






Update: Version 2.0 is up! There are now 4 versions to chose from:

(If you are wondering what the differences are between the screen and print: the screen has less color variations, and no gradients to make it easier to read.)

- One designed for on-screen reading: [PDF \(Excel 2011 source\)](#),
- One designed for hard-copy reading: [PDF \(Excel 2011 source\)](#),
- One designed for monochrome printing: [PDF \(Excel 2011 source\)](#), and
- One designed for Red/Green color blindness -- a "Blue" theme [PDF \(Excel 2011 source\)](#).

The color coded Legend / Keys helps provide guidance for your experience level.

-  = Essential
-  = Basic
-  /  = Advanced
-  = Expert

If you don't like this version (I'd love to hear why) try these, you may find them more to your liking...

- http://www.viemu.com/a_vi_vim_graphical_cheat_sheet_tutorial.html
- <http://tnerual.eriogerg.free.fr/vim.html>
- <http://www.lagmonster.org/docs/vi.html>
- <http://jrmiii.com/2009/03/06/learning-vim-the-pragmatic-way.html>

Thanks to all the redditors for the feedback! ~ Michael

The on-screen 150 DPI .png version is shown...

Esc Normal	Revision 2.0 Sept. 11, 2011	Vim 7.3+ :version	Vim Cheat Sheet for Programmers										Copyright © 2011 May be freely distributed! Sharing is Caring.	http://michael.PeopleOfHonorOnly.com/vim/	
HOW-TO make Vim not suck out of the Box: <code>:help statusline</code> <code>:set nocompatible ruler laststatus=2 showcmd showmode number</code>															
Best tips: http://vim.wikia.com/				Best scripts: http://www.vim.org/scripts/index.php				Search <code>:set incsearch ignorecase smartcase hsearch</code>				Remove useless splash screen <code>:set shortmess+=!</code>			
Ctrl-~		Ctrl-1	Ctrl-@	Ctrl-#	Ctrl-\$	Ctrl-%	Ctrl-^	Ctrl-&	Ctrl-*	Ctrl-(Ctrl-)	Ctrl-_-	Ctrl-+	Ctrl-~	Ctrl-~
toggle case		extern filter	play macro	prev identifier	goto match	soft ---	repeat :s	next identifier	begin sentence	end sentence	cur line	asciiformat	asciiformat	asciiformat	asciiformat
goto mark		1	2	3	4	5	6	7	8	9	0	hard ---	asciiformat	asciiformat	asciiformat
14 block select		*w* window...		scroll line ↑		12 .redo	10 ctags return	scroll line ↓		half page		Ctrl-I	prev mark	9 ↑	Normal
Q ex mode		W WORD		E end WORD		R Replace	T ← until char	Y copy line		U undo line		I insert ---	O open ↑	P paste ↑	{ paragraph
q record macro		W word		e end word		r replace char	t until char →	y copy		u undo		i insert	o open ↓	p paste ↓	[misc.
Tab															
7 incr. #				10 half page ↓		page ↓	file/cursor info	Ctrl-H	Ctrl-J	redraw		Ctrl-;	Ctrl-;	Ctrl-;	Ctrl-;
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	: cmd line	register	goto col#	goto col#
a append ↓		s subst char		d del		f ← find char →	g extra	h ←	j ↓	k ↑	l →	next	goto mark	goto mark	goto mark
Caps															
Ctrl-^		7.11 decr. #		Normal / Cancel		block select		page ↑		9.16 ↓	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	
Shift-z		extra		x del char →		c change		v select chars		b \ word		n find "next"		m set mark	
Unused & Duplicate keys \ Ctrl-K Ctrl-S (free) Ctrl-L (redraw) 13 near dup of o 14 Ctrl-Q = Ctrl-V 15 Ctrl-J = Ctrl-M = "N"															
Legend: The search direction is relative; next is the initial direction, previous is the opposite direction. : repeat same initial direction find. : repeat opposite initial direction find. Note: : only searches cursor line, : searches buffer.															
Macro Register name (0-9a-zA-Z) required Op Motion req.; act between cursor & dst Cmd Command Ins Command and 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 n Normal Esc ^ [^ c i Insert a i r s v Visual v V ^ v ^ q o Op pending c d y < > c Command Line : / ? !															
word Foo ([src] b dst y b len) ; WORD Foo (src) b dst y b len) ;															
Startup vim <filename> +123 goto line 123 vim <file> - -t Foo edit at tag 'Foo' vim <file> - -c "/Foo" cmd: find 'Foo' & edit vim -g or gvim start GUI ver. Linux :set guifont=ProgygyTinyT12 OSX :set guifont=ProgygyTinyT12 gvimdiff <file> <file2> <file3>															
Broken Keys Ctrl-I = Tab, Ctrl-I = ESC Vim is still unable to map certain keys for your own use... Caps, Ctrl-1, Ctrl-Shift-1, Ctrl-I, Ctrl-1, etc.															
Code = < > << >> :set backspace=indent,eol,start allow backspace join lines :set shiftwidth= indent width for ai toggle auto-indent :set lisp lisp indent mode															
:set tabstop= expandtab :set expandtab= listchars=... :set list= listchars=... :set colorcolumn= visible right margin indicator noremap + :set /<\\</<CR> block comment															
:set tabstop= expandtab :set expandtab= listchars=... :set list= listchars=... :set colorcolumn= visible right margin indicator noremap + :set /<\\</<CR> block comment															
set tab stop every nth col tab stop hard/soft tabs tab> :trail,~nbp,%:eol\$ toggle whitespace Visible right margin indicator block comment															
:set numbers= toggle line numbers :set wrap= toggle linewrap display :set showmatch= highlight matching () :set spell= set spell :set dictionary= set next bad :set thesaurus= help spell 12 paste register 0-9a-zA-Z or ... + clipboard (or *) help <C-TRL-... last del/copy filename															
File / Directory :Explore or :e :set browser= one of buffer last															

