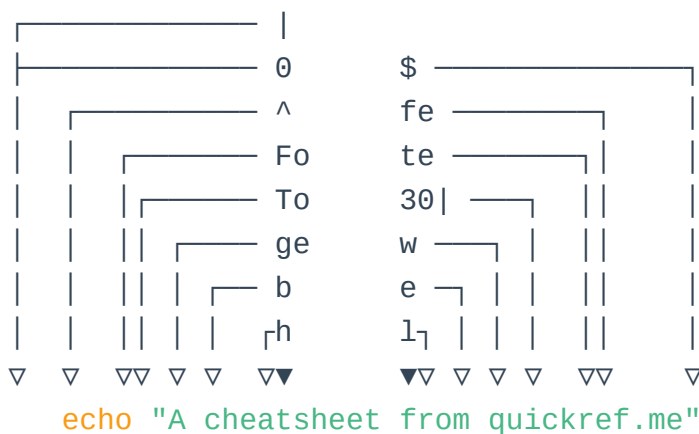# Vim

A useful collection of Vim 8.2 quick reference cheat sheets to help you learn vim editor faster.

Getting started
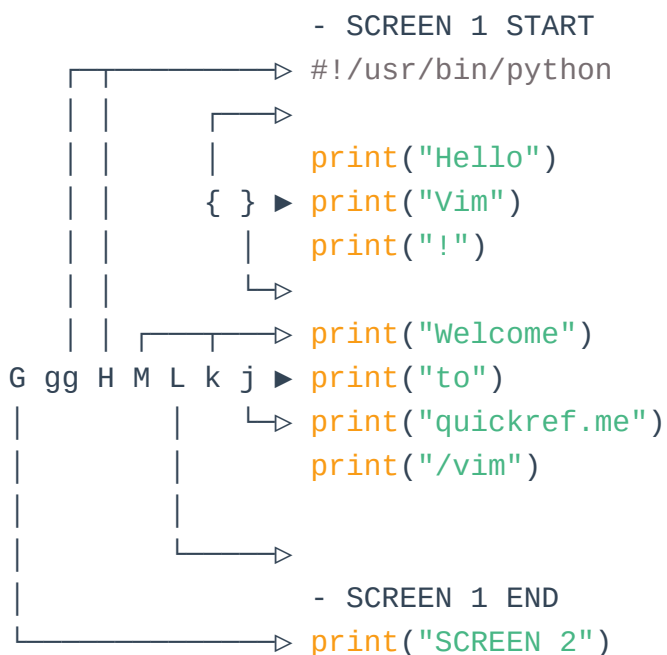
Motion Diagrams

▼/► Cursor   ▽/▷ Target

Left-right motions

```
 _____ |
|_____ 0    $ _____
| _____ ^   fe _____ |
| | _____ Fo   te _____ |    |
| | | ____ To   30| ____   ||   |
| | || ____ ge   w ____ |   ||   |
| | || | __ b    e ┐ | |   ||   |
| | || | | ┌h     l┐ | | |   ||   |
▽  ▽  ▽▽ ▽ ▽ ▽▼    ▼▽ ▽ ▽ ▽  ▽▽   ▽
    echo "A cheatsheet from quickref.me"
```

Up-down motions

```
                      - SCREEN 1 START
  _____▷ #!/usr/bin/python
 | |         ____▷
 | |        |        print("Hello")
 | |     { } ►  print("Vim")
 | |         |        print("!")
 | |         └__▷
 | |   _____▷ print("Welcome")
G gg H M L k j ►  print("to")
 |         |   └__▷ print("quickref.me")
 |         |        print("/vim")
 |         |
 |         └_____▷
 |                    - SCREEN 1 END
 └_____▷ print("SCREEN 2")
```

Motions

h | j | k | l	Arrow keys
`<C-u>` / `<C-d>`	Half-page up/down
`<C-b>` / `<C-f>`	Page up/down

Words

b / w	Previous/Next word
ge / e	Previous/Next end of word

Line

0 (zero) / $	Start/End of line
^	Start of line (non-blank)

Character

Fe / fe	Move to previous/next e
To / to	Move before/after previous/next o
| / n|	Go to first/nth column

Document

gg / G	First/Last line
:n | nG	Go to line n
} / {	Next/Previous empty line

Window

H / M / L	Top/Middle/Bottom screen
zt / zz / zb	Top/Center/Bottom this line

Insert Mode

i / a	Insert before/after cursor
I / A	Insert start/end of line
o / O (letter)	Insert new line below/above
s / S	Delete char/line and insert
C	Change to end of line
gi	Insert at last insert point

