1.vi:visual文本文件编辑

2.vim:vi improved

**一.基本操作**

1.移动光标

normal模式下的操作：h(向左) j（向下）k(向上) l（向右）

2..VIM的进入和退出：

进入insert模式： a append 追加文本 i insert

回到normal模式： ESC

:q! 这种方式退出编辑器会丢弃进入编辑器以来的所做的改动。

3.文本编辑之删除：

删除光标所在位置的字符：x

4.文本编辑之插入：i 5.文本编辑之添加：A

6.编辑文件： :wq保存退出

**二.删除类命令：d [number] motion**

删除光标所在位置的字符：x

dw：从光标处删除至一个单词的末尾,不包括它的第一个字符。

de: 从光标处删除至一个单词的末尾,

d$:从当前光标删除至行末。

使用计数指定动作：

例如：2w: 使光标向后移动两个单词。

0：移动光标到行首。

使用计数以删除更多：d number motion

例如：d2w:删除两个大写字母单词

删除整行：dd

例如：删除两整行：2dd

**三. 撤销类命令: u U**

输入u来撤销最后执行的命令,输入多次u撤销之前的多次命令。

输入U撤销对整行的修改。

ctrl+r:可以重做被撤销的命令，也就是撤销掉撤销命令。

**四.置入类命令：**

1）dd 将该行删除，会将该行保存到vim的一个寄存器中。

2）将光标移动到准备置入的位置的上方

3）然后在normal模式下输入p将该行粘贴置入。

另一个置换类命令： 输入大写的R可连续输入多个字符。

**五.替换类命令：**

输入r和要替换成的字符。

**六.更改类命令： c [number] motion**

要改变文本直到一个单词的末尾：ce

(注意：ce命令不仅仅是删除了一个单词，也进入了插入模式)

**c$ 替换当前光标到行末的内容。**

**七. 定位及文件状态：**

ctrl+g 显示当前编辑文件中当前光标所在行位置以及文件状态信息。

G 直接跳转到文件中的最后一行

gg 使得当前光标直接跳转到文件第一行。

行号+G 返回到具体的某一行

**八.搜索类命令:**

/+字符串 在当前文件中查找该字符串

n 查找同上一次的字符串

N 向相反方向查找同上一次的字符串

用？替代/ 逆向查找字符串

ctrl+O 跳转回较旧的位置

ctrl+I 跳转到较新的位置

输入 :set ic (ignorecase)，查找 的是忽略大小写的选项

输入 :set noic 禁用忽略大小写

输入 :set is(incsearch) 查找短语时显示部分匹配

输入 :set hls(hlsearch) 高亮显示所有的匹配短语。

(选项名可以用完整版本，也可以使用缩略版本)

(在选项前加上no,可以关闭选项) 例如：set noic

**九.配对括号的查找：**

% 查找配对的括号 ) ] }

把光标放在任何一个 （ { [ 处，接着按% ，此时光标的位置应当在配对的括号处，再次按%，便可以跳回到配对的第一个括号处。

**十.替换命令：**

:s/old/new后的 改变光标所在行的第一个匹配串

:s/old/new/g（全局） 替换一行内所有的匹配串

替换两行之间出现的每个匹配串：

:#,#s/old/new/g #,#代表的是替换操作的若干行中首尾两行的行号

:%s/old/new/g 替换整个文件中的每个匹配串。

:%s/old/new/gc 找到整个文件中的每个匹配串，并且对每个匹配串提示是否进行替换。

**十一.在vim内执行外部命令的方法：:!command**

输入:! 然后接着输入一个外部命令可以执行该外部命令

**十二.关于保存文件的更多信息：**

要将对文件的改动保存到文件中: :w filename

获取当前目录的内容： :!dir 或者 :!ls

删除文件： 在MS-DOS下，输入： :!del filename

在Unix下，输入： :!rm filename

一个具有选择性的保存命令: v motion :w filename

v键使得Vim进入可视模式进行选取。四处移动光标使得选取区域变大或者变小

**十三.提取和合并文件：**

向当前文件中插入另外的文件的内容： :r filename

:r !dir 可以读取dir命令的输出并将其放置到当前文件的光标位置后面。

**十四.打开类命令：**

o 将在光标的下方打开新的一行并进入插入模式。

O 将在光标的上方打开新的一行

**十五. 附加类命令**

a 在光标之后插入文本。

**十六.复制粘贴文本：**

y复制，p粘贴

**十七.获取帮助信息：**

使用在线帮助系统的方法：

法一： 按HELP键

法二： 按F1键

法三： 输入 :help

进入帮助窗口后，输入 ctrl+w ,可以在窗口之间跳转。

输入 :q 关闭帮助窗口

提供一个正确的参数给 :help命令，可以找到关于该主题的帮助。

例如： :help w

:help insert-index

**十八.创建启动脚本：**

启用vim的特性

为了使用更多的特性，需要创建一个vimrc文件

1.开始编辑vimrc文件，具体命令取决于您所使用的操作系统：

:edit ~/.vimrc 这是Unix系统使用的命令

:edit $VIM/\_vimrc 这是MS-Windows系统使用的命令

2.接着读取vimrc示例内容：

:r $VIMRUNTIME/vimrc\_example.vim

3.保存文件： :write

下次启动Vim时，编辑器具有了语法高亮功能，可以把喜欢的各种设置添加到vimrc文件中。

了解更多信息请输入 :help vimrc-intro

**十九.补全功能：**

使用ctrl+d 和 Tab键 进行命令行补全。

1. 请确保 Vim 不是在以兼容模式运行： :set nocp

2. 查看一下当前目录下已经存在哪些文件，输入： :!ls 或者 :!dir

3. 现在输入一个目录的起始部分，例如输入： :e

4. 接着按 CTRL-D 键，Vim 会显示以 e 开始的命令的列表。

5. 然后按 <TAB> 键，Vim 会补全命令为 :edit 。

6. 现在添加一个空格，以及一个已有文件的文件名的起始部分，例如： :edit FIL

7. 接着按 <TAB> 键，Vim 会补全文件名(如果它是惟一匹配的)。

提示：补全对于许多命令都有效。您只需尝试按 CTRL-D 和 <TAB>。

它对于 :help 命令非常有用。

0 行

^ 当前行第一个非空字符

$ 行尾

w 下一个单词或标点符号的开头

e 到单词末尾，包括最后一个字符。

$ 到当前行末

W 下一个单词的开头，忽略标点符号

b (back) 上一个单词或标点符号的开头

B 上一个单词的开头，忽略标点符号

Ctrl+f 向下翻页

Ctrl+b 向上翻页

numberG 到第number行

G 文件末尾

在插入模式下如何前、后、上、下，移动光标？

ctrl-f, ctrl-b, ctrl-p, ctrl-n

在插入模式下快捷键如何删除当前光标前的一个单词？如何删除当前光标到行首？如何删除当前光标到行尾？

ctrl-w, ctrl-u, ctrl-k