[illegible]

The monochrome hard-copy 300 DPI .png version:

Esc

Normal

Revision 2.0

Sept. 11, 2011

Vim 7.3+

:version

Vim Cheat Sheet for Programmers

Copyright © 2011
May be freely distributed!
Sharing is Caring.

http://michael.PeopleOfHonorOnly.com/vim/

HOW-TO make Vim not suck Out of the Box: :help statusline :set nocompatible ruler laststatus=2 showcmd showmode number

Best tips: <http://vim.wikia.com/> Best scripts: <http://www.vim.org/scripts/index.php> Search :set incsearch ignorecase smartcase hsearch Remove useless splash screen :set shortmess+=!

Best tips: <http://vim.wikia.com/>

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

Search :set incsearch ignorecase smartcase hsearch

Remove useless splash screen :set shortmess+=!

Ctrl-~ toggle case

Ctrl-! extern filter

Ctrl-@ play macro

Ctrl-# prev identifier

Ctrl-\$

Ctrl-→

Ctrl-% goto match

Ctrl-^ soft

Ctrl-& repeat *

Ctrl-* next identifier

Ctrl-(begin sentence

Ctrl-) end sentence

Ctrl-_ cur line

Ctrl-+ auto-format

1 goto mark

2

3

4

5

6

7

8

9

0 hard

↑

↓

=

14 block select

Ctrl-W window...

scroll line ↑

12 :redo

10 ctags return

scroll line ↓

half page ↑

Ctrl-I prev mark

9 ↑

Normal

ctags identifier

Q ex mode

W WORD ↘

E end WORD ↘

R Replace

T ← until 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 ← until char →

y copy

u undo

i ↘ insert

o open ↓

p paste ↓

[misc.

] misc.

Tab

7 incr. #

10 half page ↓

page ↓

file/cursor info

Ctrl-H

15 Ctrl-J

redraw

Ctrl-;

Ctrl-:

Ctrl-'

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 mark

goto col#

a append ↘

s subst char

d del

f find char →

g extra

h ←

j ↓

k ↑

l →

next

goto mark

goto col#

Caps

Ctrl-^ :suspend 7,11 decr. #

Normal / Cancel

block select

page ↑

9,16 Ctrl-M

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

Shift-Z extra

x del char →

c change

v select chars

b word

n find next

m set mark

prev

repeat cmd

find

Unused & Duplicate keys

Ctrl-K Ctrl-S (free)

13 Ctrl-L (redraw)

near dup of 1

14 Ctrl-Q = Ctrl-V

15 Ctrl-J = Ctrl-M = "N"

Legend:

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

Macro

Op

Cmd

Ins

Move

Find

tag

Code

Extra

Modes

n

i

v

o

c

word

WORD

Startup

Register name (0-9a-zA-Z) required

Motion req.; act between cursor & dst

Command

Command and enter insert mode

Moves cursor or defines range for op

Search (↖ = reverse, ↘ = forward)

ctags / diffs / folding

Code formatting, whitespace, etc.

