vi/vim 文本编 辑简介

北京市计算中心 云平台

vi 简介

vi 的三种工作 方式

vim 的环境发量

vi/vim 文本编辑简介

北京市计算中心 云平台

2016. 08

vi 简介

vi/vim 文本编 辑简介

北京市计算中 心 云平台

vi 简介

方式 的三种工作

vim 的环境到 量 vi 命令是 UNIX 操作系统和类 UNIX 操作系统中最通用的全屏幕 纯文本编辑器

Linux 中的 vi 编辑器叫 vim ,是 vi 的增强版 (vi Improved),与 vi 编辑器完全兼容,而且实现了很多增强功能 Ubuntu 中 vim 的安装:sudo apt-get install vim

(a) A new file



(b) An existing file

vi 的三种工作方式

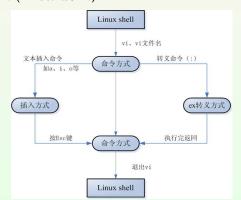
vi/vim 文本纬 辑简介

北京市计算中心 云平台

vi 简介

vi 的三种工作 方式

- 命令方式 (默认方式)
- 插入方式
- 末行方式 (ex 转义方式)



vi 的插入方式

vi/vim 文本纲 辑简介

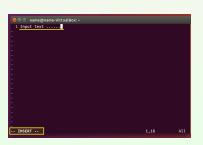
北京市计算中 心 云平台

vi 简介

vi 的三种工作 方式

Vim 的环境发量

■ 默认命令方式下按 "A"、"a"、"O"、"o"、"I"、"i" 键进入插入模式。



2 按Esc键退出插入模式

- a: 当前字符后添加文本
- A: 在行末添加文本
- i: 当前字符前插入文本
- I: 在行首插入文本
- o: 当前行后面插入一空行
- ○: 当前行前面插入一空行
- r: 替换光标位置的字符
- R: 替换光标开始的字符

vi 的命令方式

vi 的三种工作 方式

h: 光标向左移动一个字符

■ i: 光标移动至下一行

b: 光标向左移动一个词

G: 光标移动到最后一行

■ \$: 光标移动到本行末

■ x: 删除光标位置的字符

D: 删除光标位置到行末

■ dG: 删除光标位置到文件尾 dw: 删除光标位置的单词

u: 撤销上一次操作

!: 光标向右移动一个字符

k: 光标移动到上一行

w: 光标向右移动一个词

gg: 光标移动到第一行

^: 光标移动到本行第一个字符处

dd: 删除光标所在行

d0: 删除光标位置到行首

:重复上一次操作

■ vv: 复制当前行到编辑缓冲区

■ pp: 将编辑缓冲区内容拷贝到光标下一行

vi 的命令方式

vi/vim 文本约 辑简介

北京市计算中 心 云平台

vi 简介

vi 的三种工作 方式

vim 的坏项变 量

- Ctrl+u: 屏幕向上滚动半屏 Ctrl+d: 屏幕向下滚动半屏
- Ctrl+b: 屏幕向上滚动一屏 Ctrl+f: 屏幕向下滚动一屏
- Ctrl+g: 显示当前编辑文本信息
- ndd: 删除光标所在行起的n行字符
- nyy: 复制光标所在行起的n行字符
- :n: 光标跳转至第n行行首
- :\$: 光标跳转至末行行首
- ▼ZZ: 必要时写盘并退出 vi
- v: VISUAL 状态
- V:VISUAL LINE 状态
- Ctrl+v: VISUAL BLOCK 状态

