

变量命名规范

1. 命名规范:

- 由字母(A-Za-z)、数字(0-9)、下划线(_)、美元符号(\$)组成, 如: usrAge, num01, _name
- 严格区分大小写。var app; 和 var App; 是两个变量
- 不能以数字开头。18age 是错误的
- 不能是关键字、保留字。例如: var、for、while

2. 其他一些可以遵循的规则:

- 变量名得有意义。MMD BBD a1 a2 aaa
- 遵守驼峰命名法。首字母小写, 后面单词的首字母需要大写。myFirstName
- 使用易读的命名, 比如 userName 或者 shoppingCart。
- 离诸如 a、b、c 这种缩写和短名称远一点, 除非你真的知道你在干什么。
- 变量名在能够准确描述变量的同时要足够简洁。不好的例子就是 data 和 value, 这样的名称等于什么都没说。如果能够非常明显地从上下文知道数据和值所表达的含义, 这样使用它们也是可以的。
- 脑海中的术语要和团队保持一致。如果网站的访客称为“用户”, 则我们采用相关的变量命名, 比如 currentUser 或者 newUser, 而不要使用 currentVisitor 或者一个 newManInTown。

3. 需要注意大写字母是可以使用的, 并且**大小写敏感**。也就是说A和a是两个变量。

命名的推荐及不推荐举例

推荐: 变量使用小驼峰命令, 前缀为修饰形容词, 后面为名词

```
let name="zhangsan"
let minAge=12
let lastName="gis"
```

注意: 变量中不允许出现拼音; 拼音在变量命名中总显得累赘; 不能达到见字知意的效果; 如下集中拼音都是不规范的格式

```
// 不规范
let babaName="xiaobai"
let fatherXingMing="xiaobai"
let chinaMianJi=960
let zhongguoArea=960
// 规范
let fatherName="xiaobai"
let chinaArea=960
```

常量通常使用大写; 用下划线连接两个单词

```
const DOMAIN = 'xiaobaigis.com';
const MAX_COUNT = 10;
```

函数通常使用小驼峰式命名法, 前缀为动词, 后面为主体名词

```
// 得到姓名
function getName(){}
// 设置性别
function setSex(){}
// 是否男性
function isMale(){}
```

常用的前缀动词规范推荐如下:

```
/*
can    判断是否可执行某个动作
has    判断是否含有某个值
is     判断是否为某个值
get    获取某个值
set    设置某个值
load   加载某些数据
*/
```

类和构造函数使用大写驼峰命名(首字母大写)

```
function Person(){
}
class Animal{
  constructor(option) {
  }}

```

标识符、关键字、保留字

1. 标识符

标识(zhi)符: 就是指开发人员为变量、属性、函数、参数取的名字。
标识符不能是关键字或保留字。

2. 关键字

关键字: 是指 JS 本身已经使用了的字, 不能再用它充当变量名、方法名。
包括: break、case、catch、continue、default、delete、do、else、finally、for、function、if、in、instanceof、new、return、switch、this、throw、try、typeof、var、void、while、with、abstract arguments、boolean、break、bytecase、catch、char、class*、constcontinue、debugger、default、delete、dodouble、else、enum、eval、export*extends* false、final、finally、floatfor、function goto、if、implementsimport*、in、instanceof、int、interfacelet、long、native、new、nullpackage、private、protected public、returnshort、static、super*、switch、synchronizedthis、throw、throws、transient true try、typeof、var、void、volatilewhile、with、yield等等

3. 保留字

保留字：实际上就是预留的“关键字”，意思是现在虽然还不是关键字，但是未来可能会成为关键字，同样不能使用它们当变量名或方法名。

包括：boolean、byte、char、class、const、debugger、double、enum、export、extends、final、float、goto、implements、import、int、interface、long、native、package、private、protected、public、short、static、super、synchronized、throws、transient、volatile 等。

注意：如果将保留字用作变量名或函数名，那么除非将来的浏览器实现了该保留字，否则很可能收不到任何错误消息。当浏览器将其实现后，该单词将被看做关键字，如此将出现关键字错误。

下列都是非常正确的变量命名：

```
var haha = 250;
```

```
var xixi = 300;
```

```
var a1 = 400;
```

```
var a2 = 400;
```

```
var abc_123 = 400;
```

```
var $abc = 999;
```

```
var $o0_0o$ = 888;
```

```
var $ = 1000;
```

```
var _ = 2000;
```

```
var ____ = 3000;
```

下列都是错误的命名：

```
var a-1 = 1000; //不能有怪异符号
```

```
var a@ = 2000; //不能有怪异符号
```

```
var 2year = 3000; //不能以数字开头
```

```
var a¥ = 4000; //不能有怪异符号
```

```
var a*#$$@$ = 5000; //不能有怪异符号
```

```
var a b = 300; //不能有空格
```