`gi` — Insert at last insert point

`Esc` | `<C-[>` — Exit insert mode

`:w` — Save

`:q` — Close file

`:wq` | `:x` | `ZZ` — Save and quit

`:wqa` — Save and quit all files

`:q!` | `ZQ` — Force quit

`:qa` — Close all files

`:qa!` — Force quit all files

`:w` new.txt — Write to `new.txt`

`:sav` new.txt — Save and edit `new.txt`

`:w` !sudo tee % — Write to readonly file

Normal Mode

`r` — Replace one character

`R` — Enter Replace mode

`u` / `3u` — Undo changes 1 / 3 times

`U` — Undo changes on one line

`J` — Join with next line

`<C-r>` /5 `<C-r>` — Redo changes 1 / 5 times

Cut and paste

`x` — Delete character (Cut)

`p` / `P` — Paste after/before

`xp` — Swap two characters

`D` — Delete to end of line (Cut)

`dw` — Delete word (Cut)

`dd` — Delete line (Cut)

`ddp`	Swap two lines
`yy`	Yank line (Copy)
`"*p` | `"+p`	Paste from system clipboard
`"*y` | `"+y`	Paste to system clipboard
In visual mode	
`d` | `x`	Delete selection (Cut)
`s`	Replace selection
`y`	Yank selection (Copy)

`.`	Repeat last command
`;`	Repeat latest `f`, `t`, `F` or `T`
`,`	Repeat latest `f`, `t`, `F` or `T` reversed
`&`	Repeat last `:s`
`@:`	Repeat a command-line command

Visual mode

`v`	Enter visual mode
`V`	Enter visual line mode
`<C-v>`	Enter visual block mode
`ggVG`	Select all text
`>` / `<`	Shift text right/left

Macros

`qi`	Record macro `i`
`q`	Stop recording macro
`@i`	Run macro `i`
`7@i`	Run macro `i` 7 times
`@@`	Repeat last macro

Operators

d	w
Operator	Motion

Combine operators with motions to use them

Available Operators

d	Delete
y	Yank (copy)
c	Change (delete then insert)
p	Paste
=	Formats code
g~	Toggle case
gU	Uppercase
gu	Lowercase
>	Indent right
<	Indent left
!	Filter through external program

Examples

Combination	Description
dd	Delete current line
dj	Delete two lines
dw	Delete to next word
db	Delete to beginning of word
dfa	Delete until a char

Combination	Description
d/hello	Delete until hello
cc	Change current line
yy	Copy current line
>j	Indent 2 lines
ggdG	Delete a complete document
gg=G	Indent a complete document

Counts

```
[count] <operator> <motion>
<operator> [count] <motion>
```

2dd	Delete 2 lines
6yy	Copy 6 lines
d3w	Delete 3 words
d5j	Delete 5 lines downwards
>4k	Indent 4 lines upwards

Text objects

Usage

v	i / a	p
Operator	inner / around	Text object
Operate with an operator inner or around text blocks		

Text objects

p	Paragraph
w	Word
W	WORD (surrounded by whitespace)

s	Sentence
[({<	A [], (), or {} block
])}>	A [], (), or {} block
'"`	A quoted string
b	A block [(
B	A block in [{
t	A HTML tag block

See `:help text-objects`

diw	Delete inner word
dis	Delete inner sentence
di"	Delete in quotes
da"	Delete in quotes (including quotes)
dip	Delete a paragraph

Selections

vi"	Select inner quotes "...."
va"	Select quotes "...."
vi[	Select inner brackets [....]
va[	Select brackets [....].
viw	Select inner word
vip	Select inner paragraph
vipip	Select more paragraph

Misc

ciw	Change inner word
ci"	Change inner quotes
cit	Change inner tags (HTML)

`cip`	Change inner paragraph
`yip`	Yank inner paragraph
`yap`	Yank paragraph (including newline)

Multiple files

`:e file`	Edit a file in a new buffer
`:bn`	Go to the next buffer
`:bp`	Go to the previous buffer
`:bd`	Remove file from buffer list
`:b 5`	Open buffer #5
`:b file`	Go to a buffer by file
`:ls`	List all open buffers
`:sp file`	Open and split window
`:vs file`	Open and vertically split window
`:hid`	Hide this buffer
`:wn`	Write file and move to next
`:tab ba`	Edit all buffers as tabs

`<C-w>`	`s`	Split window
`<C-w>`	`v`	Split window vertically
`<C-w>`	`w`	Switch windows
`<C-w>`	`q`	Quit a window
`<C-w>`	`T`	Break out into a new tab
`<C-w>`	`x`	Swap current with next

