A *Vim* Tutorial

sunlu.electric@gmail.com

December 7, 2021

Vi IMproved (Vim) is a clone, with additions, of Bill Joy's *vi* text editor program for *Unix*. *Vim* is a free and open-source software initially developed by Bram Moolenaar, and has become the default text editor of many (if not all) *Unix/Linux* based operating systems, some of which have only command-line based human-machine interface and *Vim* is the only build-in text editor. In 2006, *Vim* was voted the most popular editor amongst *Linux Journal* readers.

1 Brief Introduction to *Vim*

Vim is a text editor that supports both text user interface and graphical user interface. In its graphical interface, *gVim*, menus and toolbars for commonly used commands are integrated. However, it is mostly convenient and flexible when the text interface is used. *Vim* also provides supports for 100+ commonly used programming languages, thus is popular among programmers.

Vim has a built-in tutorial *vimtutor* for the beginners. In a *Linux* system where *Vim* is installed, running command `vimtutor` in shell starts the tutorial.

2 *Vim* Modes

A total of 12 different editing modes are defined in *Vim*, 6 of which are variants of the other 6 basic modes. Each mode has its unique functions and associated operating commands. Some of the most commonly used modes are given in Table 1.

Table 1: Commonly used modes in *Vim*.

Mode	Description
Normal	Default mode. It is used to navigate the cursor in the text, search and replace text pieces, and run basic text operations such as undo, redo, cut (delete), copy and paste.
Insert	It is used to insert keyboard inputs into the text, just like commonly used text editors today.
Visual	It is similar to Normal mode but areas of text can be highlighted. Normal mode commands can be used on the highlighted text.
Cmdline	It supports a single line command input, such as save and quit, at the bottom of the *Vim* window. After running this command line, Vim quits Cmdline mode automatically.

Table 2: Commonly used shortcut keys for mode switching.

Shortcut Key	Description
`Esc`	Quit current mode and switch to Normal mode.
`EscEsc`	Quickly quit current mode and switch to Normal mode.
`i`	Switch to Insert mode from Normal mode.
`I`	Switch to Insert mode from Normal mode and ignore blank space(s) of the current line.
`v`	Switch to Visual mode from Normal mode.
`:`	Switch to Cmdline mode from Normal mode.
`ce`	Delete the selected word and switch to Insert mode from Normal mode.
`c$`, `C`	Delete from the cursor to the end of the line and switch to Insert mode from Normal mode.
`s`	Delete the selected character and switch to Insert mode from Normal mode.
`S`	Delete current line and switch to Insert mode from Normal mode.
`o`	Create a new line beneath the cursor and switch to Insert mode from Normal mode.
`O`	Create a new line above the cursor and switch to Insert mode from Normal mode.
`a`	Move the cursor to next character and switch to Insert mode from Normal mode.
`A`	Move the cursor to the end of the line and switch to Insert mode from Normal mode, i.e. amend to the current line.

Table 3: Commonly used shortcut keys in Normal mode for navigation.

Shortcut Key	Description
`h`,`j`,`k`,`l`	Navigate one character by $\leftarrow$, $\downarrow$, $\uparrow$, $\rightarrow$ respectively.
`gj`,`gk`	Navigate on visual line by $\downarrow$, $\uparrow$ respectively.
(Number+)`w`	Navigate to the next word (repeat Number times).
(Number+)`W`	Navigate to the next word ignoring punctuation (repeat Number times).
(Number+)`b`	Navigate to the previous word (repeat Number times).
(Number+)`B`	Navigate to the previous word ignoring punctuation (repeat Number times).
`g`	Navigate to the beginning of the current line.
`gg`	Navigate to the first character of the text.
`G`	Navigate to the first character of the last line of the text.
Number + `G`	Navigate to the first character of Number-th line of the text.
`Ctrl+o`	Unplace cursor position.
`Ctrl+i`	Replace cursor position.

3 Shortcut Keys

Vim is highly customizable and shortcut keys can be defined and mapped with complicated operations. Some commonly built-in shortcut keys are given in Tables 2, 3, 4, 5 and 6.

Table 4: Commonly used shortcut keys in Normal mode for cut (delete), copy, paste and other basic operations.

Shortcut Key	Description
(Number+)`x`	Cut the selected character (repeat Number times).
(Number+)`dw`	Cut the selected word (repeat Number times).
`d$`	Cut from the cursor to the end of the line.
(Number+)`dd`	Cut the selected line (repeat Number times).
`yw`	Copy the selected word.
`y$`	Copy from the cursor to the end of the line.
`yy`	Copy the selected line. To copy multiple lines, it is mostly convenient to use Visual mode.
`P`	Paste the cut or copied content at the cursor.
`r`+Character	Replace the selected character with Character.
(Number+)`u`	Undo (repeat Number times).
(Number+)`R`	Redo (repeat Number times).

Table 5: Commonly used shortcut keys for search and replace.

Shortcut Key	Description
`f`+Character	Find the next appearance of Character.
`F`+Character	Find the previous appearance of Character.
`/`+Word+`Enter`	Find the next appearance of Word.
`?`+Word+`Enter`	Find the next appearance of Word.
	Once an appearance of the word is found, use `n` to direct to the previous occurrence and `N` to the next occurrence.
`:%s/OldString/NewString/g`	Replace `OldString` with `NewString` in the text.
`:L1,L2 s/OldString/NewString/g`	Replace `OldString` with `NewString` from `L1` to `L2` of the text.
	Replace `/g` with `/gc` so that each replacement of an instance needs confirm "Y/N".

Table 6: Commonly used shortcut keys for save, quit and OS admin.

Shortcut Key	Description
`:quit`, `:q`, `:exit`, `:x`, `ZZ`	Quit.
`:q!`	Quit without save.
`:wq`	Save and quit.
`:!`+LinuxCommand+`Enter`	This would temporarily return to OS and run LinuxCommand, then re-run *Vim* environment.