

高效率编辑器 VIM—操作篇，非常适合 VIM 新手

虽然从很久前就开始用 VIM 了，但一直都是半吊子，翻来覆去只用自己会的命令。最近为了提高书写代码的效率，还有 coding 时候的乐趣，又重新钻研了一下 VIM，发现了一篇很好的 VIM 入门的文章，原文是英文版的，我觉得非常适合 VIM 使用入门，所以翻译了过来。这里是简单的介绍了 VIM 的操作方式，并没有说为什么要用 VIM，如果你想知道答案可以去 Google，VIM 被誉为编辑器之神。

这篇教程写了在不同工作模式下使用 VIM 的一些基本技巧——即插入模式 (insert mode)，命令模式 (command mode)，存取文件等。目的是帮助刚刚接触 VIM 的新手更加有效率的使用这个出色的编辑器。

说明：在这篇文章里面，<C-X> 代表 Ctrl + X——就是按住 Ctrl 键然后再按 X。而且你可以在很多情况下使用 :help command 来获得大部分命令的帮助，这个是 VIM 的内部帮助文件命令。

高效率移动

在插入模式之外

基本上来说，你应该尽可能少的呆在插入模式里面，因为在插入模式里面 VIM 就像一个“哑巴”编辑器一样。很多新手都会一直呆在插入模式里面，因为这样易于使用。但 VIM 的强大之处在于他的命令行模式！你会发现，在你越来越了解 VIM 之后，你就会花越来越少的时间使用插入模式了。

使用 h、j、k、l

使用 VIM 高效率编辑的第一步，就是放弃使用箭头键。使用 VIM，你就不用频繁的在箭头键和字母键之间移来移去了，这会节省你很多时间。当你在命令模式时，你可以用 h、j、k、l 来分别实现左、下、上、右箭头的功能。一开始可能需要适应一下，但一旦习惯这种方式，你就会发现这样操作的高效之处了。

在你编辑你的电子邮件或者其他有段落的文本时，你可能会发现使用方向键和你预期的效果不一样，有时候可能会一次跳过了很多行。这是因为你的段落在 VIM 看来是一个大的长长的行。这时你可以在按 h、j、k 或者 l 之前键入一个 g，这样 VIM 就会按屏幕上面的行如你所愿的移动了。

在当前行里面有效的移动光标

很多编辑器只提供了简单的命令来控制光标的移动（比如左、上、右、下、到行首/尾等）。VIM 则提供了很多强大的命令来满足你控制光标的欲望。当光标从一点移动到另外一点，在这两点之间的文本（包括这两个点）称作被“跨过”，这里的命令也被称作是 motion。（简单说明一下，后面会用到这个重要的概念）

这里是常用到的一些命令 (motion)：

- fx: 移动光标到当前行的下一个 x 处。很明显，x 可以是任意一个字母，而且你可以使用 ; 来重复你的上一个 f 命令。
- tx: 和上面的命令类似，但是是移动到 x 的左边一个位置。（这真的很有用）
- Fx: 和 fx 类似，不过是往回找。

- w: 光标往前移动一个词。
- b: 光标往后移动一个词。
- 0: 移动光标到当前行首。
- ^: 移动光标到当前行的第一个字母位置。
- \$: 移动光标到行尾。
-): 移动光标到下一个句子。
- (: 移动光标到上一个句子。

在整个文件里面有效移动光标

VIM 有很多命令，可以用来到达文件里面你想到达的地方。下面是一些在文件里面移动的命令：

- <C-F>: 向下移动一屏。
- <C-B>: 向上移动一屏。
- G: 到文件尾
- numG: 移动光标到指定的行 (num)。(比如 10G 就是到第 10 行)
- gg: 到文件首
- H: 移动光标到屏幕上面
- M: 移动光标到屏幕中间
- L: 移动光标到屏幕下面
- *: 读取光标处的字符串，并且移动光标到它再次出现的地方。
- #: 和上面的类似，但是是往反方向寻找。
- /text: 从当前光标处开始搜索字符串 text，并且到达 text 出现的地方。必须使用回车来开始这个搜索命令。如果想重复上次的搜索的话，按 n。
- ? text: 和上面类似，但是是反方向。
- ma: 在当前光标的位置标记一个书签，名字为 a。书签名只能是小写字母。你看不见书签的存在，但它确实已经在那里了。
- `a: 到书签 a 处。注意这个不是单引号，它一般位于大部分键盘的 1 的左边。
- `.: 到你上次编辑文件的地方。这个命令很有用，而且你不用自己去标记它。

高效的输入

使用关键词自动完成

VIM 有一个非常漂亮的关键词自动完成系统。这表示，你可以输入一个长词的一部分，然后按一下某个键，然后 VIM 就替你完成了这个长词的输入了。举个例子：你有一个变量名为 iAmALongAndAwkwardVarName 在你写的代码的某个地方。也许你不想每回都自己一个一个字母的去输入它。

使用关键词自动完成功能，你只需要输入开始几个字母（比如 iAmAL），然后按 <C-N>（按住 Ctrl，再按 N）或者 <C-P>。如果 VIM 没有给出你想要的词，继续按，直到你满意为止，VIM 会一直循环它找到的匹配的字符串。

聪明的进入插入模式

很多新手进入插入模式都只是用 i。这样当然可以进入插入模式，但通常不是那么合适，因为 VIM 提供了很多进入插入模式的命令。下面是最常用的一些：

- i: 在当前字符的左边插入

- I: 在当前行首插入
- a: 在当前字符的右边插入
- A: 在当前行尾插入
- o: 在当前行下面插入一个新行
- O: 在当前行上面插入一个新行
- c{motion}: 删除 motion 命令跨过的字符, 并且进入插入模式。比如: c\$, 这将会删除从光标位置到行尾的字符并且进入插入模式。ct!, 这会删除从光标位置到下一个叹号 (但不包括), 然后进入插入模式。被删除的字符被存在了剪贴板里面, 并且可以再粘贴出来。
- d{motion}: 和上面差不多, 但是不进入插入模式。