

## vi/vim编辑器使用方法详解

vi编辑器是所有Unix及Linux系统下标准的编辑器，他就相当于windows系统中的记事本一样，它的强大不逊色于任何最新的文本编辑器。他是我们使用Linux系统不能缺少的工具。由于对Unix及linux系统的任何版本，vi编辑器是完全相同的，学会它后，您将在Linux的世界里畅行无阻。

vim 具有程序编辑的能力，可以以字体颜色辨别语法的正确性，方便程序设计；

vim可以当作vi的升级版，他可以用多种颜色的方式来显示一些特殊的信息。

vim会依据文件扩展名或者是文件内的开头信息，判断该文件的内容而自动的执行该程序的语法判断式，再以颜色来显示程序代码与一般信息。

vim里面加入了很多额外的功能，例如支持正则表达式的搜索、多文件编辑、块复制等等。这对于我们在Linux上进行一些配置文件的修改工作时是很棒的功能。

### 1.vi/vim是什么？

- ①vi是Visual Interface的缩写，即 可视化接口
- ②vim是vi iMprove的缩写，即 vi的增强版（具有语法着色功能）

### 2.vim模式有哪些？如何切换？



#### ①三种模式

命令模式（默认）：刚进入vim的时候，默认就是 命令模式，可以复制行，删除行等。

输入模式：可以输入内容。

末行模式：在最下边，除编辑模式，可以输入诸多管理员命令

#### ②模式转换

命令模式--输入模式：

- i：在当前光标所在字符的前面，转为输入模式
- I：在当前光标所在行的行首转换为输入模式
- a：在当前光标所在字符的后面，转为输入模式
- A：在光标所在行的行尾，转换为输入模式
- o：在当前光标所在行的下方，新建一行，并转为输入模式
- O：在当前光标所在行的上方，新建一行，并转为输入模式
- s：删除光标所在字符
- r：替换光标处字符

输入模式--命令模式

ESC键

命令模式--末行模式

输入：即可 转为末行模式



在末行模式下

- 10d：可删除第10行
- 10,20d：删除10-20行
- set nu：显示行号
- ! 命令：可以执行bash命令

末行模式--命令模式

ESC键



### 3.使用vim如何打开文件？

- ① vim /path/to/somefile : vim后跟文件路径及文件名，如果文件存在，则打开编辑文件窗口，如果文件不存在，则创建文件

```
[root@libaoliang ~]# vim /etc/inittab
```

- ② vim + # : 打开文件，并定位到第#行，#代表数字

```
[root@libaoliang ~]# vim +3 /etc/inittab
```

```
# inittab is only used by upstart for the default runlevel.
#
# ADDING OTHER CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
# System initialization is started by /etc/init/rcS.conf
#
# Individual runlevels are started by /etc/init/rc.conf
#
# Ctrl-Alt-Delete is handled by /etc/init/control-alt-delete.conf
#
# Terminal gettys are handled by /etc/init/tty.conf and /etc/init/serial.conf,
# with configuration in /etc/sysconfig/init.
#
# For information on how to write upstart event handlers, or how
# upstart works, see init(5), init(8), and initctl(8).
#
# Default runlevel. The runlevels used are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault: /etc/inittab文件有26行，884个字符
~
"/etc/inittab" 26L, 884C
```

左下角提示：光标定位在第3行，第1列

3,1 All

- ③ vim + : 打开文件，定位到最后一行

```
[root@libaoliang ~]# vim + /etc/inittab
```

- ④ vim +/PATTERN : 打开文件，定位到第一次被PATTERN匹配到的行的行首

```
[root@libaoliang ~]# vim +/start /etc/inittab
```

```
# inittab is only used by upstart for the default runlevel.
#
# ADDING OTHER CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
#
# System initialization is started by /etc/init/rcS.conf
#
# Individual runlevels are started by /etc/init/rc.conf
#
# Ctrl-Alt-Delete is handled by /etc/init/control-alt-delete.conf
#
# Terminal gettys are handled by /etc/init/tty.conf and /etc/init/serial.conf,
# with configuration in /etc/sysconfig/init.
#
# For information on how to write upstart event handlers, or how
# upstart works, see init(5), init(8), and initctl(8).
#
# Default runlevel. The runlevels used are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
```

### 4.关闭文件

末行模式下：

w: 保存  
q: 退出  
wq 或 x: 保存退出, wq 和 x 都是保存退出  
q!: 强制退出  
w!: 强制保存, 管理员才有权限

命令模式下:

zz: 保存并退出

## 5.移动光标

命令模式下:



- ①逐字符移动
  - h: 向左
  - l: 向右
  - j: 向下
  - k: 向上
  - #h: 移动#个字符
- ②以单词为单位移动
  - w: 移到下一个单词的词首
  - e: 跳至当前或下一个单词的词尾
  - b: 跳至当前或上一个单词的词首
  - #w: 表示移动#个单词
- ③行内跳转
  - 0: 绝对行首
  - ^: 行首第一个非空白字符
  - \$: 绝对行尾
- ④行间跳转
  - #G: 跳转到第#行
  - G: 最后一行



末行模式下:

- ①行间跳转  
直接输入行号 回车即可

## 6.翻页:

命令模式下

Ctrl+f: 向下翻一页  
Ctrl+b: 向上翻一页

Ctrl+d: 向下翻半屏  
Ctrl+u: 向上翻半屏

## 7.删除

命令模式下



- ①删除单个字符:
  - x: 删除光标所在处单个字符
  - #x: 删除光标所在处及向后共#个字符
- ②删除命令: d
  - d命令跟 跳转命令组合使用 如: dw 表示删除光标所在位置到下一个单词词首所有字符
  - #dw, #de, #db : #dw表示 删除当前光标所在处及向后第#个单词词首所有字符
  - dd: 删除当前光标所在行
  - #dd: 删除包含当前光标所在行内的#行



