

# THE ONE PAGE FOR... VIM!

By: Rudson Ribeiro Alves and Almir Mendes de Araújo[m3nd3s]

Last update in: 31/05/2009

Source: vim manual, vimtutor

## Vim (Vi IMproved)

Vim is a text editor that is upwards compatible to Vi. It can be used to edit all kinds of plain text. It is especially useful for editing programs.

## Basic synopsis

`Vim [options] [file]`

The options may be given in any order, before or after filenames. Options without an argument can be combined after a single dash.

### +**[num]**

For the first file the cursor will be positioned on line "num".

### -c **{command}**

{command} will be executed after the first file has been read. If the {command} contains spaces it must be enclosed in double quotes.

-g If Vim has been compiled with GUI support, this option enables the GUI. If no GUI support was compiled in, an error message is given and Vim aborts.

-x Use encryption when writing files. Will prompt for a crypt key.

-y Start Vim in easy mode, just like the executable was called "evim" or "eview". Makes Vim behave like a click-and-type editor.

## Files

`/usr/share/vim/vimrc`

System wide Vim initializations.

`~/.vimrc`

Your personal Vim initializations.

`.vim`

Your personal Vim customization path

## Modes

The Vim editor is a modal editor. That means that the editor behaves differently, depending on which mode you are in. If the bottom of the screen displays the file-name or is blank, you are in normal mode.

**[ESC]**

normal mode

**i** insert mode, indicator displays --INSERT--

**v** visual mode, the indicator shows --VISUAL--

## Moving cursor

**h** key or left ket to move left

**l** key or right key to move right

**k** key or up key to move up

**j** key or down key to move down

**0** move to the start of the line

**\$** move to the and of the line

**w** move to next word

**e** move to the end to the next word

## Delete/Undo

**x** delete the character under the cursor

**d[n]d** or **[n]dd**

to delete a whole n lines

**d[n]w** or **[n]dw**

delete n words from the cursor to the end of a line

**df[letter]** delete everything until [letter]

**d\$** delete from the cursor to the end of a line

**d[moving key]**

delete in cursor moving direction

**u** undo the last commands

**CTRL-R** redo unded commands

**U** return the line to its original state

## Paste/Replace

**p** put previously deleted text after the cursor

**r[letter]**

replace the character at the cursor with [letter]

**ce** to change until the end of a word. Delete the word and places you in Insert mode

**c[n][motion]**

to change n words. Delete [n] words in the motion direction and places you in insert mode

**c\$** to change to the end of line. Delete to the end of line and places you in insert mode

## Cusor Location/File Status

**G** to move you to the bottom of the file

**gg** to move you to the start of the file

**CTRL-G**

to show your location in the file and the file status

## Search

**/** followed by a phrase to search for the phrase

**n** to search the next occurrence

**N** to previous occurrence

**?** instead of **/**, to search for a phrase in the backward direction

## Substitute Command

---

**:s/old/new**

to substitute the first occurrence of 'old' by the 'new'

**:s/old/new/g**

to substitute the all occurrence of 'old' by the 'new' in current line

**:%s/old/new/g**

to substitute the all occurrence of 'old' by the 'new' in all file

**:%s/old/new/gc**

use 'c' in the end of sentence to prompt for substitute or not

## External command and save file

---

**:[command]**

to execute an external command ( **:!ls** – list current directory)

**:w <filename>**

write text with <filename>

**:q**

quit

**:q!** or **ZQ**

trash all changes and quit

**:wq** or **:x** or **ZZ**

write and quit

## Select text to write/insert file

---

**v motion :w FILENAME**

v starts Visual mode. You can move the cursor around to make the selection bigger or smaller. Then you can use an operator to do something with the text. For example, **d** deletes the text

**:r FILENAME**

to insert the contents of a file

**:r !ls**

reads the output of the ls command and puts it below the cursor

## Open/Append Command

---

**o** to open a line below the cursor and place you in Insert mode

**O** to open a line above the cursor and place you in Insert mode

**a** to insert text after the cursor

**A** to insert text in the end of the line

**R** to replace more than one character

## Copy and Paste text

---

**y** in visual mode (press v) to copy text, and **p** to paste it::

1. Start Visual mode with **v** or **V** (line mode) and move the cursor to just before "first";
2. Type **y** to yank (copy) the highlighted text;
3. Move the cursor to insert point;
4. Type **p** to put (paste) the text.

**yy** Yank the current line. Use **p** to paste.

**[n]yy** Yank [n] lines. After, move the cursor to insert point and press **p** to paste.

## Using tabs in vim

---

**vim -p file1 file2 file3 ...**

to open one tab page for each file

**:tabf[ind] <mask/file/dir>**

to search for a file in your current path and open it in a new tab

**:tabnew <file/dir>** or **:tabe[dit] <file/dir>**

to open a new tab

**:tabc[lose][!]**

to close current tab page. **!** to trash all changes

**<CTRL+PageUp>** or **gT**

go to the previous tab page

**<CTRL+PageDown>** or **gt**

go to the next tab page