# **Vi操作**

## **键盘图**

# vi-vim-cheat-sheet-sch

## **常用**

字符按键以及Shift组合，输入字符

ENTER，回车键，换行

BACK SPACE，退格键，删除光标前一个字符

DEL，删除键，删除光标后一个字符

方向键，在文本中移动光标

HOME/END，移动光标到行首/行尾

Page Up/Page Down，上/下翻页

Insert，切换光标为输入/替换模式，光标将变成竖线/下划线

ESC，退出输入模式，切换到命令模式

dd 删除游标所在的那一整行(常用)

yy 复制游标所在的那一行(常用)

## **查找**

/word 向光标之下寻找一个名称为 word 的字符串。例如要在档案内搜寻 vbird 这个字符串，就输入 /vbird 即可！ (常用)

?word 向光标之上寻找一个字符串名称为 word 的字符串。

n 这个 n 是英文按键。代表重复前一个搜寻的动作。举例来说， 如果刚刚我们执行 /vbird 去向下搜寻 vbird 这个字符串，则按下 n 后，会向下继续搜寻下一个名称为 vbird 的字符串。如果是执行 ?vbird 的话，那么按下 n 则会向上继续搜寻名称为 vbird 的字符串！

N 这个 N 是英文按键。与 n 刚好相反，为『反向』进行前一个搜寻动作。 例如 /vbird 后，按下 N 则表示『向上』搜寻 vbird 。

## **替换**

) 文件内全部替换：

:%s#abc#123#g (如文件内有#，可用/替换,:%s/abc/123/g)

--注：把abc替换成123

(或者: %s/str1/str2/g 用str2替换文件中所有的str1）

2) 文件内局部替换：

:20,30s#abc#123(如文件内有#，可用/替换,:%s/abc/123/g)

--注：把20行到30行内abc替换成123

## **删除**

x, X 在一行字当中，x 为向后删除一个字符 (相当于 [del] 按键)， X 为向前删除一个字符(相当于 [backspace] 亦即是退格键) (常用)

nx n 为数字，连续向后删除 n 个字符。举例来说，我要连续删除 10 个字符， 『10x』。

dd 删除游标所在的那一整行(常用)

ndd n 为数字。删除光标所在的向下 n 行，例如 20dd 则是删除 20 行 (常用)

d1G 删除光标所在到第一行的所有数据

dG 删除光标所在到最后一行的所有数据

d$ 删除游标所在处，到该行的最后一个字符

d0 那个是数字的 0 ，删除游标所在处，到该行的最前面一个字符

yy 复制游标所在的那一行(常用)

nyy n 为数字。复制光标所在的向下 n 行，例如 20yy 则是复制 20 行(常用)

y1G 复制游标所在行到第一行的所有数据

yG 复制游标所在行到最后一行的所有数据

y0 复制光标所在的那个字符到该行行首的所有数据

y$ 复制光标所在的那个字符到该行行尾的所有数据

p, P p 为将已复制的数据在光标下一行贴上，P 则为贴在游标上一行！ 举例来说，我目前光标在第 20 行，且已经复制了 10 行数据。则按下 p 后， 那 10 行数据会贴在原本的 20 行之后，亦即由 21 行开始贴。但如果是按下 P 呢？ 那么原本的第 20 行会被推到变成 30 行。 (常用)

J 将光标所在行与下一行的数据结合成同一行

c 重复删除多个数据，例如向下删除 10 行，[ 10cj ]

u 复原前一个动作。(常用)

[Ctrl]+r 重做上一个动作。(常用)

## **插入模式**

i, I 进入插入模式(Insert mode)：

i 为『从目前光标所在处插入』， I 为『在目前所在行的第一个非空格符处开始插入』。 (常用)

a, A 进入插入模式(Insert mode)：

a 为『从目前光标所在的下一个字符处开始插入』， A 为『从光标所在行的最后一个字符处开始插入』。(常用)

o, O 进入插入模式(Insert mode)：

这是英文字母 o 的大小写。o 为『在目前光标所在的下一行处插入新的一行』； O 为在目前光标所在处的上一行插入新的一行！(常用)

r, R

## **保存**

:w 将编辑的数据写入硬盘档案中(常用)

:w! 若文件属性为『只读』时，强制写入该档案。不过，到底能不能写入， 还是跟你对该档案的档案权限有关啊！

:q 离开 vi (常用)

:q! 若曾修改过档案，又不想储存，使用 ! 为强制离开不储存档案。

:w [filename] 将编辑的数据储存成另一个档案（类似另存新档）

:r [filename] 在编辑的数据中，读入另一个档案的数据。亦即将 『filename』 这个档案内容加到游标所在行后面

:n1,n2 w [filename] 将 n1 到 n2 的内容储存成 filename 这个档案。

:! command 暂时离开 vi 到指令行模式下执行 command 的显示结果！例如

『:! ls /home』即可在 vi 当中察看 /home 底下以 ls 输出的档案信息！

## **显示行数**

:set nu 显示行号，设定之后，会在每一行的前缀显示该行的行号

:set nonu 与 set nu 相反，为取消行号！

## **撤销 回复**

**u**  撤销上一步的操作  
**Ctrl+r** 恢复上一步被撤销的操作

https://www.baidu.com/baidu?tn=monline\_3\_dg&ie=utf-8&wd=%E6%9F%A5%E7%9C%8Blinux++%E6%9C%8D%E5%8A%A1