这里不是 Ruby语言的编码约定，只是建议，很少的一些建议。在语法正确的前提下，你可以按照自己的编码风格自由组织你的代码。

* 命名

常量全用大写的字母，用下划线分割单词。例如：MAX， ARRAY\_LENGTH。

类名和模块名用大写字母开头的单词组合而成。例如：MyClass, Person。

方法名全用小写的字母，用下划线分割单词。例如：talk, is\_prime?。

Notice:

在Ruby里,有时将“!”和“?”附于某些方法名后面。

惊叹号“!”暗示这个方法具有破坏性, 有可能会改变传入的参数。

问号“?”表示这个方法是一个布尔方法，只会返回 true 或 false。

变量和参数用小写字母开头的单词组合而成。例如：name, currentValue。

类名、模块名、变量名、参数名最好使用“名词”或者“形容词＋名词”。方法名最好使用“动词”或者“动词＋名词”。例如：aStudent.talk 。

空格规范

* 空格和圆括号

关键字之后要留空格。

逗号“，”、 分号“；”之后要留空格。 “，”、 “；”向前紧跟，紧跟处不留空格。

赋值操作符、比较操作符、算术操作符、逻辑操作符，如“=”、 “+=” “>=”、 “<=”、

“+”、“\*”、“%”、“&&”、“||”等二元操作符的前后应当加空格。

一元操作符如“!”、“~”等之后不加空格。

象“［］”、“.”、“：：”这类操作符前后不加空格。

函数名之后不要留空格，紧跟左圆括号“（”，以与关键字区别。左圆括号“（”

向后紧跟，右圆括号“）”向前紧跟，紧跟处不留空格。

def talk name

“Hi” + name

end

puts talk "kaichuan" #Hi! Kaichuan

优先规则会自动确定哪个参数被哪个方法使用。但是，生活并不总是美好的，

事情经常变得复杂

a = 5

b = 3

puts a > b ? “a > b” : “a <= b”

puts (a > b)? ("a > b") : (" a <= b")

puts a>b? "a > b" : "a <= b" #错误语句

最后一条语句，变量b 与三元条件运算符的问号“？”之间没有空格，没有圆括号，产生错误。所以建议除了极简单的情况，还是使用圆括号为好。

* 使用 return

你在定义方法的时候，在最后一行可以显式地 return 某个值或几个值，但却不是必须的。 Ruby 方法的最后一行语句如果是表达式，表达式的值会被自动返回；最后一行语句如果不是表达式，就什么也不返回。

return 并不仅仅用在方法的最后一行。使用 break 你能够跳出本层循环，如果要从多重循环体中跳出，可以使用return ，结束这个方法； return还能够从方法的某个执行点立即退出，而不理会方法的其余代码。

* 注释

养成写注释的习惯吧！你见过没有路标的高速公路吗？

注释表明了一段代码块的功能、意图或是代码块的解释，应该简洁明了，错误的注释不如没有注释。一般地，注释的位置应与被描述的代码相邻，可以放在代码的上方或右方，不要放在代码的下方。