Extended functionality; req. extra chars

Char arg req. g z Z w ' " ~ ...

Modes

n Normal

i Insert

v Visual

o Op pending

c Command Line

word Foo [src] bdst b len

Note: There is no whitespace in-between 'Foo[src]' but before/after 'dst'.

WORD Foo (src) bdst b len

Startup

vim <filename> +123

vim <file> - -t Foo

vim <file> - -c "/Foo"

vim -g or gvim

Linux

OSX

diff

vimdiff <file> <file> [<files>]

Broken Keys

Vim is still unable to map certain keys for your own use...

Caps, Ctrl-1, Ctrl-Shift-1, Ctrl-I, Ctrl-I, etc.

See: src/ops.c -o "/valid_yank_reg" for "reg. names

See: src/norm.c -o "/nv_cmds" for "q extra cmds

See: src/edit.c -o "/ctrl_x_map" for "x insert cmds

Code

set backspace=indent,eol,start

allow backspace join lines

set shiftwidth=#

set autoindent

set lisp

set tabstop=#

set expandtab

set listchars=

set list

set colorcolumn=80

noremap + :s/^/\\<CR>

set tabstop=#

set expandtab

set listchars=

set list

set colorcolumn=80

noremap + :s/^/\\<CR>

set tab stop every #th col

tab>:trail~:nbsp%:eol\$

toggle whitespace

visible right margin indicator

block comment

set numbers

set wrap

set showmatch

noremap + :s/^/\\<CR>

toggle line numbers

toggle linewrap display

highlight matching ()

set spell

set dictionary

set thesaurus

paste register 0-9a-zA-Z or ...

clipboard (<+*)

last del/copy filename

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

set spell

set dictionary

set thesaurus

Esc Normal
Revision 2.0
Sept 11, 2011

**vim 7.3+
:version**

Vim Cheat Sheet for Programmers

Copyright © 2011
May be freely distributed
http://michael.PeopleOthelloOnly.com/vim/

HOW to make Vim not suck out of the Box: `:help stasuline` `set nocompatible ruler laststatus=2 showcmd showmode number`

Search `set incsearch ignorecase smartcase hsearch` `.map <F9> e $HOME/ vimrc<CR>` `.map <F6> so $HOME/ vimrc<CR>`

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

Remove useless splash screen `set shortmess+=M`

Ctrl- toggle case		Ctrl- ! extern filter		Ctrl-@ play macro		# prev identifier		Ctrl-3 \$ →		Ctrl-4 % goto match		Ctrl-5 ^ soft ~		Ctrl-6 & repeat %		Ctrl-7 * next identifier		Ctrl-8 (begin sentence		Ctrl-9) end sentence		Ctrl-0 - cur line		Ctrl-1 + auto-format	
1		2		3		4		5		6		7		8		9		0		-		+			

14 Q ex mode		W window...		E end word		R Replace		T ← until char		Y copy line		U undo line		I insert →		O open ↑		P paste ↑		Normal		tags identifier	
q record macro		w word		e end word		r replace char		t until char →		y copy		u undo line		i insert		o open ↓		p paste ↓		{ paragraph		} paragraph	
7 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		g goto col#	
a append ↵		s subst char		d del		f find char →		g extra		h ←		j →		k ↑		l →		→		"next" f/ff/T		g goto mark	

Ctrl ^		:suspend		7.11 decr. #		Normal / Cancel		block select		page ↑		9.16 ↓		Ctrl M		Ctrl- redraw		Ctrl- Ex cmd line		Ctrl- " register		Ctrl- goto col#	
Shift #		Z quit		X ← del char		C change →		V select lines		B \ WORD		N "prev" find		M Middle screen		< undent		> indent		? find \		Unused & Duplicate keys	
		z extra		x del char →		c change		v select chars		b \ word		n find "next"		m set mark		"prev" f/ff/T		repeat cmd		/ find \		\ Ctrl-K Ctrl-S (free) 13 Ctrl-L (redraw) 14 Ctrl-Q = Ctrl-V 15 Ctrl-U Ctrl-N = M	

16 The search direction is relative, **next** is the initial direction, **previous** is the opposite direction. `n` repeat same initial direction find. `N` repeat opposite initial direction find. **Note:** `o` only searches cursor line, `O` searches buffer.

