

Vim Cheat Sheet for Programmers (screen)

HOW-TO make Vim not suck Out of the Box: :help statusline :set nocompatible ruler laststatus=2 showcmd showmode number Search :set incsearch ignorecase smartcase hlsearch Remove useless splash screen :set shortmess+=I

Best tips: <http://vim.wikia.com/> Best scripts: <http://www.vim.org/scripts/index.php>

~ toggle case	! extern filter	@ play macro	# prev identifier	\$ →	% goto match	^ soft ↑	& repeat :s	* next identifier	(begin sentence) end sentence	_ cur line	+ ↓
· goto mark	1	2	3	4	5	6	7	8	9	0 hard ↑	- ↑	= auto-format
	14 block select	^w window...	scroll line ↑	12 :redo	10 ctags return	scroll line ↓	half page	-Ctrl++	prev mark	9 ↑	Normal	ctags identifier
Tab	Q ex mode	W WORD ↘	E end WORD ↘	R Replace	T ← till char	Y copy line	U undo line	I insert ←	O open ↑	P paste ↑	{ paragraph	} paragraph
	q record macro	w word ↘	e end word ↘	r replace char	t till char →	y copy	u undo	i ← insert	o open ↓	p paste ↓	[misc.] misc.
	7 incr. #		10 half page ↓	page ↓	file/cursor info	-Ctrl++	15 Ctrl-J		redraw	-Ctrl;	-Ctrl'	-Ctrl+
Caps	A append →	S subst line	D del →	F ← find char	G goto eof / goto line#	H Top screen	J Join lines	K man page identifier	L Bottom screen	: Ex cmd line	" register	goto col#
	a append →	s subst char	d del	f find char →	g extra	h ←	j ↓	k ↑	l →	; "next" t/F/T	' goto mark	·
Ctrl ^	:suspend	7,11 decr. #	Normal / Cancel	block select	page ↑	9,16 ↓	15 -Ctrl-M	-Ctrl-	-Ctrl-	-Ctrl-	-Ctrl/	
Shift ↑	Z quit	X ← del char	C change →	V select lines	B \ WORD	N "prev" find	M Middle screen	< undent	> indent	? find ↘		
	z extra	x del char →	c change	v select chars	b \ word	n find "next"	m set mark	, "prev" t/F/T	· repeat cmd	/ find ↘		

Legend: 16 The search direction is relative; next is the initial direction, previous is the opposite direction. ~, repeat same initial direction find. N, repeat opposite initial direction find. Note: ; only searches cursor line, n N searches buffer.

Macro Register name (0-9a-zA-Z) required Motion req.; act between cursor & dst	Op Command	Cmd aAsScCRiLo enter insert mode	Move Moves cursor or defines range for op	Find Search (\ = reverse, \ = forward)	tag ctags / diffs / folding	Code Code formatting, whitespace, etc.	Extra Extended functionality; req. extra chars Char arg req. g z Z ^w ' " ...	Modes :help modes	word Foo (src , dst , len);	WORD Foo (src , dst , len);	Startup vim <filename> +123 vim <file> ... -t Foo vim <file> ... -c "/Foo"	GUI vim -g or gvim Linux :set guifont=ProggyTinyTt\ 12 OSX :set guifont=ProggyTiny:h11 diff gvimdiff <file1> <file2> [<file3>]	Broken Keys Ctrl-I = Tab, Ctrl-J = ESC Vim is still unable to map certain keys for your own use...	Ex cmd line :help cmdline :w save :q quit :q! quit w/o save :e <filename> :source % exec cmds in cur file :exec '...' do cmd GUI:gui switch to GUI ver. Move :help movement	Tags :help tags :ts list active tags ^] jump to tag under cursor ^t restore cursor before tag jump :pa complete word :ta Foo manual jump to tag 'Foo'	Diffs :help diff :diffupdate resync [c prev diff]c next diff :hi DiffAdd guifg=#rrggbb :hi DiffChange guibg=#rrggbb :hi DiffDelete :hi DiffText gui=none default bold	Folding :help folding zR fold remove zi invert all zo fold open zr fold reduce zc fold close zm fold more	Syntax :help syntax :set filetype=... c cpp sh make perl python :syntax enable	Changes :help changes Auto-complete :changes g; older change g; newer	Macros :help recording q* start recording q stop recording @* playback @@ repeat	Code :set tabstop=# :set expandtab! :set listchars=... :set list! :set colorcolumn=80 noremap + :s/^/\\<CR>	Insert mode 9 ^p prev ^n next 10 ^d undent ^t indent 11 ^x ^f filename completion ^s spelling ^s spell! ^k dictionary s next bad ^t thesaurus :help spell 12 ^r* paste register 0-9a-zA-Z or ... + clipboard (or '*') " last del/copy % filename	Unused & Duplicate keys \ Ctrl-K Ctrl-S (free) Ctrl-L (redraw) 13 ^ near dup of ' 14 Ctrl-Q = Ctrl-V 15 Ctrl-J = Ctrl-M = ^N	buffer # :buffers :new blank file/buffer :bn next file :bp prev file :bd close file :bd! force close :set lines=# :set columns=# GUI :winpos # # Windows :help windows ^w* or :wincmd *	Cursor Bookmarks :marks ma mark local 'a' 'A goto global 'A' '' prev location File / Directory :Explore Or :e . :set busedir=... buffer last
---	----------------------	--	---	--	---------------------------------------	--	--	-----------------------------	---	---	--	---	---	--	--	--	--	--	---	--	--	---	--	--	---