3;
}
console.log(x); // 1
console.log(y); //           y
```

> `const`

> 1. https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Statements/block

> 2. https://msdn.microsoft.com/zh-cn/library/bzt2dkta(v=vs.94).aspx

Node.js require global

```
//
const file = require('./1.js');

//
global.web = "Nodejs";
```

> “ http://nodejs.cn/api/globals.html

> “ stack heap

```
var word = 'hello';
str[0] = 'y';

console.log(str); // hello
```

JavaScript

```
var num1 = 10;
var num2 = num1;

num1 += 5;

console.log(num1); // 15
console.log(num2); // 20
```

> “

```
var a = {age: 12};
```

```
var b = a;

a. name = 'John'

a. age // 12
b. age // 12
b. name // 'John'
```

> 1. https://juejin.im/post/59ac1c4ef265da248e75892b
> 2. https://segmentfault.com/a/1190000008637489

3. \

(`malloc` `free` " " garbage collector

```
var n = 123; //
var s = "azerty"; //

//                          o
//
var o = {
  a: {
    b: 2
  }
};

var o2 = o; // o2      “    ”

o = 1;       //    “    ”     o o2

var oa = o2.a; //   “    ” a
//    “    ”         o2    oa

o2 = "yo"; //
           //
           //      a     oa

oa = null; // a
           //
```

> “ https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Memory_Management

ES6

ECMAScript 6

ES6 http://kangax.github.io/compat-table/es6/

ES6 https://www.css88.com/archives/9922