Legend:

- Macro** Register name (0-9a-zA-Z) required
- Op** Motion req; act between cursor & dst
- Command** Command
- Ins** Command and 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** Char arg req. `g z z w w l l ...`

Modes

- n** Normal `Esc` `^ [^ c`
- i** Insert `a i r s`
- v** Visual `V v v q`
- o** Op pending `c d y < >`
- c** Command Line `/ ? !`

word `Foo` `[src` `b dst` `j len` `] j`

WORD `Foo` `[src` `b dst` `j len` `] j`

Startup

```
vim <filename> +123      goto line 123
vim <file> -- -t Foo      edit at file 'Foo'
vim <file> -- -c 'Foo'    cmd: find 'Foo' & edit
vim -g or gvim           start GUI ver.
```

Linux `set guifont=ProgygyT1nTtL 12`

OSX `set guifont=ProgygyT1nTtL:h11`

diff `gvimdiff <file> <file> [<file>]`

Broken Keys `Ctrl-H` `Tab` `Ctrl-E` `ESC`

Vim is still unable to map certain keys for your own use...

See: `src/opa.c` or `"valid_yank_reg"` for `reg` names

See: `src/normal.c` or `"rv_cmds"` for `g` extra cmds

See: `src/oldedit.c` or `"ctrl_v_magic"` for `ctrl_v` insert cmds

<p>Ctrl <code>cmdline</code> <code>file</code> <code>insert file</code></p> <pre> :ls save :gs switch to :q quit :q! quit w/o save :e <file> :e! edit file in new buffer :source <file> :exec cmds in cur file :exec <file> :cmd cmd </pre> <p>help movement</p> <pre> < Start of Line 1st non-whitespace 0 End of Line column 0 < End of Line : move col 0 : move col # : page ↑ : page ↓ : scroll line ↑ : scroll line ↓ lg start of file og end of file #g goto line # o end of file </pre> <p>help range</p> <pre> s/Foo/Bar find Foo replace w/ Bar s/Foo/Bar/g all instances on line s/Foo/Bar/ apply to whole file : s,cur line, cur line + # lines \$ last line : s,cur line, cur line + # lines : s,cur line, cur line + # lines </pre> <p>Code <code><</code> <code>></code> <code><<</code> <code>>></code></p> <p><code>set backspace=indent,eol,start</code></p> <p>allow backspace join lines</p> <p><code>set shiftwidth=0</code> indent width for all</p> <p><code>set autoindent</code> toggle auto-indent</p> <p><code>set lisp</code> lisp indent mode</p>	<p>help tags</p> <pre> :ts list active tags :J jump to tag under cursor :R restore cursor before tag jump :P complete word :ta Foo manual jump to tag 'Foo' </pre> <p>help diff</p> <pre> < prev diff :h DiffAdd guifg=frggb j next diff :c DiffChange guifg=frggb :diffupdate :h DiffDelete guifg=frggb resync </pre> <p>help folding</p> <pre> zR fold remove zo fold open zc fold close ni invert all zf fold reduce zm fold more </pre> <p>help changes</p> <pre> :changes g? older change g? newer change </pre> <p>help syntax</p> <pre> :syntax enable :set filetype= c cpp sh make perl python Note: chose only ONE type! </pre> <p>help recording</p> <pre> q start recording @ playback q start recording @@ repeat </pre> <p>set tabstop=#</p> <p><code>set expandtab</code></p> <p><code>set listchars=</code></p> <p><code>set list</code></p> <p><code>set colorcolumn=80</code></p> <p><code>noremap < / s/\/\//<CR></code> block comment</p>	<p>map <code><Explore>CR</code> manually type <code><CR></code></p> <pre> \$0 before del/copy/paste to use register +X cut to system clipboard reg. +P paste from system clipboard 1 Number before any action repeats it 2p paste twice 3. repeat thrice 2 Repeat op to act on current line copy line dd del line << undent line >> indent line 3 # highlight words under cursor save & quit :X quit w/o save center cursor line in window scroll left zL scroll right zt scroll top zb scroll bottom top of file open file under cursor incr # under cursor (Dec / Hex) decr # under cursor (Dec / Hex) start a "new" search </pre> <p>Insert mode</p> <pre> 9 "p prev auto-complete "n next 10 "d indent \$11 "x filename completion " spelling :set spell! " dictionary :</pre>
---	--	--

Vim has 4 bugs...

- Othe then that, Vim is a beautiful code editor!