
vim 编辑器

一、 vim 编辑器简介

在 Linux 中最常见的代码编辑器包括 **vim**、**emacs**、**gedit** 等。注意：代码编辑器的**实质就是一个文本编辑器**，只不过**增加了一些代码编辑的辅助功能**，例如关键字高亮，补齐等。

vim 是“Vimsual Interface”的简称，它在 Linux 上的地位就像 Edit 程序在 DOS 上一样。它可以执行**输入、输出、删除、查找、替换、块操作**等众多文本操作，而且用户可以根据自己的需要对其进行定制，这是其他编辑器所没有的。

1. 启动与退出 vim

启动：vim filename

退出：必须先按 Esc 键回到命令行模式，然后键入“:”，此时光标会停留在最下面一行（底行模式），再键入“q”，最后按下 Enter 键即可退出。

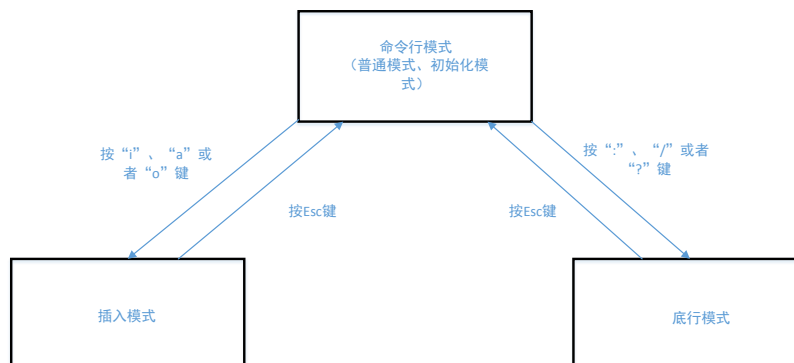
2. vim 拥有三种工作模式

命令行模式(Command mode)、插入模式(Insert mode)与底行模式(Last line mode)。

三种模式下的功能可描述如下：

- 命令行模式：也叫做“普通模式”，它是启动 vim 编辑器后的初始模式。在该模式下，主要是使用隐式命令（命令不显示）来实现光标的移动、复制、粘贴、删除等操作。但是在该模式下，编辑器并不接受用户从键盘输入的任何字符作为文档的编辑内容。
- 插入模式：在该模式下，用户输入的任何字符都被认为是编辑到某一文件的内容，并且直接显示在 vim 的文本编辑区。
- 底行模式：在该模式下，用户输入的任何字符都会在 vim 的最下面一行显示，按 Enter 键后便会执行该命令。

三种模式之间的切换如下图所示：



二、 vim 的命令行模式

命令行模式是进入 vim 后的初始模式，在该模式下主要是通过方向键来移动光标的位置，并通过相应的命令来进行文字的编辑。在插入模式下按 Esc 键，或是在底行模式下按 Esc 键，或是在底行模式下执行了错误的命令，vim 都会自动回到命令模式。

1. 移动光标

在命令行模式下，一般通过使用上下左右四个方向键来移动光标的位置。但是在有些情况下，例如使用 telnet 远程登录时，方向键就不能使用了，必须用命令行下的光标移动命令。这些命令如表 1 所示：

表 1 移动光标的常用命令

命令	操作说明
h	向左移动光标
l	向右移动光标
j	向下移动光标
k	向上移动光标
^	将光标移动到该行的开头（指第一个非空字符上）
\$	将光标移动到该行的行尾，与键盘上的 End 键功能相同
0	将光标移动到该行的行首，与键盘上的 Home 键功能相同
G	将光标移动到文档最后一行的开头（第一个非空字符）
nG	将光标移动到文档的第 n 行的开头（第一个非空字符），n 为整数
w	将光标向后移动一个字（单词）
nw	将光标向后移动 n 个字（单词），n 为整数
b	将光标向前移动一个字（单词）
nb	将光标向前移动 n 个字（单词），n 为整数
e	将光标移动到本单词的最后一个字符。如果光标所在的位置为本单词的最后一个字符，则跳到下一个单词的最后一个字符。“.”、“,”、“#”、“\”等特殊字符都会被当做一个字
{	将光标移动到前面的“{”处。这在 C 语言编程中很实用
}	将光标移动到后面的“}”处
Ctrl+b	向上翻一页，相当于 Page Up
Ctrl+f	向下翻一页，相当于 Page Down
Ctrl+u	向上移动半页
Ctrl+d	向下移动半页
Ctrl+e	向下翻一页
Ctrl+y	向上翻一行

