

| 一般做法                    | 短码做法                    |
|-------------------------|-------------------------|
| 全局变量用容易理解的名称，局部变量用短名称命名 | 变量全部用一个字母命名             |
| 函数用动词命名                 | 函数用一个字母命名               |
| 用明确的描述文字命名              | 用一个字母命名                 |
| 为结构加上合理的缩进              | 该缩进的地方也不要放进多余的空格        |
| 写成自然的形式                 | 写成最短的形式                 |
| 加上括号避免误读                | 熟知运算符优先级，尽量不用括号         |
| 把复杂的表达式分开               | 把复杂的表达式换成同等的简单表达式或近似表达式 |
| 明确编写源代码                 | 明确编写源代码                 |
| 注意副作用                   | 把副作用应用到极致               |
| 统一缩进、大括号的使用方式           | 绝不缩进，大括号能省就省            |
| 以惯用写法确保一致性              | 学会最短的惯用法，尽快转换成短码编程想法    |
| 多重分支使用 else-if          | 不管是不是多重分支，都使用条件运算       |
| 尽量不要使用宏功能               | 绝对不要用宏功能                |
| 宏自身的参数记得加上括号            | 就算非得使用宏，还是千万别用括号        |
| 为魔朮数字命名                 | 魔朮数字基本上维持原状             |
| 不要使用整数，使用字符串常量          | 整数、浮点数、字符串常量挑最短的        |
| 用语言计算对象大小               | 事先计算好对象大小               |
| 理所当然的事情就不要再写注释了         | 完全不用写注释                 |
| 为函数与全局变量加上详细说明          | 完全不用写注释                 |
| 烂原始代码就别注释了，重写比较快        | 长的源代码就别注释了，缩短比较快        |
| 注释不要与源代码冲突              | 完全不用写注释                 |
| 明确注释，不要导致混乱             | 什么都不写是最明智的              |