

Text Editors

nano:

1. **nano** is easy to use, and requires very little effort to learn.
2. To open a file, type **nano <filename>** and press Enter.
3. If the file does not exist, it will be created.
4. nano provides a two line shortcut bar at the bottom of the screen that lists the available commands. Some of these commands are:
 - **CTRL-G** - Display the help screen.
 - **CTRL-O** - Write to a file.
 - **CTRL-X** - Exit a file.
 - **CTRL-R** - Insert contents from another file to the current buffer.
 - **CTRL-C** - Show cursor position.

Introduction to vi:

1. The actual program installed on your system is **vim**, which stands for **Vi IMproved**, and is aliased to the name **vi**.
2. **vi** provides three modes, as described in the table below. It is vital to not lose track of which mode you are in. Many keystrokes and commands behave quite differently in different modes.

Mode	Feature
Command	<ul style="list-style-type: none">• By default, vi starts in Command mode.• Each key is an editor command.• Keyboard strokes are interpreted as commands that can modify file contents.
Insert	<ul style="list-style-type: none">• Type i to switch to Insert mode from Command mode.• Insert mode is used to enter (insert) text into a file.• Insert mode is indicated by an “? INSERT ?” indicator at the bottom of the screen.• Press Esc to exit Insert mode and return to Command mode.
Line	<ul style="list-style-type: none">• Type : to switch to the Line mode from Command mode. Each key is an external command, including operations such as writing the file contents to disk or exiting.• Uses line editing commands inherited from older line editors. Most of these commands are actually no longer used. Some line editing commands are very powerful.• Press Esc to exit Line mode and return to Command mode.

Working with Files in vi:

1. The below table describes the most important commands used to start, exit, read, and write files in vi. The ENTER key needs to be pressed after all of these commands.

Command	Usage
<code>vi myfile</code>	Start the editor and edit myfile
<code>vi -r myfile</code>	Start and edit myfile in recovery mode from a system crash
<code>:r file2</code>	Read in file2 and insert at current position
<code>:w</code>	Write to the file
<code>:w myfile</code>	Write out to myfile
<code>:w! file2</code>	Overwrite file2
<code>:x</code> or <code>:wq</code>	Exit and write out modified file
<code>:q</code>	Quit
<code>:q!</code>	Quit even though modifications have not been saved

Changing Cursor Positions in vi:

1. The below table describes the most important keystrokes used when changing cursor position in vi. Line mode commands (those following colon :) require the ENTER key to be pressed after the command is typed.

Key	Usage
arrow keys	To move up, down, left and right
j or <ret>	To move one line down
k	To move one line up
h or Backspace	To move one character left
l or Space	To move one character right
0	To move to beginning of line
\$	To move to end of line
w	To move to beginning of next word
:0 or 1G	To move to beginning of file
:n or nG	To move to line n
:\$ or G	To move to last line in file
CTRL-F or Page Down	To move forward one page
CTRL-B or Page Up	To move backward one page
^l	To refresh and center screen

Searching for Text in vi:

1. The table describes the most important commands used when searching for text in vi. The ENTER key should be pressed after typing the search pattern.

Command	Usage
/pattern	Search forward for pattern
?pattern	Search backward for pattern

2. The table describes the most important keystrokes used when searching for text in vi.

Key	Usage
n	Move to next occurrence of search pattern
N	Move to previous occurrence of search pattern

Working with Text in vi:

1. The table describes the most important keystrokes used when changing, adding, and deleting text in vi.

Key	Usage
a	Append text after cursor; stop upon Escape key
A	Append text at end of current line; stop upon Escape key
i	Insert text before cursor; stop upon Escape key
I	Insert text at beginning of current line; stop upon Escape key
o	Start a new line below current line, insert text there; stop upon Escape key
O	Start a new line above current line, insert text there; stop upon Escape key
r	Replace character at current position
R	Replace text starting with current position; stop upon Escape key
x	Delete character at current position
Nx	Delete N characters, starting at current position
dw	Delete the word at the current position
D	Delete the rest of the current line
dd	Delete the current line
Ndd or dNd	Delete N lines
u	Undo the previous operation
yy	Yank (copy) the current line and put it in buffer
Nyy or yNy	Yank (copy) N lines and put it in buffer
p	Paste at the current position the yanked line or lines from the buffer