

0 ^ B b h l e w E W \$

beginning of line first non-blank character previous WORD previous word previous character next character end of word beginning of next word end of WORD beginning of next WORD end of line

:e path :e! :view path :r file :q :q!  
load and edit return to load file in insert con- quit quit, discard  
given file unmodified file read-only mode tents of file changes

:w :saveas path :browse old :cd :pwd  
write file save with new name List recent files, allow to change dir echo current  
select one by number. working dir

^o previous jump location  
gg beginning of file  
^b back 1 page  
^u up 1/2 page  
{ beginning of paragraph  
k up 1 line  
j down 1 line  
} end of paragraph  
^d down 1/2 page  
^f forward 1 page  
G end of file  
8G to first non-blank char of line 8  
:8 to start of line 8

## Buffers

:ls :b n :b file  
list all buffers jump to buffer nr n jump to buffer with containing file

:bp :bn :bd  
jump to prev buffer jump to next buffer delete current buffer

:e path :enew :bufdo C  
load and edit given file new blank buffer apply command c to all buffers

## Inserting

I i a A  
beginning of this line before cursor after cursor end of this line

O o s S  
beginning of previous line beginning of next line substitute character substitute line

C cw r R  
change rest of line change rest of word replace 1 character replace from cursor

## Deleting

x diw dw d/ foo  
delete character delete all of word delete rest of word delete until first 'foo'

dd d0 d^ d\$  
delete line del from BOL to cursor delete 1st non-blank char to cursor del from cursor to EOL

d} :1,.d :.,\$d dt c  
del to end of paragr. del from line 1 to cursor del from curs to EOF delete up to char c

## Replace

:s/foo/bar/g  
In current line replace all (/g) 'foo' with 'bar'

:%s/foo/bar/g  
In all lines replace all 'foo' with 'bar'

:%s/foo/bar/gic  
In all lines replace all 'foo' with 'bar', case-insensitive, and ask for confirmation

:3,9s/foo/bar/g  
In lines 3 to 9 (incl) replace all 'foo' with 'bar'

:.,.+5/^/ /g  
From current line to 5 line further, prepend 3 spaces at the beginning of each line

:s/\In current line replace all occurrences of 'foo' that are a separate word, with 'bar'

:5,\$s/foo//g  
From current line to last line, replace all 'foo' with nothing (= delete 'foo')

:%s/\t/ /g  
In all lines, replace tab with 4 spaces

:%s/red\|pink/blue/g  
In all lines, replace all instances of "red" and "pink" with "blue"

: '<,'>s/foo/bar/g  
In the visually pre-selected lines, replace foo with bar.

## Selecting (visual)

v V ^v gv  
select per character select per line select rect-angular block reselect prev selection

d y P p  
delete selection yank (copy) selection paste before cursor paste after cursor

o O aw iw  
go to start of selection go to end of selection select word & whitespace select inner word

a", a', a(, a[, aB, at  
i", i', i(, i[, iB, it

Version with a: include "" " () [] {} <>  
Version with i: exclude "" " () [] {} <>

## Copying

yy yiw yw  
yank this line yank inner word yank rest of word

y0 y^ y\$  
yank from BOL to cursor yank from 1st non-blank char to cursor yank from cursor to EOL

## Undo/Redo

u :earlier 4m  
undo take document to state of 4 min ago

^r :later 50s  
redo 50 sec later in document state

## Search

\* #  
Go to next match of exact current word Go to prev match of exact current word

g\* g#  
Idem as \* but not exact Idem as # but not exact

/foo ?foo  
Search forward for foo, not exact Search backward for foo, not exact

n N  
Go to next match Go to prev match

/\Search forward for foo, exact no search highlights

## Move

:m 5 :m 0 :m \$ :m ' }-1  
mv current line to after line 5 mv current line to before line 1 mv current line to after last line mv current line to end of paragraph

:2,4m \$ :2,4m 9 :.,.+2m 9  
mv lines 2,3,4 to after last line mv lines 2,3,4 to after line 9 mv 3 lines starting from the current one to after line 9

## vim-multiple-cursors

^n Selects current word in visual mode, and puts cursor at end of it. Pressing it again find next occurrence and puts a cursor at the end.

^p Remove last virtual cursor and go back to last one.

^x Skip last virtual cursor and go to next occurrence of the word.

After all cursors are placed, you may opt to go out of visual mode (press v) and use the normal commands.

## Search/Replace in multiple files

:vimgrep /foo/g \*.py Search in all files. Use :copen to get a list of matches.

:bufdo %s/foo/bar/ge | update  
Apply to all buffers. Ignore pattern not found errors (e). Save file if changes were made (update).

:arg \*.cpp Create arglist with C++ files and load them into buffers  
:argadd \*.h Add header files to arglist, and load into buffers  
:argdo %s/foo/bar/ge | update Apply to every file in arglist

netrw & Ctrl-p	
<b>:edit read write sftp://user@hostname/path</b>	
<b>:edit</b> folder	(folder can be . )
<b>:Vexplore</b>	split left-right and show file browser
<b>i</b>	toggle between file browser view options
<b>/path</b>	move cursor to nearest match in file browser
<b>:^p filename</b>	Fuzzy search file. Use ^f and ^b to toggle between files, buffers, and <u>most recently used</u> . Can be used for “ <b>Open Recent...</b> ”

Global command	
<b>: [range] g/pattern/command</b>	
<b>:g/foo/d</b>	Delete all lines that contain ‘foo’
<b>:.,\$g!/foo/m\$</b>	Move all lines from cursor to EOF, that do not contain ‘foo’, to EOF.
<b>:g/^/m0</b>	For all lines move them to the beginning of the file = reverse file.
<b>:g/^foo/s/\$/bar</b>	For all lines starting with ‘foo’, substitute/append ‘bar’ to EOL.

