

Modes of Operation

Command Mode - keystrokes are interpreted as commands to control vi's operation. vi will always start in command mode.

Input Mode - keystrokes are interpreted as data to be inserted into the file.

Cursor Movement

h	left	^b	scroll backward/up full-screen/24 lines
j	down/forward	^u	scroll up/backward half-screen/12 lines
k	up/back	^d	scroll down/forward half-screen/12 lines
l (e _l)	right	^f	scroll forward/down full-screen 24 lines

G The "Go to" Key puts the cursor on the last line in the file. When preceded by a number, it will put the cursor on that line number.

Inserting Characters

All of these commands put vi into the input mode:

i inserts characters before the cursor position.
I inserts characters at the beginning of the line

a inserts characters after the cursor position.
A inserts characters at the end of the line

O opens a new line to start inserting data above the cursor line.
o opens a new line to start inserting data below the cursor line.

The `ESC` key puts vi back into the command mode.

Deleting Characters

x deletes the character the cursor is on.
X deletes the character to the left of the cursor position.

dd deletes the line the cursor is on. This command puts the line it is on in a special vi buffer. The contents of this buffer can be retrieved at any time with the vi "put" (P or p) command. See **Copying and Moving Text** section.

These commands can be preceded by a count that will affect that many characters (x or X) or lines (dd).

Replacing Characters

R puts vi into a replace or "type-over" mode. Press the `ESC` key to get back to command mode.

r lets you replace the character the cursor is on with the next one typed.

Searching for Text

- / allows you to search forward through the text for a pattern.
- ? allows you to search backward.

Follow the command with the pattern you want to find, then press return. The cursor will be positioned at the next occurrence (/), or previous occurrence (?), of the pattern.

- n repeats the previous search in the same direction
- N repeats the previous search in the opposite direction

Copying and Moving Text

yY The "yank" command copies one or more lines. Like the **dd** command, the **yY** command places the lines it acts on in a special vi buffer. The contents of this buffer can be retrieved at any time with the vi "put" (**P** or **p**) command.

- P** puts the contents of the special buffer in the text above the line the cursor is on.
- p** puts the contents of the special buffer in the text below the line the cursor is on.

Moving text is accomplished by first deleting the desired text object, moving the cursor to the new desired position, then using the "put" (**P** or **p**) command to put the text back from the buffer.

Slightly Advanced Features

- If **w** moves the cursor to the beginning of the next word, and
- if **b** moves the cursor to the beginning of the previous word, and
- if **\$** moves the cursor to the end of the line, and
- if **0** moves the cursor to the beginning of the line, then these commands can be used in combination with the **d** (delete), **y** (yank), and **c** (change) commands in interesting ways.

For example:

- | | | | |
|------------|----------------------------|------------|------------------------|
| cw | changes word | c3w | changes three words |
| d\$ | deletes to end of line | d6b | deletes six words back |
| y0 | yanks to beginning of line | y2w | yanks two words |

Starting and Stopping the Editor

To start editing a file with vi, you should type "vi filename" at the UNIX shell prompt. To stop the editor and return to the shell you have several options:

- ZZ** saves any changes you have made to your file, ends your editing session with vi, and returns you to the shell prompt.
- :wq ret** saves any changes you have made to your file, ends your editing session with vi, and returns you to the shell prompt.
- :q ret** to quit if you haven't made any changes.
- :q! ret** to quit if you have made changes but don't want to save them.