**Google JavaScript代码风格指南**

修正版本 2.28

原文:<http://google-styleguide.googlecode.com/svn/trunk/javascriptguide.xml>

*Aaron Whyte*

*Bob Jervis*

*Dan Pupius*

*Eric Arvidsson*

*Fritz Schneider*

*Robby Walker*

每个风格点都有一个展开/收起按钮以便你可以得到更多的信息:▶. 你可以将全部展开或收起:

▶ 全部展开/全部收起

**内容树**

|  |  |
| --- | --- |
| JavaScript语言规范 | 变量 常量 分号 嵌套函数 块内函数声明 异常处理 自定义异常 标准特性 基本包装类型 多级原型结构 方法定义 闭包 eval() with() {} this for-in 循环 关联数组 多行字符串字面量 Array和Object字面量 修改内置对象的原型 IE下的条件注释 |
| JavaScript代码风格规范 | 命名 自定义toString()方法 延迟初始化 明确作用域 代码格式化 括号 字符串 可见性（私有和受保护的字段） JavaScript 类型 注释 内部类和枚举 编译压缩 提示和技巧 |

**背景**

JavaScript是一门客户端脚本语言，Google经常用它来晒优越，本文档列出了一些在做JS项目时需要注意的地方。都是高富帅整理的，还望各位屌丝们有时间都好好看看多学学，别对这个世界太消极了！（译者注：所谓的客户端呢就是浏览器或本地软件环境，市面上所有的浏览器都支持JS，JS发展到现在统一WEB端再进军移动互联网后，相信在一段时间内都是不可被轻易替代的语言，所以有兴趣的同学们可以多关注一下）

**JavaScript语言规范**

**变量（var）**

▶

每个变量声明都要加上var关键字噢。

**常量**

▶

常量命名用类似NAMES\_LIKE\_THIS这样的形式。没事干了可以用@const来标记它是常量，但永远不要用const关键字来进行常量声明。

**分号**

▶

每一语句的结尾都要加上分号噢。

**嵌套函数**

▶

可以使用，木有问题。

**块内函数声明**

▶

呀咩嗲。

**异常处理**

▶

支持合理使用。

**自定义异常**

▶

支持合理使用。

**标准特性**

▶

总是优越于非标准特性之上。

**基本包装类型**

▶

主动 new 基本包装类型（Boolean/Number/String）的人儿永远也成不了高富帅！

**多级原型结构**

▶

不是怎么推荐使用。

**方法定义**

▶

Foo.prototype.bar = function() { ... };

**闭包**

▶

上古之大杀器，顺则飞龙在天，滥则亢龙有悔。

**eval()**

▶

只用于反序列化。（反序列化的意思是从字节流中重构对象，这里指的应该是JSON字符串重构成对象，或是执行服务器返回的JS语句）

**with() {}**

▶

别用。

**this**

▶

仅在构造函数，方法，闭包中去使用它。

**for-in 循环**

▶

只在 object/map/hash 要遍历键值的时候使用。

**关联数组**

▶

永远不要用 Array 去做 map/hash/associative 要做的事情。

**多行字符串字面量**

▶

不要！

**Array和Object字面量**

▶

推荐使用。

**修改内置对象的原型**

▶

别这样做。

**IE下的条件注释**

▶

别用。

**JavaScript代码风格规范**

**命名**

▶

通常，使用类似于 functionNamesLikeThis, variableNamesLikeThis, ClassNamesLikeThis, EnumNamesLikeThis, methodNamesLikeThis, 和 SYMBOLIC\_CONSTANTS\_LIKE\_THIS 这样的命名方式（驼峰式）

**自定义 toString() 方法**

▶

应该总是成功调用，无副作用。

**延迟初始化**

▶

可以。

**明确作用域**

▶

始终需要明确。

**代码格式化**

▶

一两句话说不明白，展开了详细讨论

**括号**

▶

只在需要的时候使用。

**字符串**

▶

单引号（'）比双引号（"）更好。

**可见性（私有和受保护的字段）**

▶

鼓励使用JSDoc中的 @private 和 @protected 进行标注。

**JavaScript类型**

▶

强烈建议使用编译器。

**注释**

▶

使用 JSDoc

**内部类和枚举**

▶

内部类和枚举的定义应和顶级类放在同一文件内。

只需要声明顶级类。

**编译压缩**

▶

推荐使用。

**提示和技巧**

▶

JavaScript 习习更健康。

**吐槽部分**

*坚持一致原则*。

如果你要编辑代码，先花几分钟看看它的代码风格，如果它这么做，那你也应该这么做。keleyi.com

风格统一了，就有了一个共同思维的环境，参与者就可以专注的看你要说什么，而不是先想你是在说哪星球的语言。 虽然我们在这里提出统一样式规则，但就只是想让大家都知晓并借鉴而对自己的风格进行修正。 当然，保持自己独有的风格也是很重要的。balabala……