vi 的末行方式

vi/vim 文本约 辑简介

北京市计算中 心 云平台

vi 简介

vi 的三种工作 方式

vim 的坏项变 量 ■ /exp:向前搜索字符串exp ?exp:向回搜索字符串exp

■ n: 重复前一搜索命令 N: 重复前一搜索命令

■ :w: 执行存盘操作 :w file: 编辑内容写入文件file

■ :w» file: 内容写到文件file原内容之后

■ :w! file: 强制写入文件file

■ :q: 退出 vi :q!: 强制退出 vi, 放弃编辑修改内容

■ :wq : 存盘并退出 vi

■ :x:必要时写盘并退出 vi

■ :s/old/new: 将光标所在行的第一个字符串old替换为new

■ :s/old/new/g: 将光标所在行的所有字符串old替换为new

vi 的末行方式

vi/vim 文本组 辑简介

北京市计算中心 云平台

V1 间**7** [

vi 的三种工作 方式

vim 的坏境变 量

- :n1,n2s/old/new/g: 将第N1到N2行的第一个字符串old替换为new
- :%s/old/new: 将所有行的第一个字符串old替换为new
- ■:%s/old/new/g: 将所有行的第一个字符串old替换为new
- :set nu: 编辑时显示行号 :set nonu: 编辑时不显示行号
- :set list: 显示不可见字符 :set nolist: 隐藏不可见字符
- :set all: 显示全部环境设置
- :!cmd: 在 vi 中执行 shell 命令cmd

vim 的标签窗口

vi/vim 文本纲 辑简介

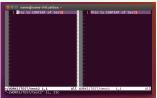
北京市计算中 心 云平台

V1 |BJJ|

vi 的三种工作 方式

- :tabnew file: 在 vi 新标签窗口中打开一个新的编辑文件file 按gt在标签窗口间切换
- :vertical diffsplit file: 对比当前文件和新文件file的差别 按Ctrl+w在标签窗口间切换





- :qall: 退出所有的标签窗口
- :qall!: 强制退出所有的标签窗口

vim 的环境变量设置文件 ~/.vimrc

vi/vim 文本编 辑简介

北京市计算中心 云平台

vi 简介

vi 的三种工作 方式

```
🚳 🖨 🗊 name@name-VirtualBox: ~
 6 " allow backspacing over everything in insert mode
 7 set backspace=indent.eol.start
 9 if has("vms")
     set nobackup
11 else
     set backup
13 endif
14 set history=50
15 set ruler
                          " show the cursor position all the time
16 set showcmd
17 set incsearch
18 set spell
19 set nu
20 set encoding=utf-8 fileencodings=utf-8.gbk.cp936.default.latin1
22 filetype plugin indent on
23 set completeopt=longest.menu
25 let g:SuperTabRetainCompletionType=2
".vimrc" 92 lines --5%--
                                                        5,0-1
                                                                       5%
```

Windows 下的 vim 的环境变量设置文件

vi/vim 文本编 辑简介

北京市计算中心 云平台

vi 简介

vi 的三种工作 方式

```
vimrc_example.vim =+ (C:/Program Files/Vim/vim74) - GVIM
文件(F) 编辑(E) 工具(T) 语法(S) 缓冲区(B) 窗口(W) 帮助(H)
급 및 및 급 | D G | X 는 다 다 및 및 및 조 스 스 A | Y 의 a | ? A
     set historu=58
                            " keep 50 lines of command line history
     et ruler
                           " show the cursor position all the time
     et showend
                           " displau incomplete commands
     set incsearch
                           " do incremental searching
     set spell
    set nu
     set encoding=cp936 fileencodings=utf-8 gbk cp936 default latin1
 22 Filetype pluqin indent on
 23 set completeopt=longest menu
 25 let q:SuperTabRetainCompletionType=2
 26 let g:SuperTabDefaultCompletionType="\C-X\\C-0\\"
 28 let Tlist Show One File=1
 29 let Tlist Exit OnluWindow=1
 31 let g:winManagerWindowLayout='FileExplorer|TagList'
 32 nmap wm :WMToggle<cr
 34 let q:miniBufExplMapCTabSwitchBufs - 1
 35 let g:miniBufExplMapWindowNavVim = 1
 36 let q:miniBufExplMapWindowNavArrows = 1
 37 nnoremap <silent> <F12> :A<CR
 39 " For Win32 GUI: remove 't' flag from 'quioptions': no tearoff menu entries
 48 map 0 gg
 42 inoremap <C-U> <C-G>u<C-U>
 44 " In many terminal emulators the mouse works just fine, thus enable it.
 45 if has('nouse'
 46 set nouse-a
 47 endif
 49 if &t Co > 2 || has("qui running")
      set hlsearch
     colorscheme elflord
```

vim 键盘分布图

vi/vim 文本纲 辑简介

北京市计算中 心 云平台

vim 的键盘分 布