2. 复制和粘贴

复制和粘贴是在编辑文本时最常用的操作之一，可以大大地节约用户重复输入的时间。常用的赋值粘贴命令如表 2 所示：

表 2 复制粘贴的常用命令

命令	操作说明
----	------

yy	复制光标所在行的整行内容
yw	复制光标所在的单词的内容
nyy	复制光标所在行开始向下的 n 行内容，n 为正整数，表示复制的行数
nyw	复制光标所在字开始向后的 n 个字，n 为正整数，表示复制的字数
p	粘贴，将复制的内容粘贴在光标所在的位置

3. 删除

vim 编辑器中的删除操作可以是一次删除一个字符，也可以是一次删除多个字符，或者整行字符，vim 命令行模式常用的删除命令如表 3 所示：

表 3 删除文本的常用命令

命令	操作说明
x	删除光标所在位置的字符，同键盘上的 Delete 键
X	删除光标所在位置的前一个字符
nx	删除光标所在位置及其之后的 n-1 个字符，n 为整数
nX	删除光标所在位置及其之前的 n-1 个字符，n 为整数
dw	删除光标所在位置的单词
ndw	删除光标所在位置及其之后的 n-1 个单词，n 为整数
d0	删除当前光标所在位置之前的所有字符
d\$	删除当前光标所在位置之后的所有字符
dd	删除光标所在行
ndd	删除光标所在行及其向下 n-1 行，n 为整数
nd+上箭头	删除光标所在行及其向上 n 行，n 为整数
nd+下箭头	删除光标所在行及其向下 n 行，n 为整数

4. 其他命令

命令行模式下其他常用的命令包括字符替换、撤销操作、符号匹配等命令，这些也是在使用 vim 时经常遇到的命令，其操作如表 4 所示：

表 4 其他常用操作命令

命令	操作说明
r	替换光标所在位置的字符，例如 rx 是指将光标所在位置的字符替换为 x
R	替换光标所到之处的字符，直到按下 Esc 键为止
u	表示恢复功能，即撤销上一次操作
U	取消对当前行所做的所有改变
.	重复上一次命令
ZZ	保存文档后退出 vim 编辑器
%	符号匹配功能，在编辑时若输入 “%(”，系统将会自动匹配相应的 “)”

三、 vim 的插入模式

插入模式是 vim 编辑器最简单的模式，因为在此模式下没有那些繁琐的命令。用户从键盘输入的任何有效字符都被看作是写进当前正在编辑的文本中的内容，并显示在 vim 的文本编辑区。

也就是说，只要在插入模式下，才可以进行文字的输入操作。表 5 所示为从命令行模式切换至插入模式的几个常用命令。当在插入模式下时，可以按 Esc 键回到命令行模式。

表 5 命令模式切换至插入模式的命令

命令	操作说明
i	从光标所在的位置开始插入新的字符
I	从光标所在行的行首开始插入新的字符
a	从光标所在的位置的下一个字符开始插入新的输入字符
A	从光标所在行的行尾开始插入新的字符
o	新增加一行，并将光标移动到下一行的开头并在此插入字符
O	在当前行的上面新增加一行，并将新光标移动到上一行的开头并在此插入字符

