

Substitute

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

:s/foo/bar/q

:%s/foo/bar/q

In all lines replace all 'foo' with 'bar'

:%s/foo/bar/gic

In all lines replace all 'foo' with 'bar', caseinsensitive, and ask for confirmation

:3,9s/foo/bar/q

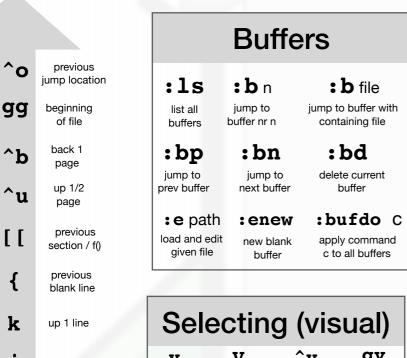
:.,.+5/^/

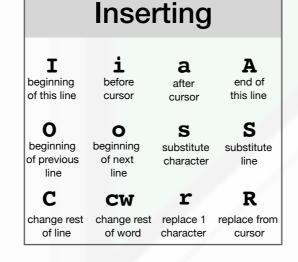
In lines 3 to 9 (incl) replace all 'foo' with 'bar'

From current line to 5 line further, prepend 3

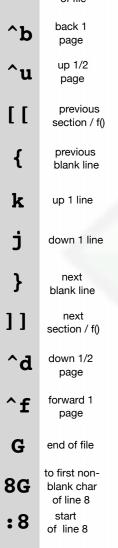
spaces at the beginning of each line

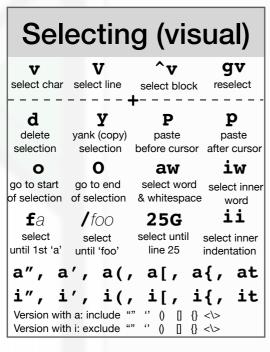
/q







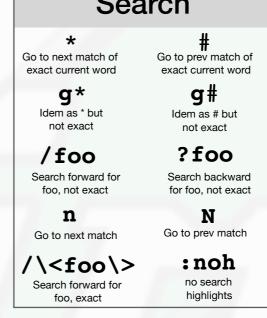




Copying				
УУ	yiw	уw		
yank this line	yank inner word	yank rest of word		
y0	у^	y\$		
yank from	yank from 1st	yank from		
BOL to	non-blank	cursor		
cursor	char to cursor	to EOL		
Undo/Redo				

:UndotreeToggle

u undo



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

Ocaron			
* next match of current word	Go to prev match of exact current word	:s/\ <foo\>/bar/g In current line replace all occurrences of 'foo' that are a separate word, with 'bar'</foo\>	
g* em as * but lot exact	g# Idem as # but not exact	:5,\$s/foo//g From current line to last line, replace all 'foo' with nothing (= delete 'foo')	
foo ch forward for o, not exact	?foo Search backward for foo, not exact	:%s/\t/ /g In all lines, replace tab with 4 spaces :%s/red\ pink/blue/g	
n to next match < foo \ > ch forward for foo, exact	N Go to prev match no search highlights	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.	

\mathbf{F}_{C} jump to jump to matching prev occur bracket of c

Move :2,4m \$:m '}-1 mv this line my lines 2,3,4 to mv this line to to after line n end of paragr. after last line :2,4m 9 :.,.+2m 9:m \$ 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) Get out of edit mode <esc> Repeat for next match Go to next/prev match if need to skip one

:earlier 4m

:later 50s



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

netrw & Ctrl-p

:edit|read|write sftp://user@hostname/path

:edit folder (folder can be.)

:Vexplore split left-right and show file browser

toggle between file browser view options/pathmove cursor to nearest match in file browser

P filename Fuzzy search file. Use ^f and ^b to toggle between files, buffers, and **m**ost **r**ecently

used. Can be used for "Open Recent..."

Global command

:[range]g/pattern/command

:q/foo/d Delete all lines that contain 'foo'

Move all lines from cursor to EOF, that do not contain 'foo', to EOF.

For all lines move them to the beginning of the file = reverse file.

:g/^foo/s/\$/bar For all lines starting with 'foo', substitute/append 'bar' to EOL.

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

Case switching

Toggle case of character under cursor

g~iw Toggle case of the inner word

g~\$ Toggle case from cursor to EOL

gUw Convert rest of word to large caps

guu Convert current line to upper case

gu0 Convert from BOL to cursor to small caps.

