

# Object.isFrozen()

**Object.isFrozen()** 方法判断一个对象是否被冻结 [frozen](#)。

## 语法

```
Object.isFrozen(obj)
```

## 参数

**obj**

被检测的对象

## 描述

一个对象是冻结的（frozen）是指它不可 [扩展](#)，所有属性都是不可配置的（non-configurable），且所有数据属性（data properties）都是不可写的（non-writable）。数据属性是指那些没有取值器（getter）或赋值器（setter）的属性。

## 例子

```
1 // 一个对象默认是可扩展的,所以它也是非冻结的.
2 assert(Object.isFrozen({}) === false);
3
4 // 一个不可扩展的空对象同时也是一个冻结对象.
5 var vacuouslyFrozen = Object.preventExtensions({});
6 assert(Object.isFrozen(vacuouslyFrozen) === true);
7
8 // 一个非空对象默认也是非冻结的.
9 var oneProp = { p: 42 };
10 assert(Object.isFrozen(oneProp) === false);
11
12 // 让这个对象变的不可扩展,并不意味着这个对象变成了冻结对象,
13 // 因为p属性仍然是可以配置的(而且可写的).
14 Object.preventExtensions(oneProp);
15 assert(Object.isFrozen(oneProp) === false);
16
```

```
17 // ...如果删除了这个属性,则它会成为一个冻结对象.
18 delete oneProp.p;
19 assert(Object.isFrozen(oneProp) === true);
20
21 // 一个不可扩展的对象,拥有一个不可写但可配置的属性,则它仍然是非冻结的.
22 var nonWritable = { e: "plep" };
23 Object.preventExtensions(nonWritable);
24 Object.defineProperty(nonWritable, "e", { writable: false }); // 变得不可
25 assert(Object.isFrozen(nonWritable) === false);
26
27 // 把这个属性改为不可配置,会让这个对象成为冻结对象.
28 Object.defineProperty(nonWritable, "e", { configurable: false }); // 变得
29 assert(Object.isFrozen(nonWritable) === true);
30
31 // 一个不可扩展的对象,拥有一个不可配置但可写的属性,则它仍然是非冻结的.
32 var nonConfigurable = { release: "the kraken!" };
33 Object.preventExtensions(nonConfigurable);
34 Object.defineProperty(nonConfigurable, "release", { configurable: false });
35 assert(Object.isFrozen(nonConfigurable) === false);
36
37 // 把这个属性改为不可写,会让这个对象成为冻结对象.
38 Object.defineProperty(nonConfigurable, "release", { writable: false });
39 assert(Object.isFrozen(nonConfigurable) === true);
40
41 // 一个不可扩展的对象,值拥有一个访问器属性,则它仍然是非冻结的.
42 var accessor = { get food() { return "yum"; } };
43 Object.preventExtensions(accessor);
44 assert(Object.isFrozen(accessor) === false);
45
46 // ...但把这个属性改为不可配置,会让这个对象成为冻结对象.
47 Object.defineProperty(accessor, "food", { configurable: false });
48 assert(Object.isFrozen(accessor) === true);
49
50 // 使用Object.freeze是冻结一个对象最方便的方法.
51 var frozen = { 1: 81 };
52 assert(Object.isFrozen(frozen) === false);
53 Object.freeze(frozen);
54 assert(Object.isFrozen(frozen) === true);
55
56 // 一个冻结对象也是一个密封对象.
57 assert(Object.isSealed(frozen) === true);
58
59
```

```
60 | // 当然,更是一个不可扩展的对象.  
    |  
    | assert(Object.isExtensible(frozen) === false);  
    |  
    |
```

## 注意

在 ES5 中, 如果参数不是一个对象类型, 将抛出一个 `TypeError` 异常。在 ES6 中, `non-object` 参数将被视为一个冻结的普通对象, 因此会返回 `true`。

```
1 | Object.isFrozen(1);  
2 | // TypeError: 1 is not an object (ES5 code)  
3 |  
4 | Object.isFrozen(1);  
5 | // true                                     (ES2015 code)
```

## 规范

Specification	Status	Comment
<a href="#">ECMAScript 5.1 (ECMA-262)</a> Object.isFrozen	<div>ST</div> Standard	Initial definition. Implemented in JavaScript 1.8.5
<a href="#">ECMAScript 2015 (6th Edition, ECMA-262)</a> Object.isFrozen	<div>ST</div> Standard	
<a href="#">ECMAScript Latest Draft (ECMA-262)</a> Object.isFrozen	<div>LS</div> Living Standard	

## 浏览器兼容性

	Desktop	Mobile
Feature	Firefox (Gecko)	Chrome Internet Explorer Opera Safari
Basic support	4 (2.0)	69未实现5.1

## 相关链接

- Object.freeze
- Object.isExtensible

- `Object.seal`
- `Object.isSealed`