

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  
[[ previous section / f()  
{ previous blank line  
k up 1 line  
j down 1 line  
} next blank line  
]] next section / f()  
^d down 1/2 page  
^f forward 1 page  
G end of file  
8G to first non-blank char of line 8  
:8 start of line 8

## Buffers

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

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

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

## Inserting

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

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

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

## Deleting

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

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

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

## Substitute

: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/\<foo\>/bar/g  
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 char select line select block reselect

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

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

fa /foo 25G ii  
select select select until select inner  
until 1st 'a' until 'foo' line 25 indentation

a", a', a(, a[, a{, at  
i", i', i(, i[, i{, it

Version with a: include "" " 0 [] {} <>  
Version with i: exclude "" " 0 [] {} <>

## Copying

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

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

## Undo/Redo

u undo :earlier 4m  
^r redo :later 50s

:UndotreeToggle

## Search

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

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

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

n N  
Go to next match Go to prev match

/\<foo\> :noh  
Search forward for no search  
foo, exact highlights

## Move

:m n :2,4m \$ :m '}-1  
mv this line mv lines 2,3,4 to mv this line to  
to after line n after last line end of paragr.

:m \$ :2,4m 9 :.,.+2m 9  
mv this line mv lines 2,3,4 mv 3 lines from  
to after to after this line to  
last line line 9 after line 9

## Search & Replace

/foo Search & highlight all 'foo', go to 1st match

cgn Change current search pattern (gn)

<esc> Get out of edit mode

. Repeat for next match

n / N Go to next/prev match if need to skip one

## 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.
<b>:g/foo/#</b>	Show with line nrs. Then go to a line.

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. <i>/Detect*</i> ).
<b>:stag name</b>	Jump to (1st) match of the tag in a new window. Name can be a regular expression (e.g. <i>/Detect*</i> ).
<b>:ts name</b>	List all tag matches, and lets you select one. If name is absent, use list of last <b>:tag</b> 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 <b>:Header!</b>
<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. Forward.
<b>\\b</b>	Idem but backward
<b>\\s char</b>	Trigger character motion, then select letter.

Markers	
<b>m a</b>	Set marker named ‘a’ under cursor
<b>m a!</b>	Same but override previous marker ‘a’
<b>’a</b>	Go to start of line containing marker ‘a’
<b>`a</b>	Jump to marker ‘a’ (back quote)
<b>:marks</b>	Show a list of markers
<b>:delm ab / !</b>	Delete marker ‘a’ and ‘b’ / all markers
<b>:c `a</b>	Change from cursor to marker ‘a’
<b>:]’ / [’</b>	Jump to next/prev line with marker
<b>`.</b>	Special marker: last change in buffer

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

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

Macros			
<b>  qa</b>	Start recording to register ‘a’	<b>@a</b>	Execute macro
<b>  ...</b>	Fancy commands	<b>5@a</b>	Execute 5 times
<b>  q</b>	Stop recording	<b>:reg</b>	List all macros

Special Text Objects			
<b>gn</b>	Current search pattern	<b>i i</b>	Inner indentation level, no line above
<b>aa</b>	Function arg under cursor, with comma	<b>a i</b>	Indentation level + one line above ( <i>Python</i> )
<b>ia</b>	Function arg under cursor, without comma ( <i>inner</i> )	<b>a I</b>	Indentation level + line above + line below ( <i>C++</i> )
Example in C++ : caI : change current indentation level incl. { } above/below			

Miscellaneous	
<b>^a / ^x</b>	Incr/decr the nr under cursor. Useful to automate.
<b>f / F char</b>	Jump to next/prev occurrence of <i>char</i> 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>:ab asap as soon as possible</b>	Replace abbreviation on the fly
<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