四、 vim 的底行模式

vim 底行模式也叫做“最后行模式”，是指可以在界面最底部的一行输入控制操作命令，主要用来进行一些文字编辑的辅助功能，例如字符串搜寻、替换、保存文件以及推出 vim 等。不同于命令行模式，底行模式下输入的命令都会在最底部的一行中显示，按 Enter 键 vim 便会执行底行的命令了。

在命令行模式下输入冒号“:”，或者是使用“?”和“/”键，就可以进入底行模式了。比起命令行模式的诸多操作命令，底行模式的操作命令就少很多了，如表 5 所示：

表 5 底行模式下的常用命令

命令	操作说明
q	退出 vim 程序，如果对文件进行过修改，则必须先保存文件
q!	强制退出 vim 而不保存文件
x	(exit) 保存文件并退出 vim
x!	强制保存文件并退出
w	(write) 保存文件，但不退出 vim
w!	对于只读文件，强制保存修改的内容，但不退出 vim
wq	保存文件并退出 vim，同 x
E	在 vim 中创建新的文件，并可为文件命名
N	在本 vim 窗口中打开新的文件
w filename	另存文件，filename 参数指定新文件名，不退出 vim
w! filename	强制另存文件，filename 参数指定新文件名，不退出 vim
r filename	(read) 读入 filename 指定的文件的内容，将其插入到光标的位置

set nu	在 vim 的每行开头处显示行号
s/pattern1/pattern2/g	将光标当前的字符串 pattern1 替换为 pattern2
%s/pattern1/pattern2/g	将所有行的字符串 pattern1 替换为 pattern2
g/pattern1/s//pattern2	将所有行的字符串 pattern1 替换为 pattern2
num1,num2s/pattern1/pattern2/g	将行 num1 到 num2 的字符串 pattern1 替换为 pattern2
/	查找匹配字符串功能。用“/字符串”命令模式，系统便会自动查找，并突出显示所有找到的字符串，然后转到找到的第一个字符串。如果想继续向后寻找，可以按 n 键；向前寻找按 N 键
?	也可以使用“? 字符串”查找特定字符串，它的使用与“/字符串”相似，但它是向前查找字符串

五、 vim 的可视模式

vim 还有一种模式就是可视模式。可视模式相当于高亮选取文本后的普通模式。可视模式具有子模式，以行为单位进行选取的**可视行模式**，使用“V”键进入（也就是 **Shift+v**）；和以块为单位进行选取的**可视块模式**，使用“**Ctrl+v**”键进入。退出的话按两次 Esc 键。

1. 进入可视模式

命令	操作说明
v	进入面向字符的可视模式
V	进入面向行的可视模式
进入面向行的可视模式	进入面向列块的可视模式
Ctrl + v	重选上次的高亮选区

2. 在可视模式之间切换

命令	操作说明
Esc	回到普通模式
Ctrl + [回到普通模式
v / V / Ctrl + v	在对应的模式下会切换到普通模式，即普通模式下执行两次一样的命令还是普通模式
v	在不对应的模式下会切换进入面向字符的可视模式
V	在不对应的模式下会切换进入面向行的可视模式
Ctrl + v	在不对应的模式下会切换进入面向列块的可视模式

3. 切换高亮选区的活动端

高亮选区的范围由其两个端点界定。其中一端固定，而另一端可以随光标自由移动。可以用字母 o 来切换活动端点，从而调整选区的开始的位置。

命令	操作说明
o	切换高亮选区的活动端

4. 重复执行面向行的可视命令

命令	操作说明
.	在行可视模式执行操作后，执行此命令会重选相同范围的文本行并执行一样的操作
V j > .	进入面向行的可视模式，再用 j 调整了行边界，执行命令缩进命令> 在普通模式下执行.会将重复刚选中的可视行并执行缩进。 建议，只要可能，最好用操作符命令，而不是可视命令

5. 从可视模式到插入模式

在可视模式中，i 和 a 键会被当做对象的组成部分，按 i 或 a 不会进入到插入模式。得用 I 和 A 键进入插入模式。