目录

[vim基础入门 1](#_Toc477798673)

[基础 2](#_Toc477798674)

[历史 2](#_Toc477798675)

[模式 2](#_Toc477798676)

[模式化编辑器 2](#_Toc477798677)

[模式转换： 2](#_Toc477798678)

[启动vi: 3](#_Toc477798679)

[退出 3](#_Toc477798680)

[跳转 3](#_Toc477798681)

[词间跳转 3](#_Toc477798682)

[行号跳转 4](#_Toc477798683)

[删除 4](#_Toc477798684)

[可视化模式 4](#_Toc477798685)

[撤消 5](#_Toc477798686)

[搜索 5](#_Toc477798687)

[复制粘贴 5](#_Toc477798688)

[最简单的复制粘贴 5](#_Toc477798689)

[粘贴时保存原始格式 5](#_Toc477798690)

[查找并替换 5](#_Toc477798691)

[末行模式 6](#_Toc477798692)

[特殊用法 6](#_Toc477798693)

[窗口拆分 6](#_Toc477798694)

[打开多个文件 7](#_Toc477798695)

[定制属性 7](#_Toc477798696)

[批量注释 7](#_Toc477798697)

[删除注释 7](#_Toc477798698)

# vim基础入门

## 基础

### 历史

Linux 哲学：使用ASCII纯文本来保存程序配置信息

行编辑器：sed, Stream Editor

全屏幕编辑器：nano, emacs, vi(virtual interface) =》vim(vi improved)

### 模式

#### 模式化编辑器

编辑模式：命令 （默认模式）

输入模式：原样不动输入到屏幕上

末行模式：最下方一行可输入命令

#### 模式转换：

编辑---》输入

i: 当前光标所在处插入

a: 当前光标字符后面插入

o: 当前光标所在行下方新增空白行插入

I: 当前光标所在行的行首插入

A: 当前光标所在行的行尾插入

O: 当前光标所在行上方新增空白行输入

====================================

**c 命令，既包含模式转换又有删除功能**

cc 删除一行后进入输入模式

cw 删除一个词或当前光标到词末字符后进入输入模式

cb 删除光标所在前的词后进入输入模式

输入 –》 编辑

直接按Esc

编辑 –》 末行

：

末行 –》 编辑

Esc

输入 -xxx-> 末行

### 启动vi:

vi [option] FILE

目录必须先存在

打开文件时直接到达第n行

**vi +n file**

光标定位到最后一行

**vi + file**

光标定位到第一个指定的字符

**vi +/tag\_name file**

### 退出

末行模式下：q! 强制不保存退出

wq 保存退出

w! 强制保存

#### 跳转

##### 词间跳转

w跳到下一个单词的词首

e:跳转到后面一个的词尾

b:跳转到前面一个的词首

^ 非空行首

o 绝对行首

$ 行尾

##### 行号跳转

末行模式 键入行号

G 跳到最后一行

gg 跳到第一行

##### 删除

x:删除光标

dw 删除光标所在一个单词

dd 删除光标所在一行

末行模式：

:1,10d 删除第一行到第十行

:1,$d 删除所有行 ,$表示末行， . 表示光标所在的行号

p粘贴

删除或复制是整行，在当前光标所在行下粘贴

删除的是单词或行的部分，则在当前光标所在处前方粘贴

y 复制

y w b e ^ 0 $ 组合

yy 复制一行

### 可视化模式

v 按字符选取

V 按行选取

### 撤消

u 撤消上一次编辑操作

ctrl+r 撤消最后一次的撤消操作

#nu 撤消最近n次操作

### 搜索

/keyword

?keyword

会高亮关键词

### 复制粘贴

#### 最简单的复制粘贴

yy 复制

p 粘贴

#### 粘贴时保存原始格式

使用粘贴板里的数据时，会自动增加缩进，要保留原格式，必须要开启paste功能

:set paste

关闭

:set nopaste

### 查找并替换

: 定界s/查找的内容/替换的内容/gi

查找的内容：支持正则

替换的内容：不支持正则

修饰符：

g 全局替换

i ignore case

例子:

:1,20s/django/DJANGO/g

&用于引用查找到的所有字符串

例子：查找以n开头中间两个不定后以e结尾，加上r

:1,20s/n..e\>/&r/g

定界符：

% :表示所有行

1,20 表示1到20行

例:

注意如果有斜线，则要用反斜线转义

:%s/\/etc\/sysconfig\/network/&s/g

### 末行模式

与shell交互

:!shell command

保存数据

:1,50w /path/filename

### 特殊用法

#### 窗口拆分

ctrl-w, s:水平拆分

ctrl-w, v:垂直拆分

ctrl-w, q:关闭当前窗口

#### 打开多个文件

vim file1 file2

末行模式： next 切到下一个文件

prev 切到上一个文件

打开多个并水平显示

vim –o file1 file2

打开多个并垂直显示

vim –O file1 file2

#### 定制属性

末行模式下

:set [no]nu 显示、关闭行号

:syntax on[off] 开启语法高亮

如果要一直生效，则要修改配置文件

全局： /etc/vimrc

个人：~/.vimrc

#### 批量注释

1. 按v进入VISUAL模式
2. 选中要注释的行
3. 按ctrl+v进入VISUAL BLOCK模式
4. 大写I进入插入模式,输入要注释的字符串,如#或//（这里比较特殊，会以为是在光标行输入）
5. 按两次esc

#### 删除注释

1. 进入VISUAL BLOCK模式
2. 选中注释的字符串，按d