Vimtex

\11 Start/stop continuous compiling mode

 \mathbf{V} Jump to PDF viewer at the same point of the cursor

\1c Clean auxiliary files

[Jump to the previous (sub)section

]] Jump to the next (sub)section

% Move between matching delimiters

Tags and YCM

: tag name Jump to (1st) match of the tag. :stag shows it in a new window. Name can be a regex (e.g. /Detect*).

:ts name List all tag matches, and lets you select one. If name is absent, use list of last :tag command.

Jump to tag List all matches of Go back In insert mode: show under cursor tag under cursor one jump window with f() def.

:Tagbar Toggle show/hide tagbar

Double-click or press enter on a tag to jump

• Press space on a function to display arg list.

:!ctags -R Generate tags file, recursively, in current dir

Windows

:split :vsplit :only :close

split topbottom

split left-right

keep only current window

close current

Indenting

CSV

:WhatColumn Show current column nr

:nrColumns Show total numbers of columns

:DeleteColumn 2 Delete column nr 2

:MoveColumn 2 4 Move col 2 to right next to col 4

:DupColumn Duplicate current column

: Header 2 Open separate fixed window with first 2 header lines. Close with :Header!

:%ArrangeColumn! Prettify the columns by aligning them

:NewDelimiter; Change the delimiter to; :1,10Sort 3 Sort lines 1-10 of column 3

:SearchInColumn 1 /foo/

Search for patter foo in column 1

:%Substitute 1,4/foo/bar/gc

Subst in cols 1-4 foo with bar, globally + ask for confirmation

Fugitive

:Gblame Show git blame info for an indexed file.

:Gedit SHA Show info for blob, tree, commits, and tags

:Gedit master:python/transits.py Read-only!

:Glog -- % Show the git log of current file per commit, and lets you select a commit. Press enter on parent commits, to go further back.

:Gstatus Show the git index

:Gdiff Shows differences between working copy of the current file and the index, in a split window. **:q** to exit.

In case of merge conflicts: open 3-way differences.

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

gv

: GV Open commit browser. Browse, press enter to get more info on the commit. In *visual mode*: show commit info on the selected lines.

: GV! Only show commit info of the current file

: q Quit the commit browser

Special Text Objects

gn Current search pattern

ii Inner indentation level, no line above

aa Function <u>arg</u> under cursor, with comma

a<u>i</u> Indentation level + one line above (*Python*)

ia Function **a**rg under cursor, without comma (*inner*)

a<u>I</u> Indentation level + line above + line below (C++)

Example in C++ : caI : change current indentation level incl. { } above/below

Macros

qa Start recording to register 'a'Ga Execute macroFancy commands5@a Execute 5 times

q Stop recording

:req List all macros

Outliner (.otl files)

\cb\cx\cd\n\-Insert
checkboxToggle
checkboxDelete
checkboxSet fold
level to nDraw
dashed line

\s \S \t \dSort sub-tree Sort sub-tree Append time Append date

under cursor under cursor stamp stamp
ascending descending HH:MM:SS YYYY:MM:DD

Folds

ZO/ZCOpen/close
Open/close all
fold under
Open/close all
folds under

cursor

cursor

zO/zC zm/zr
Open/close all folds under Increase /reduce fold level

zj/zk
Go to start of next/previous fold

Clipboard

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

(Un)commenting

 ^▼
 Block select

 Select
 Select the block

 I
 Insert mode at beginning

 //
 Write comment symbols

<esc> Exit & apply to all lines

 ^▼
 Block select

 Select
 Select comment symbols

 d
 delete

Markers

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

Miscellaneous

^a / ^x Incr/decr the nr under cursor.

J Join this line with the next one

set scrollbind Scroll synchronously. Set in all windows that need to be in-sync. Add! to to undo.

ab fe for example Replace abbreviation on the fly

set list Show all the invisible characters

set nolist Stop showing all the invisible characters

After startup: load last edited file

^n In insert mode: word completion
• Repeat last change in normal mode

Repeating

Repeat search - forward ; Rpt last f c - forward

Repeat search - backward $^{\prime\prime}$ Rpt last f $_{c}$ - backward

Rpt last change in Normal mode

Easymotion

\\w Word motion, then select appropriate letter. Forward.

\\b Idem but backward

\\s char Trigger character motion, then select letter.