Case switching	
<b>~</b>	Toggle case of character under cursor
<b>g~iw</b>	Toggle case of the inner word
<b>g~\$</b>	Toggle case from cursor to EOL
<b>gUw</b>	Convert rest of word to large caps
<b>gUU</b>	Convert current line to upper case
<b>gu0</b>	Convert from BOL to cursor to small caps.

Vimtex	
<b>\ll</b>	Start/stop continuous compiling mode
<b>\lv</b>	Jump to PDF viewer at the same point of the cursor
<b>\lc</b>	Clean auxiliary files
<b>[ [</b>	Jump to the previous (sub)section
<b>] ]</b>	Jump to the next (sub)section
<b>%</b>	Move between matching delimiters

Ctags & TagBar	
<b>:tag name</b>	Jump to (1st) match of the tag. Name can be a regular expression (e.g. /Detect*).
<b>:stag name</b>	Jump to (1st) match of the tag in a new window. Name can be a regular expression (e.g. /Detect*).
<b>:ts name</b>	List all tag matches, and lets you select one. If name is absent, use list of last :tag command.
<b>^]</b>	Jump to tag under the cursor
<b>^w}</b>	Show tag match in sep window
<b>g]</b>	List all matches of tag under cursor
<b>^o</b>	Go back one jump
<b>:Tagbar</b>	Toggle show/hide tagbar
<b>:!ctags -R</b>	Generate tags file, recursively

Windows			
<b>:split</b>	<b>:vsplit</b>	<b>:only</b>	<b>:close</b>
split top-bottom	split left-right	keep only current window	close current

Indenting					
<b>&gt;&gt;</b>	<b>&lt;&lt;</b>	<b>&gt;</b>	<b>&lt;</b>	<b>&gt;ap</b>	<b>&gt;aB</b>
indent current line	un-indent current line	indent selected lines	un-indent selected lines	indent current parag.	indent current block

CSV	
<b>:WhatColumn</b>	Show current column nr
<b>:nrColumns</b>	Show total numbers of columns
<b>:DeleteColumn 2</b>	Delete column nr 2
<b>:MoveColumn 2 4</b>	Move col 2 to right next to col 4
<b>:DupColumn</b>	Duplicate current column
<b>:Header 2</b>	Open separate fixed window with first 2 header lines. Close with :Header!
<b>:%ArrangeColumn!</b>	Prettify the columns by aligning them
<b>:NewDelimiter ;</b>	Change the delimiter to ;
<b>:1,10Sort 3</b>	Sort lines 1-10 of column 3
<b>:SearchInColumn 1 /foo/</b>	Search for patter foo in column 1
<b>:%Substitute 1,4/foo/bar/gc</b>	Subst in cols 1-4 foo with bar, globally + ask for confirmation

Fugitive	
<b>:Gblame</b>	Show git blame info for an indexed file.
<b>:Gedit SHA</b>	Show info for blob, tree, commits, and tags
<b>:Gedit master:python/transits.py</b>	Read-only!
<b>:Gdiff</b>	Open 3-way diff of current file, to solve conflicts
E.g. if you have conflicts after:  (feature) \$ git checkout master (master) \$ git merge feature  Left window: version of the master branch Right window: version of the feature branch Center window: current working copy  Keep your cursor in the center window (working copy).  :diffget //2 include left version in working copy :diffget //3 include right version in working copy :diffupdate update the diff highlighting	
<b>:Glog -- %</b>	Show the git log of this file per commit, and let's you select a commit
<b>:Gstatus</b>	Show the git index

gv	
<b>:GV</b>	Open commit browser. Browse, press enter to get more info on the commit. In <b>visual mode</b> : show commit info on the selected lines.
<b>:GV!</b>	Only show commit info of the current file
<b>:q</b>	Quit the commit browser

Easymotion	
<b>\\w</b>	Trigger word motion, then select appropriate letter
<b>\\s char</b>	Trigger character motion, then select letter.

Folds			
<b>zo/zc</b>	<b>zO/zC</b>	<b>zm/zr</b>	<b>zj/zk</b>
Open/close 1 fold under cursor	Open/close all folds under cursor	Increase /reduce fold level	Go to start of next/previous fold

Clipboard	
<b>:w !pbcopy</b>	Write selected lines to clipboard (visual mode)
<b>:r !pbpaste</b>	Insert contents of clipboard at current position

# Miscellaneous

<b>f</b> <i>char</i>	Jump to the first occurrence of char on this line
<b>J</b>	Join this line with the next one
<b>:set scrollbind</b>	Scroll synchronously. To be set in all windows that need to be in-sync. Add ! to command to undo.
<b>:set listchars=eol:¬,tab:&gt;.,trail:~,extends:&gt;,precedes:&lt;,space:␣</b>	
<b>:set list</b>	Show all the invisible characters
<b>:set nolist</b>	Stop showing all the invisible characters

# Outliner (.otl files)

<b>\cb</b> Insert checkbox	<b>\cx</b> Toggle checkbox	<b>\cd</b> Delete checkbox	<b>\n</b> Set fold level to <i>n</i>	<b>\-</b> Draw dashed line
<b>\s</b> Sort sub-tree under cursor ascending	<b>\S</b> Sort sub-tree under cursor descending	<b>\t</b> Append time stamp HH:MM:SS	<b>\d</b> Append date stamp YYYY:MM:DD	