

Vim实战指南(一): 基础编辑命令

Introduction

Vim和Emacs作为两大神级文本编辑器各有利弊，本实战指南不加讨论的介绍Vim.

本实战指南不是教你按照Vim的键位图来编辑文本，而是从实战的角度，教你如何快速的用Vim来开发你的程序，节省你的开发时间。

为什么不用IDE？IDE固然强大，但并不是每一个环境都有现成的IDE可用，尤其是SSH远程计算机，就一个bash解析器可以使用，完全没有图形化界面。IDE的另一个弊端就是需要反复的安装升级更新。

当然IDE也有省事的时候，例如开发Java和C#，IDE自带的debug调试工具，可以非常便利的调试程序。

但是在类Unix系统上开发，我还是推荐学习熟练地使用Vim。何为熟练？完全舍弃你的鼠标，只使用键盘快速的进行开发即为熟练。

介绍Vim(全称是vi improved)之前，自然首先要从vi(全称是visual editor)开始。

对于初学者来说，vi的学习曲线可能比较陡峭。vi分为插入模式(insert mode)以及命令行模式(command mode)。

vi

在Unix/Linux的bash下，输入vi来编辑文本。注意，当你使用vi修改文本时，并不是真正的修改了存放在磁盘上的文件，而是修改了该文件在内存中的拷贝副本。只有当你显示的保存文件时，该内存中的副本才会被写入磁盘，并覆盖文件。

```
$ vi testfile
```

Command Mode

在命令行模式下，vi的常用命令可以按以下方式速记：

编辑类

i: insert, 进入插入模式, 新字符插入在光标前
a: append, 进入编辑模式, 新字符插入在光标后
c: change, 修改
d: delete, 删除
p: put, 放置, 可以将d删除的内容, 放置在光标后面
y: yank, 拷贝
r: replace, 替换, 和c不同在于, 不必进入编辑模式即可替换
s: substitute, 替代, 和c不同在于, 可以只修改一个字符而非整个字
x: x, 和d不同在于, 可以只删除一个字符而非整个字
~: change case, 替换大小写
.: repeat, 重复上一条命令
u: undo, 撤销上一条命令
J: join, 将两行合并为一行

保存退出类

q: quit, 退出, 如果有未保存的修改则无法退出
q!: force quit, 强制退出
w: write edits to disk (save file), 保存文件
w!: force write, 强制保存
ZZ: quit and save edits, 保存文件并退出
e!: revert your changes, 回滚所有修改至原始状态

移动光标类

h: left, 向左移动光标
j: down, 向下移动光标
k: up, 向上移动光标
l: right, 向右移动光标
O: digit zero, move to beginning of line, 移动到行首

\$: move to end of line, 移动到行尾
w: move by word, 按字向后移动光标(包括标点)
W: move by large word, 按字向后移动光标(忽略标点)
b: move backward by word, 按字向前移动光标(包括标点)
B: move backward by large word, 按字向前移动光标(忽略标点)
e: move to end of word, 移动到字尾(包括标点)
E: move to large end of word, 移动到字尾(忽略标点)
G: go to end of the file, 移动到文件末尾最后一行

记住, vi对于命令区分大小写, l和i不是同一个命令。vi的命令不会显示在屏幕上, 每一个命令后面不需要敲击回车(Enter)。

当你不知道你处于哪个模式下时, 连续按3下Esc总能让你回到命令行模式。

组合命令

vi的便捷性在于你可以组合命令, 通过数字+字母, 或者字母+字母, 甚至数字+字母+字母, 可以将单命令构造出你想操作的组合命令

3h: 3 left, 向左移动3次光标, 等同于lll
3W: 3 move by large word, 按3个字向后移动光标(忽略标点等)
1G: go to line 1, 移动到文件首行
3G: go to line 3, 移动到文件第三行
cw: change, move by word, 修改后面的一个字
c3b: change, move backward by 3 words, 修改前面的三个字
c\$: change, move to end of line, 修改光标后面整行文字
c0: change, move to beginning of line, 修改光标之前整行文字
dw: delete word, 删除字
d3w: delete 3 words, 删除三个字
3p: put 3 times, 重复放置3次

甚至你可以重复组合

cc: change line, 修改整行

dd: delete line, 删除整行

3dd: delete 3 line, 删除三行

yy: yank line, 拷贝整行

Summary

上述的基本命令一定要熟记于心, 加强练习。不要觉得命令太多, 仔细研究会发现命令之间有规律可循。只需要记住几个基础命令, 通过场景构造便可以得到组合命令, 这也是vi的便利以及魅力所在。

(本文出自csprojectedu.com, 转载请注明出处)