| \n | toggle relative number | | | | |
|----------------------|--|---|--|----------------|---|
| \d | dispatch and make current file | | | | |
| \w | toggle word wrap | | | | |
| F2 | YcmCompleter GoTo | 36.14.1 | | tmux | |
| F3 | YcmCompleter GoToDeclaration | Multiple cursors | | | enter text mode, hjkl to move, /? to search, C-ud |
| F4 | YcmCompleter GoToDefinition | C-n in Normal | highlights the current word under the cursor in Visual mode/finds next | C-a [| to page |
| F5 | open nerd tree | C-n in Visual | virtual cursor at every line and leaves you in Normal mode. | C-a: | enter command mode |
| F8 | toggle ctag bar | C-p in Visual | remove the current virtual cursor and go back to the previous virtual cursor location. | C-a c | create window |
| tcomment | | | remove the current virtual cursor and skip to the next virtual cursor | C-a. | rename window |
| gc{motion} | Toggle inline comments | C-x in Visual | location. | C-a - | split horizontal |
| gc[Count]c{motion} | Toggle comment with count and motion | : | | C-a | split vertical |
| gcc | Toggle comment for the current line | {r}MultipleCursorsFine | d regex | C-a 19 | jump to window |
| g<{motion} | Uncomment region | Ctrlp | | Buffer | |
| g | Uncomment the current line | С-р | CtrlP file | C-w ghjk | move to the left/bottom/top/right window |
| g | Uncomment the current region as block | \b | CtrlP buffer | C-w W | move to next window |
| g>{motion} | Comment region | \m | CtrlP mixed | :sp | horizontal split |
| g>c | Comment the current line | \r | CtrlP most recently used | :vsp | vertical split |
| g>b | Comment the current region as block | \t | CtrlP buffer tags | :b | open buffer with filename |
| gc | Toggle comments (visual mode) | F5 | (in CtrlP) purge the cache | :b | open buffer with number |
| g> | Comment selected text (visual mode) | C-f/C-b | (in CtrlP) cycle between modes | :bd | close current buffer |
| C Tags | Comment selected text (visual mode) | C-j/C-k | (in CtrlP) up/down | :bn | next buffer |
| _ | iump to tog | C-d | (in CtrlP) switch to filename only search | :bp | prev buffer |
| C-] C-t | jump to tag | C-r | (in CtrlP) switch to regexp mode | C-6 | last buffer |
| | ctags jump back from tag | Format | (iii thiii) thiill to the grap in the | :enew | new buffer, with filename creates |
| gd -D | goto local declaration | \fx | format XML | :e | reload buffer, with filename creates |
| gD | goto global declaration | \fj | format JSON | :bufdo e | reload buffer all buffers, runs foreach buffer |
| g] | list matching tags | :%!xmllintformat - | format xml in whole file | A | reload burier an buriers, runs foreach burier |
| gC-] | jump to tag if defined once otherwise list matching tags | :{range}!xmllint | Tormat XIII III WHOLE THE | \a | switch to the header/cpp file |
| :tags | list tag stack | format - | format xml in selection | | ** |
| \ut | update ctags | :%!python -m | | \as | splits and switches |
| Cscope | | json.tool | format json in selection | \av \at | vertical splits and switches new tab and switches |
| C-\ s | show uses of symbol in quickfix | | | | |
| Find | | v | enter Visual Mode | \an | cycles through next match |
| \gf (Normal) | (global find) start empty Ag for cpp | C-v | enter Visual Mode | Easy Motion | |
| \gf (Visual) | (global find) Ag with selection for cpp | C-v | in insert mode adds tab char | \\w | easy motion word forward |
| \f (Normal) | (find)start empty Ag in current file | retab | fix tabs to spaces | \\b | easy motion word back |
| \f (Visual) | (find) Ag with selection in current file | :so ~/.vimrc | Reload vimre | \\k | easy motion up line |
| Replace | | .so ~.√.\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\ | run external command on current file | \\j | easy motion down line |
| rx | replace all chars in selection with x | | | Errors | easy motion down time |
| :{range}sort u | sort and remove dups | :noh | turn off search highlight | | tangariana nyay arrag |
| :{range}sort | sort | vim -d file1 file2 | diff 2 files | [q | :cprevious prev error |
| :{range}sort! | sort reverse | % | in {range} will use whole file |]q | :cnext next error |
| :{range}sort n | numeric sort | :set pastetoggle :set paste | turn off paste formatting | [Q | cfirst first line |
| :s/foo/bar/g | Change each 'foo' to 'bar' in the current line. | set paste | tan on paste formatting |]Q | :clast last line |
| :%s/foo/bar/gc | Change each 'foo' to 'bar' in all the lines., but ask for confirmation first. | | | :copen | open Quickfix |
| :{range}s/foo/bar/gc | Change each 'foo' to 'bar' in all lines within a visual selection, but ask for confirmation first. | | | | |
| | | | | | |

 $:!ctags -R --c++-kinds=+p --fields=+iaSl --extra=+q --links=yes \; .$

:g/pattern/d

remove every line matching pattern

update tag file

:! find . -iname *.c - iname *.c - iname *.h - iname *.pp - .cscopelist.tmp; cscope - q - R - b - i .cscopelist.tmp; rm .cscopelist.tmp update csope db - cscopelist.tmp; rm .cscopelist.tmp update csope db - cscopelist.tmp; rm .cscopelist.tmp; r