

Basic Vi Commands

What is **Vi**?

- It is the default text editor that comes with the Linux/Unix operating system. Vi is a full-screen, powerful command-based editor. You can use vi to create or edit files in your Linux/Unix system, e.g., editing some configuration files to change some settings of your system or to set appropriate environments for your software.
- The way you manipulate text in vi is different from other common text editors, such as Notepad or Notepad++.

Vi Modes

Vi has two modes of operation:

- **Insert mode:** for editing a file. The insert mode begins upon entering an insertion or change command. Pressing the <Esc> (*Escape*) key turns off the Insert mode.
- **Command mode:** for issuing vi commands. The editor begins in the command mode. Every character typed is a command that does something to the text file being edited, or it may even cause the vi editor to enter the insert mode.

Access and Exit Vi


- Type **vi** (or **vim**) at the prompt to access the vi editor. You will be using the improved vi editor (VIM). Vi is case-sensitive.
- Type **vi filename**, if the file named filename exists, the first page (or screen) will be displayed; if the file does not exist, then an empty file and screen are created into which you may enter text.
- To exit vi, first press the <Esc> (*Escape*) key, and type one of the following:
 - **:wq** ↵ saves the changes to the file and quits vi
 - **:x** ↵ the same as **:wq**
 - **:q** ↵ quits vi if no change has been made
 - **:q!** ↵ quits vi without saving the changes
- In CentOS, you can type **ZZ** in the command mode to exit vi and save changes (note: no colon is used).

Insert Mode: Inserting or Adding Text

Command	Functionality
i	Insert text before cursor
I	Insert text at the beginning of current line
a	Append text after cursor
A	Append text of the end of current line
o	Insert a new line below the current line
O	Insert a new line above the current line

Command Mode: Moving the Cursor

h	Move left one space
j	Move down one space
k	Move up one space
l	Move right one space
w	Move to next word
b	Move to beginning of word
e	Move to end of word
(Move one sentence back
)	Move one sentence forward
{	Move one paragraph back
}	Move one paragraph forward

0 _(zero)	Move to beginning of line
\$	Move to end of line
1G	Move to 1 st line
nG	Move to n th line
G	Move to last line
:n 	Move to n th line
H	Move to top of screen
M	Move to middle of screen
L	Move to bottom of screen
%	Move to associated (), {}, or []


Command Mode: Moving the Screen

Ctrl-f	Move forward one screen
Ctrl-b	Move backward one screen
Ctrl-d	Move forward one half screen
Ctrl-u	Move backward one half screen


Commands for Modifying Text

r	Replace single character under cursor
R	Replace characters, starting with current cursor position, until an <Esc> hit
cw	Change the current word with new text, starting with the character under cursor, until an <Esc> hit
cNw	Change N words beginning with character under cursor, until an <Esc> hit; e.g., c2w changes 2 words
C	Change (replace) the characters from the current cursor position to the end of the line
cc	Change (replace) the entire current line, and stop when <Esc> is hit
Ncc or cNc	Change (replace) the next N lines, starting with the current line, and stop when <Esc> is hit

Commands for Deleting Text

x	Delete single character under cursor
Nx	Delete N characters, starting with character under cursor
dw	Delete the single word beginning with character under cursor
dNw	Delete N words beginning with character under cursor; e.g., d5w deletes 5 words
D	Delete the remainder of the line, starting with current cursor position
dd	Delete the entire current line
:d 	Delete the entire current line
Ndd or dNd	Delete N lines, beginning with the current line, e.g., 5dd deletes 5 lines

Commands for Cutting and Pasting

yy	Copy (or "yank") the current line into the buffer
:y 	Copy (or "yank") the current line into the buffer
Nyy or yNy	Copy (or "yank") the next N lines, including the current line, into the buffer
p	Paste the line(s) in the buffer into the text after the current line

Commands for Searching Text

/string	Search forward for occurrence of the string in text
?string	Search backward for occurrence of the string in text
n	Move to next occurrence of search string
N	Move to next occurrence of search string in opposite direction

Other Useful Commands

Command	Functionality
u	Undo last change
U	Undo all the changes to the line
.	Repeat the last text-changing command
J	Join lines
~	Toggle upper and lower case
:set nu(mber)	Turn on line number
:set nu! :set nonumber :set nonu	Turn off line number