

ES6新增数据类型symbol



jCodeLife

0.477

2020.09.05 22:09:16 字数 566 阅读 70

编辑文章

ES6 引入了一种新的原始数据类型 `symbol`，表示独一无二的值。可以通过 `typeof` 检测出来。

所以现在通过 `typeof` 能检测出 7 种数据类型类型了哦，分别是：`number`、`string`、`boolean`、`undefined`、`object`、`function` 以及新增的 `symbol`

`symbol` 类型的变量通过 `Symbol` 函数产生

```
1 let sym = Symbol();
2 typeof sym;//symbol
```

`Symbol` 是一个函数，但是不是构造函数，所以不能 `new`；如果使用 `new` 会报错：`Uncaught TypeError: Symbol is not a constructor`

```
1 typeof Symbol//function
2 new Symbol//Uncaught TypeError: Symbol is not a constructor
```

`symbol` 可以作为对象的 `key` 来使用，可以保证不会与其他属性名产生冲突；但是把变量用 `[]` 括起来

```
1 let sym1 = Symbol();
2 let sym2 = Symbol();
3 let obj = {
4   [sym1]: 'aaa'
5 }
6 obj[sym2] = "bbb"
7 console.log(obj[sym1])//aaa
8 console.log(obj[sym2])//bbb
```

`symbol` 类型的变量通过 `Symbol` 产生，特点是独一无二，所以不相等

```
1 let sym1 = Symbol();
2 let sym2 = Symbol();
3 sym1 == sym2 //false
```

为了好区分 `symbol` 变量，所以 `Symbol` 函数可以接收一个参数，用于表示 `symbol` 变量的相关描述信息

```
1 let s1 = Symbol('这是s1');
2 let s2 = Symbol('这是s2');
3 console.log(s1.toString());//'Symbol(这是s1)'
4 console.log(s2.toString());//'Symbol(这是s2)'
```

就算相同描述信息产生的 `symbol` 变量也是不相等

```
1 Symbol('aaa') == Symbol('aaa')//false
```

推荐阅读

如何理解JavaScript中的对象？

阅读 203

js 判断字符串中是否包含某个字符串

阅读 143

vue数据更新了，视图不更新，怎么解决？

阅读 294

vue sku商品

阅读 207

JavaScript基础篇（三）作用域和闭包

阅读 57

symbol变量作为属性名，通过 `for...in`、`for...of` 以及 `Object.keys()`、`Object.getOwnPropertyNames()`、`JSON.stringify()` 都不能遍历出来，但它也不是私有属性，有一个 `Object.getPrototypeOfSymbols()` 方法，可以获取对象的所有 symbol 类型的属性名

```
1 let a = Symbol('a');
2 let b = Symbol('b');
3 const obj = {
4   [a]: 'apple',
5   [b]: 'banana'
6 };
7 const objectSymbols = Object.getPrototypeOfSymbols(obj);
8 console.log(objectSymbols) // [Symbol(a), Symbol(b)]
```

有的时候，我们希望描述相同对应的值就相同，这也是能做到的，使用 `Symbol.for()`，它接受一个字符串作为参数，然后搜索有没有以该参数作为名称的 Symbol 值。如果有，就返回这个 Symbol 值，否则就新建一个以该字符串为名称的 Symbol 值，并将其注册到全局

```
1 let s1 = Symbol.for('aaa');
2 let s2 = Symbol.for('aaa');
3 s1 === s2 // true
```

所以多次调用调用 `Symbol.for("xxx")`，每次都会返回同一个 Symbol 值，但是多次调用 `Symbol("xxx")`，会返回多个不同的 Symbol 值；

可以使用 `Symbol.keyFor()` 返回一个已登记的 Symbol 类型值的 key；未登记的 Symbol 值，所以返回 undefined

```
1 let s1 = Symbol.for("foo");
2 Symbol.keyFor(s1) // "foo"
3 let s2 = Symbol("foo");
4 Symbol.keyFor(s2) // undefined
```

以上就是symbol相关的内容，Symbol身上还有其他的東西，但是我觉得不太常用，所以先不列出来了！



3人点赞 >



笔记 (2)



"小礼物走一走，来简书关注我"

赞赏支持

还没有人赞赏，支持一下



jCodeLife 书山有路勤为径，学海无涯苦作舟

总资产30 (约2.22元) 共写了13.6W字 获得330个赞 共25个粉丝

被以下专题收入，发现更多相似内容

投稿管理

收入我的专题

前端开发那些事 II

写下你的评论...

评论 0

赞 3

...

推荐阅读

如何理解JavaScript中的对象？

阅读 203

js 判断字符串中是否包含某个字符串

阅读 143

vue数据更新了，视图不更新，怎么解决？

阅读 294

vue sku商品

阅读 207

JavaScript基础篇（三）作用域和闭包

阅读 57



推荐阅读

如何理解JavaScript中的对象?

阅读 203

js 判断字符串中是否包含某个字符串

阅读 143

vue数据更新了，视图不更新，怎么解决?

阅读 294

vue sku商品

阅读 207

JavaScript基础篇（三）作用域和闭包

阅读 57



写下你的评论...

评论0

赞3

