

# vi basic commands

## Summary of most useful commands

©Copyright 2013-2005, Free Electrons, <http://free-electrons.com>. Latest update: Jul 23, 2014  
Free to share under the terms of the Creative Commons Attribution-ShareAlike 3.0 license. Sources, translations and updates on our free training materials: [http://free-electrons.com/training/intro\\_unix\\_linux](http://free-electrons.com/training/intro_unix_linux). Thanks to: Liubo Chen.

## Entering command mode

[Esc] Exit editing mode. Keyboard keys now interpreted as commands.

## Moving the cursor

**h** (or left arrow key) move the cursor left.  
**l** (or right arrow key) move the cursor right.  
**j** (or down arrow key) move the cursor down.  
**k** (or up arrow key) move the cursor up.  
[Ctrl] **f** move the cursor one page **f**orward .  
[Ctrl] **b** move the cursor one page **b**ackward.  
**^** move cursor to the first non-white character in the current line.  
**\$** move the cursor to the end of the current line.  
**G** go to the last line in the file.  
**nG** go to line number *n*.  
[Ctrl] **G** display the name of the current file and the cursor position in it.

## Entering editing mode

**i** insert new text before the cursor.  
**a** append new text after the cursor.  
**o** start to edit a new line after the current one.  
**O** start to edit a new line before the current one.

## Replacing characters, lines and words

**r** replace the current character (does not enter edit mode).  
**s** enter edit mode and **s**ubstitute the current character by several ones.  
**cw** enter edit mode and **c**hange the **w**ord after the cursor.  
**C** enter edit mode and **c**hange the rest of the line after the cursor.

## Copying and pasting

**yy** copy (**y**ank) the current line to the copy/paste buffer.  
**p** paste the copy/paste buffer after the current line.  
**P** Paste the copy/paste buffer before the current line.

## Deleting characters, words and lines

All deleted characters, words and lines are copied to the copy/paste buffer.

**x** delete the character at the cursor location.

**dw** delete the current word.  
**D** delete the remainder of the line after the cursor.  
**dd** delete the current line.

## Repeating commands

**.** repeat the last insertion, replacement or delete command.

## Looking for strings

**/string** find the first occurrence of *string* after the cursor.  
**?string** find the first occurrence of *string* before the cursor.  
**n** find the **n**ext occurrence in the last search.

## Replacing strings

Can also be done manually, searching and replacing once, and then using **n** (next occurrence) and **.** (repeat last edit).

**n,ps/str1/str2/g** between line numbers *n* and *p*, substitute all (**g**: global) occurrences of *str1* by *str2*.  
**1,\$s/str1/str2/g** in the whole file (**\$**: last line), substitute all occurrences of *str1* by *str2*.

## Applying a command several times - Examples

**5j** move the cursor 5 lines down.  
**30dd** delete 30 lines.  
**4cw** change 4 **w**ords from the cursor.  
**1G** go to the first line in the file.

## Misc

[Ctrl] **l** redraw the screen.  
**J** join the current line with the next one  
**u** undo the last action

## Exiting and saving

**ZZ** save current file and exit vi.  
**:w** **w**rite (save) to the current file.  
**:w file** **w**rite (save) to the *file* file.  
**:q!** **q**uit vi without saving changes.

## Going further

**vi** has much more flexibility and many more commands for power users!  
It can make you extremely productive in editing and creating text.

Learn more by taking the quick tutorial: just type **vimtutor**.