`<C-w>` `-` / `+`		Decrease/Increase height
`<C-w>` `<` / `>`		Decrease/Increase width
`<C-w>` `|`		Max out the width
`<C-w>` `=`		Equally high and wide
`<C-w>` `h` / `l`		Go to the left/right window
`<C-w>` `j` / `k`		Go to the up/down window

Tabs

`:tabe [file]`	Edit file in a new tab
`:tabf [file]`	Open if exists in new tab
`:tabc`	Close current tab
`:tabo`	Close other tabs
`:tabs`	List all tabs
`:tabr`	Go to first tab
`:tabl`	Go to last tab
`:tabm 0`	Move to position 0
`:tabn`	Go to next tab
`:tabp`	Go to previous tab

Normal mode	
`gt`	Go to next tab
`gT`	Go to previous tab
`2gt`	Go to tab number 2

Search and Replace

Search

`/foo`	Search forward

`/foo\c`	Search forward (case insensitive)
`?foo`	Search backward
`/\v\d+`	Search with regex
`n`	Next matching search pattern
`N`	Previous match
`*`	Search for current word forward
`#`	Search for current word backward

Replace LINE

`:[range]s/{pattern}/{str}/[flags]`

`:s/old/new`	Replace first
`:s/old/new/g`	Replace all
`:s/\vold/new/g`	Replace all with regex
`:s/old/new/gc`	replace all (Confirm)
`:s/old/new/i`	Ignore case replace first
`:2,6s/old/new/g`	Replace between lines 2-6

Replace FILE

`:%s/{pattern}/{str}/[flags]`

`:%s/old/new`	Replace first
`:%s/old/new/g`	Replace all
`:%s/old/new/gc`	Replace all (Confirm)
`:%s/old/new/gi`	Replace all (ignore case)
`:%s/\vold/new/g`	Replace all with regex

Ranges

`%`	Entire file
`'<,'>`	Current selection
`5`	Line 5

`5,10`	Lines 5 to 10
`$`	Last line
`2,$`	Lines 2 to Last
`.`	Current line
`,3`	Next 3 lines
`-3,`	Forward 3 lines

Global command

`:[range]g/{pattern}/[command]`

`:g/foo/d`	Delete lines containing foo
`:g!/foo/d`	Delete lines not containing foo
`:g/^\s*$/d`	Delete all blank lines
`:g/foo/t$`	Copy lines containing foo to EOF
`:g/foo/m$`	Move lines containing foo to EOF
`:g/^/m0`	Reverse a file
`:g/^/t.`	Duplicate every line

Inverse :g

`:[range]v/{pattern}/[command]`

`:v/foo/d`	Delete lines not containing foo (also `:g!/foo/d`)

Flags

`g`	Replace all occurrences
`i`	Ignore case
`I`	Don't ignore case
`c`	Confirm each substitution

Substitute expression (magic)

`& \0`	Replace with the whole matched

\1...\9	Replace with the group 0-9
\u	Uppercase next letter
\U	Uppercase following characters
\l	Lowercase next letter
\L	Lowercase following characters
\e	End of \u, \U, \l and \L
\E	End of \u, \U, \l and \L

```
:s/a\|b/xxx\0xxx/g                  # Modifies "a b"     to "xxxaxxx xxxbxxx"
:s/test/\U& file/                   # Modifies "test"    to "TEST FILE"
:s/\(test\)/\U\1\e file/            # Modifies "test"     to "TEST file"
:s/\v([abc])([efg])/\2\1/g         # Modifies "af fa bg" to "fa fa gb"
:s/\v\w+/\u\0/g                     # Modifies "bla bla"  to "Bla Bla"
:s/\v([ab])|([cd])/\1x/g           # Modifies "a b c d"  to "ax bx x x"
:%s/.*/\L&/                          # Modifies "HTML"     to "html"
:s/\v<(.)(\w*)/\u\1\L\2/g          # Make every first letter of a word
uppercase
:%s/^\(.*\)\n\1/\1/                  # Remove duplicate lines
:%s/<\/\=\(\w\+\)\>/\U&/g          # Convert HTML-Tags to uppercase
:g/^pattern/s/$/mytext              # Find and append text to the end
:g/pattern/norm! @i                 # Run a macro on matching lines
/^\(.*\)\(\r\?\n\1\)\+$             # View the duplicates lines
:v/./,/./-j                          # Compress blank lines into a blank line
:g/<p1>/,/<p2>/d                    # Delete inclusively from <p1> to <p2>
```

Vimdiff

```
$ vimdiff file1 file2 [file3]
$ vim -d file1 file2 [file3]
```

```
:[range]diffget [bufspec]
:[range]diffput [bufspec]
```

