

ES5

来源网址: <http://blog.csdn.net/lihongxun945/article/details/46867397>

ES5 新增内容

ES6 最近炒得火热, 准备写一下ES6规范的主要内容。在这之前先复习一下ES5相关的内容。

ES5 总体来说改动很少, 最重要的是如下几个改动:

use strict

可以在函数定义的第一行通过一个字符串语法把这个函数声明为 **严格模式**

```
function() {  
    "use strict";  
}
```

在严格模式下, 会有如下一些限制:

- 未声明的变量赋值直接报错, 而不是变成全局变量
- 函数默认的 this 将变成 undefined 而不是默认指向 window
- 对只读属性进行写操作, 删除不可删除属性, 对不可拓展的对象添加属性均会抛出异常
- 禁止使用 with 语句
- 重名的属性会报错, 重名的函数参数会报错

JSON 支持

默认支持 JSON.parse 和 JSON.stringify

对象和属性

现在可以对一个对象的属性设置 getter setter:

```
var obj = {};  
Object.defineProperty(obj, "name", {  
    get: function() {  
        return name.toUpperCase();  
    },  
    set: function(n) {
```

```
    name=n;
  }
})
```

可以定义一个对象的属性是否可以修改、是否可以删除、是否可以添加:

```
Object.defineProperty( obj, "value", {
  value: true,
  writable: false,
  enumerable: true,
  configurable: true
});
```

现在可以直接**冻结**一个对象，冻结分两种：

- `Object.seal(obj)`, 使对象的属性配置无法修改，等价于给每一个属性都设置成 `configurable=false`
- `Object.freeze(obj)`, 除了 `seal` 之外，它还会是属性值本身无法被修改，相当于对每个属性同时进行了 `configurable=false` 和 `writable=false`

现在可以更方便的遍历对象的属性了：

- `Object.getOwnPropertyNames`
- `Object.keys`

数组新增方法

数组新增了如下常用的方法：

- `map`
- `every`
- `filter`
- `forEach`
- `indexOf`
- `reduce`
- `reduceRight`
- `some`

所以你可以放心的写如下代码：

```
[1, 2, 3].map(function(a) {return a*a;})
```

- Firefox 3.6+
- Safari 5+

- Chrome 8+
- Internet Explorer 9+