末行模式下:



末行模式下 位置符号

StartADD, EndADDd

.: 表示当前行

\$.: 最后一行

+#: 向下的#行

\$-#: 向上的#行

末行模式下, 删除命令d可以跟以上 位置符号 相结合使用

StartADD, EndADDd: 删除StartADD到EndADDd位置的所有行, 例如: 10, 15d: 删除10到15行所有内容

.d: 删除光标所在行, 同dd

## 8. 粘贴命令: p

p (小写p): 如果删除的或复制的为整行内容, 则粘贴到光标所在行的下方, 如果复制或删除的内容为非整行, 则粘贴至光标所在字符的后面

P (大写P): 如果删除的或复制的为整行内容, 则粘贴到光标所在行的上方, 如果复制或删除的内容为非整行, 则粘贴至光标所在字符的前面

## 9. 复制命令: y



用法同d命令

命令模式

yy: 复制当前行

yw (y#w, ye, y#e, yb, y#b): 复制光标当前位置到下个单词词首内容.#代表数字

y#G: 复制当前位置当#行处的所有内容

末行模式

StartADD, EndADDy: 复制StartADD到EndADDd位置的所有行, 例如: 10, 15d: 复制10到15行所有内容

+3y: 复制当前位置向下3行



## 10. 修改c: 先删除内容, 在转换为输入模式

c: 用法同d

## 11. 替换: r

r: 单字符替换, 会替换光标选中字符

R: 替换模式, 可以做字符替换。

## 12. 撤销编辑操作: u (undo)

u: 撤销前一次编辑操作

#u: 直接撤销#次操作

Ctrl+r: 撤销最近一次撤销操作

## 13. 重复前一次编辑操作.

.: 比如 上一次dd删除了一行, 现在输入. 就可以重复dd删除行的操作

## 14. 可视化模式:

命令模式:

v: 按字符选取, 键盘控制光标 划过的区域被选取

V: 按矩形选取, 键盘控制光标划过的行被选取

## 15. 查找:

末行模式:

/PATTERN: 从当前位置向后查找

? PATTERN: 从当前位置向前查找

n: 向下查找

N: 向上查找

## 16. 查找并替换

在末行模式下使用s命令

```
ADDR1,ADDR2s/PATTERN/String/gi
ADDR1,ADDR2 表示地址；比如1, $表示全文  （%也表示全文）
g：表示全局替换
i：表示忽略大小写
```

## 17.vim打开多个文件



```
vim FILE1 FILE2 FILE3
    末行模式下
        :next 切换到下一个文件
        :prev 切换到上一个文件
        :last 切换到最后一个文件
        :first 切换到第一个文件
    退出多个文件
        qa：全部退出
```



## 18.分屏显示一个文件



```
vim打开文件后
    Ctrl+w,s：水平拆分窗口
    Ctrl+w,v：垂直拆分窗口
在窗口间切换光标
    Ctrl+w, ARROW
    比如 Ctrl+w,→：（使用箭头方向键）光标切换到右边的窗口
    比如 Ctrl+w, h(左),  j(下),  k(上),  l(右) 也可以切换窗口
在末行模式下
    :qa退出所有窗口
```



## 19.分窗口编辑多个文件

```
vim -o：水平分割显示
vim -O：垂直分割显示
例如：vim -o /etc/rc.sysinit /etc/inittab  ：水平分割显示/etc/rc.sysinit和/etc/inittab 这2个文件
```

## 20.将当前文件中部分内容保存为另外一个文件

```
末行模式下
    w：表示保存当前文件
    w /path/to/somewhere：将当前文件另存为到路径指定的地方
    ADDR1,ADDR2w /path/to/somewhere：将ADDR1到ADDR2指定的地址文本 另存为到路径指定的地方
```

## 21.将另一个文件的内容填充在当前文件中(r表示读)

```
末行模式下
    r： /path/to/somefile
例如：vim /etc/rc.sysinit 打开文件后，末行模式 输入 :r /etc/inittab ：会将 inittab文件的所有内容附加到rc.sysinit文件光标所在的位置
```

## 22.跟shell交互（vim 末行模式执行shell）

```
末行模式
! command即可
例如： ! ls /etc/：会列出/etc目录下所有文件及目录 按enter后回到vim编辑状态
```

## 23.高级话题



```
在末行模式下
①显示或取消行号
    set nu (set number)：显示行号
    set nonu：取消显示行号
②显示忽略或区分字符大小写
```

```
set ic (set ignorecase) : 忽略大小写
set noignorecase: 区分大小写
```

### ③设定自动缩进

```
set ai (set autoindent) : 自动缩进
set noai: 取消自动缩进
```

### ④查找到的文本高亮或取消高亮显示

```
set hlsearch: 高亮显示搜索到的文本
set nohlsearch: 取消高亮显示搜索到的文本
```

### ⑤语法高亮

```
syntax on: 打开语法高亮显示
syntax off: 关闭语法高亮显示
```



## 24.vim配置文件

全局vim配置文件: /etc/vimrc

个人vim配置文件: ~/.vimrc

[root@libaoliang ~]# echo "set nu" >> /root/.vimrc # root用户用vim打开文件时, 显示行号。