`do` / `:diffget`	Obtain (get) difference
`dp` / `:diffput`	Put difference
`:dif`	Re-scan differences
`:diffo`	Switch off diff mode
`:1,$+1diffget`	Get all differences
`ZQ`	Quit without changes
See: Ranges	

Folds

`zo` / `zO`	Open
`zc` / `zC`	Close
`za` / `zA`	Toggle
`zv`	Open folds for this line
`zM`	Close all
`zR`	Open all
`zm`	Fold more (foldlevel += 1)
`zr`	Fold less (foldlevel -= 1)
`zx`	Update folds

Jumping

`]c`	Next difference
`[c`	Previous difference

Miscellaneous

Case

`vU`	Uppercase character
`vu`	Lowercase character
`~`	Toggle case character
`viw` `U`	Uppercase word
`viw` `u`	Lowercase word
`viw` `~`	Toggle case word
`VU` / `gUU`	Uppercase line
`Vu` / `guu`	Lowercase line
`V~` / `g~~`	Toggle case line
`gggUG`	Uppercase all text
`ggguG`	Lowercase all text
`ggg~G`	Toggle case all text

`<C-o>`	Go back to previous
`<C-i>`	Go forward
`gf`	Go to file in cursor
`ga`	Display hex, ascii value

`:h`	Help open help view
`:edit!`	Reload current file
`:2,8m0`	Move lines 2-8 to 0
`:noh`	Clear search highlights
`:sort`	Sort lines
`:ter`	Open a terminal window
`:set paste`	Enable Insert Paste sub-mode
`:set nopaste`	disable Insert Paste sub-mode

`:cq`	Exiting with an error
	(aborting Git)

`%`	Nearest/matching `{[()]}`
`[(` ⎸ `[{` ⎸ `[<`	Previous `(`, `{` or `<`
`])` ⎸ `]{` ⎸ `]>`	Next `)`, `}` or `>`
`[m`	Previous method start
`[M`	Previous method end

Counters

`<C-a>`	Increase number
`<C-x>`	Decrease number

Tags

`:tag Classname`	Jump to first definition of Classname
`<C-]>`	Jump to definition
`g]`	See all definitions
`<C-t>`	Go back to last tag
`<C-o> <C-i>`	Back/forward
`:tselect Classname`	Find definitions of Classname
`:tjump Classname`	Find definitions of Classname (auto-select 1st)

Formatting

`:ce 8`	Center lines between 8 columns
`:ri 4`	Right-align lines at 4 columns
`:le`	Left-align lines

See `:help formatting`

Marks

`` `^ ``	Last position of cursor in insert mode
`` `. ``	Last change in current buffer

`"`	Last exited current buffer
`0`	In last file edited
''	Back to line in current buffer where jumped from
``	Back to position in current buffer where jumped from
`[`	To beginning of previously changed or yanked text
`]`	To end of previously changed or yanked text
`<`	To beginning of last visual selection
`>`	To end of last visual selection
ma	Mark this cursor position as a
`a	Jump to the cursor position a
'a	Jump to the beginning of the line with position a
d'a	Delete from current line to line of mark a
d`a	Delete from current position to position of mark a
c'a	Change text from current line to line of a
y`a	Yank text from current position to position of a
:marks	List all current marks
:delm a	Delete mark a
:delm a-d	Delete marks a, b, c, d
:delm abc	Delete marks a, b, c

Calculator

`<C-r>` = 7*7	Shows the result
`<C-r>` = 10/2	Shows the result

Do this in INSERT mode

Shell

:!<shell>	Interpret Shell Command
:r!<shell>	Read in output of shell

| `:r!date` | Insert date |
| `:!!date` | Replace current line with date |

`<C-r><C-w>`	Insert current word into the command line
`<C-r>"`	Paste from " register
`<C-x><C-f>`	Auto-completion of path in insert mode

Remove duplicate lines

```
:sort | %!uniq -u
```

To number the lines in the file

```
:%!cat -n
```

Copy whole doc to clipboard

```
:%w !pbcopy            # Mac OS X
:%w !xclip -i -sel c   # GNU/Linux
:%w !xsel -i -b        # GNU/Linux
```

Also see

Devhints (devhints.io)
Vim cheatsheet (vim.rotrr.com)
Vim documentation (vimdoc.sourceforge.net)
Interactive Vim tutorial (openvim.com)

POPULAR

Bash scripting
Vim
RegEX
QuickRef

RECENT

C++
Java
Tmux
Lsof

WEBSITE

Privacy Policy
About