

命令	解释
vi (vi是vim的软链接) vi 文件名 vi 文件名 + (行数)	编辑文件 若文件不存在,则会新建,需要保存 ---立即去到出错行(默认最后一行)_
--set nu	设置行数可见
末行模式	
-w 保存 :w 文件名 另存为,但不会切换 用于阶段性备份 -wq = -x 保存退出 -q 退出 -q! 强制退出 -e . edit打开内置文件浏览器,不用退出直接去别的文件编辑 -n 文件名 new新建文件	
分屏命令 :sp[文件名] split 横向增加分屏 :vsp[文件名] vertical split 纵向 窗口管理都是需要添加Ctrl + w 的	
命令模式	
重复次数: 先输入数字,再跟上命令,可以让命令重复执行	
移动和选择	
左下上右,hjkl	
行内移动 w -word 向后移动一个单词 b -back 向前移动一个单词 0 零 行首 ^ 行首,非空白 \$ 行尾	
行数移动 gg -go 文件顶部 G -go 文件末尾 数字gg或者数字G 跳至具体行数	
屏幕移动 Ctrl + b back 向上翻页 Ctrl + f forward 向下翻页 H -Head 屏幕顶部 M -Middle 屏幕中间 L -Low 屏幕底部	
移动(程序) {} 上一段 下一段 括号切换 %括号匹配,跳去括号位置	
标记	

m+任意字母 mark +任意一个字母
' +字母 返回标记位置
同名会替换,删除都一起没了

编辑操作--删除,复制,粘贴,替换,编排

可视模式(多选)

v 普通

V 可视行模式

Ctrl + v 可视块 --垂直方向选择文本

删除

x -cut 删除光标所在字符或者所选

d(移动命令) 删除移动命令对应的内容

dd 删除行

D 删除至行尾

常用组合:

复制粘贴

y(移动命令) copy 复制

yy 复制一行

p paste 粘贴

替换

r replace 替换当前字符 命令模式

R relace 进入替换模式 替换模式--多个替换

缩排

>> 向右增加缩进

<< 向左

. 重复上次命令

查找

/str 查找str n: next 查找下一个 N:查找上一个

*星号 向下找当前光标单词 向上找当前光标单词

查找和替换--末行模式

:%s///g -基本格式

:%s/旧文本/新文本/g --全局替换

:s/旧文本/新文本/g --可视区域替换 ---可视行模式

:s/旧文本/新文本/gc --可视区域确认替换 ---可视行模式 confirm

插入命令

O 在当前行前面插入一空行

o 在当前行后面插入一空行

A 在行末添加文本 append

撤销和重复

u -undo 撤销上次命令

ctrl + r 恢复撤销

编辑

1) 切换分屏窗口

分屏窗口都是基于 `CTRL + W` 这个快捷键的, `w` 对应的英文单词是 `window`

命令	英文	功能
<code>w</code>	window	切换到下一个窗口
<code>r</code>	reverse	互换窗口
<code>c</code>	close	关闭当前窗口, 但是不能关闭最后一个窗口
<code>q</code>	quit	退出当前窗口, 如果是最后一个窗口, 则关闭 vi
<code>o</code>	other	关闭其他窗口

1) 上、下、左、右

命令	功能	手指
<code>h</code>	向左	食指
<code>j</code>	向下	食指
<code>k</code>	向上	中指
<code>l</code>	向右	无名指



- 删除命令可以和 **移动命令** 连用, 以下是常见的组合命令:

```
* dw      # 从光标位置删除到单词末尾
* d0      # 从光标位置删除到一行的起始位置
* d}      # 从光标位置删除到段落结尾
* ndd     # 从光标位置向下连续删除 n 行
* d代码行G # 从光标所在行 删除到 指定代码行 之间的所有代码
* d'a     # 从光标所在行 删除到 标记a 之间的所有代码
```

1. `y` - `yes` 替换
2. `n` - `no` 不替换
3. `a` - `all` 替换所有
4. `q` - `quit` 退出替换

vim三种工作模式

03. 三种工作模式

- `vi` 有三种基本工作模式:

1. 命令模式

- 打开文件首先进入命令模式, 是使用 `vi` 的入口
- 通过 命令 对文件进行常规的编辑操作, 例如: 定位、翻页、复制、粘贴、删除.....
- 在其他图形编辑器下, 通过 快捷键 或者 鼠标 实现的操作, 都在 命令模式 下实现

2. 末行模式 -- 执行 保存、退出 等操作

- 要退出 `vi` 返回到控制台, 需要在末行模式下输入命令
- 末行模式 是 `vi` 的出口

3. 编辑模式 -- 正常的编辑文字



异常处理:---突然退出文件,没保存

1,重新打开文件 2,会出现以下界面:

2.2 异常处理

- 如果 `vi` 异常退出, 在磁盘上可能会保存有 交换文件
- 下次再使用 `vi` 编辑该文件时, 会看到以下屏幕信息, 按下字母 `d` 可以 删除交换文件 即可

提示: 按下键盘时, 注意关闭输入法

```
E325: 注意
发现交换文件 ".hello.py.swp"
所有者: python    日期: Wed Jul  8 16:04:19 2099
文件名: ~python/Desktop/hello.py
修改过: 是
用户名: python    主机名: ubuntu
进程 ID: 10754
正在打开文件 "hello.py"
日期: Wed Jul  8 15:21:23 2099

(1) Another program may be editing the same file. If this is the case,
be careful not to end up with two different instances of the same
file when making changes. Quit, or continue with caution.
(2) An edit session for this file crashed.
    如果是这样, 请用 ":recover" 或 "vim -r hello.py"
    恢复修改的内容 (请见 ":help recovery")。
    如果你已经进行了恢复, 请删除交换文件 ".hello.py.swp"
    以避免再看到此消息。

交换文件 ".hello.py.swp" 已存在!
以只读方式打开([O]), 直接编辑([E]), 恢复([R]), 删除交换文件([D]), 退出([Q]), 中>
止([A]):
```

关闭中文输入法
按下字母 d

编辑命令和数字连用---Esc 返回后会重复操作你在输入数字之后的操作,操作次数为数字

演练 1 —— 编辑命令和数字连用

- 在开发中，可能会遇到连续输入 **N** 个同样的字符

在 **Python** 中有简单的方法，但是其他语言中通常需要自己输入

- 例如：********* 连续 10 个星号

要实现这个效果可以在 **命令模式** 下

- 输入 **10**，表示要重复 10 次
- 输入 **i** 进入 **编辑模式**
- 输入 ***** 也就是重复的文字
- 按下 **ESC** 返回到 **命令模式**，返回之后 **vi** 就会把第 2、3 两步的操作重复 **10** 次

连续多行增加注释---运用可视块和编辑模式

演练 2 —— 利用 可视块 给多行代码增加注释

- 在开发中，可能会遇到一次性给多行代码 **增加注释** 的情况

在 **Python** 中，要给代码增加注释，可以在代码前增加一个 **#**

要实现这个效果可以在 **命令模式** 下

- 移动到要添加注释的 **第 1 行** 代码，按 **^** 来到行首
- 按 **CTRL + v** 进入 **可视块** 模式
- 使用 **j** 向下连续选中要添加的代码行
- 输入 **I** 进入 **编辑模式**，并在 **行首** 插入，注意：一定要使用 **I**
- 输入 **#** 也就是注释符号
- 按下 **ESC** 返回到 **命令模式**，返回之后 **vi** 会在之前选中的每一行代码 **前** 插入 **#**