

### Vi introduction :

- The vi editor (short for visual editor) is a **screen editor** which is available on almost all Unix systems.
- The vi has **no menus** but instead uses **combinations of keystrokes** in order to accomplish commands.
- In this screen each **line beginning with a tildes (~)**. In this, the last line is **reserved for some commands**.
- When you open a file with vi, **the cursor is positioned at the top-left** hand corner of the screen.
- In VI editor all the **command can be perform with the ESC**.

### Starting the vi Editor

There are following way you can start using vi editor –

Command	Description
<b>vi filename</b>	Creates a new file if it already does not exist, otherwise opens existing file.
<b>vi -R filename</b>	Opens an existing file in read only mode.
<b>view filename</b>	Opens an existing file in read only mode.

Create your new file:

```
$vi testfile
```

As a result you would see a screen something like as follows –

You will notice a tilde (~) on each line following the cursor. A tilde represents an unused line. If a line does not begin with a tilde and appears to be blank, there is a space, tab, newline, or some other nonviewable character present.

So now you have opened one file to start with. Before proceeding further let us understanding few minor but important concepts explained below.

**There are 3 modes available:**

## 1. Command mode :

- ☞ This is the **default mode of vi**.
- ☞ These Commands cause **action to be taken on the file**.
- ☞ This mode is used to give some **command for navigation**, edition and copy or cut.
- ☞ This is the base mode of vi editor **if user want to switch this mode** to insert or input mode then user has to **press i or I or a or A**.
- ☞ In this mode, whatever you type is interpreted as a command.

## 2. Insert mode:

- ☞ **Insert mode** in which entered text is **inserted into the file**.
- ☞ This mode permits insertion of new text, editing of existing text or replacement of existing text or replacement of existing text.
- ☞ This mode can be invoked changing in insertion mode.

### 3. Ex mode:

- ☞ **Execute mode in** which after editing **save** and **quit** the file.
- ☞ This mode permits us to give command at the command line, and The bottom line of the VI screen is called the command line.

#### ❖ Insert Mode Command :

I	Insert text at the beginning of the line
i	Insert text before the cursor.(left of cursor)
A	Appends the text at the end of the line.
a	Append the text to after cursor.(right of cursor)
O	Insert a blank line above the line your cursor is currently located on and put text, until hit ESC.
o	Insert a blank line below the line your cursor is currently located on and put text, until hit ESC.
R	Replace text from cursor to right.
r	This replaces the one under the cursor.
S	Replace entire line.

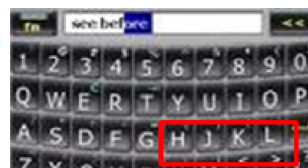
#### ❖ Cursor movement Command :

**h key** move cursor to the **left** one position

**l key** move cursor right **one** position

**j key** move cursor **down** one line

**k key** move cursor **up** one line



❖ **Closing and saving files :**

:q	Quit after a file saved. if file has not been saved, vi refused to quit without saving.
:q!	Quit without saving is
:w	Save the contents of the file.
:wq	Quit and save contents of the file.
ZZ	Save and exit the file.

❖ **Moving among words and lines :**

W	Moves the cursor forward one word
B	Moves the cursor backward one word
E	Moves to the end of a word

NOTE :

W,B,E command may be used to move between words but ignore punctuation. while w,b,e can define punctuation or space.

❖ **Screen movement command :**

H	Move cursor to the top line of the screen
M	Move cursor to the middle line of the screen
L	Move cursor to the last line of the screen
Ctrl – f	Move forward one screen
Ctrl – b	Move backward one screen
Ctrl – u	Move the screen backward half of the screen
Ctrl – d	Move the screen forward half of the screen

❖ Cutting & copy , paste, joining text in Vi :

<b>d\$, D</b>	Deletes from <b>current cursor position</b> to <b>the end of the line</b>
<b>dw</b>	Deletes from <b>current cursor position</b> to <b>the end of the word</b>
<b>dd</b>	Deletes <b>one line</b> from current cursor position.
<b>dd</b>	Delete <b>(cut) 1 line</b> from current cursor position
<b>2dd</b>	Delete <b>(cut) 2 lines</b> ( <b>3dd</b> to cut 2 lines, etc.)
<b>x</b>	Delete a <b>character</b>
<b>yy</b>	Yank <b>(copy) a single line</b>
<b>2yy</b>	Yank (copy) <b>2 lines</b> ( <b>3yy</b> to copy 3 lines, etc.)
<b>p</b>	<b>Paste lines below</b> current line.
<b>P</b>	<b>Paste lines before</b> current line.
<b>u</b>	<b>Undoes the last changes</b> you made <b>anywhere</b> in the file.
<b>U</b>	<b>Undoes all recent changes</b> to the current line. You can't have moved from the line to recover the original line.
<b>J</b>	Place your cursor on the first line to be joined text and type 'J', <b>its join 1 line.</b>
<b>4J</b>	Joins 4 line.

❖ Vi string/search

- **/[pattern]** search forward for the pattern
- **?[pattern]** search backward for the pattern

**For example:**

1. Search forward for the next line containing the string **Title** by typing **/Title**
2. Search backward for the most recent instance of Title by typing **?Title**

❖ Moving within a File Command

<b>0 or  </b>	Positions cursor at beginning of line.
<b>\$</b>	Positions cursor at end of line.
<b>(</b>	Positions cursor to beginning of current sentence.
<b>)</b>	Positions cursor to beginning of next sentence.
<b>{</b>	move a paragraph back
<b>}</b>	Move a paragraph forward
<b>[[</b>	Move a section back
<b>]]</b>	Move a section forward
<b>n </b>	Moves to the column <b>n</b> in the current line
<b>1G</b>	Move to the first line of the file
<b>G</b>	Move to the last line of the file
<b>nG</b>	Move to <b>n</b> th line of the file
<b>:n</b>	Move to <b>n</b> th line of the file