

vi / vim graphical cheat sheet

Esc

normal mode

	1	2	3	4	5	6	7	8	9	0	.	{	}
2	& repeat :s	é	" reg. spec	' goto mk. bol	(begin sentence	- prev line	è	"soft" bol down	ç	à	°) end sentence	= auto format
A append at eol	Z quit	E end WORD	R replace mode	T back 'till	Y yank line	U undo line	I insert at bol	O open above	P paste before	..	£		
a append	Z extra cmds	e end word	r replace char	t 'till	y yank	u undo	i insert mode	o open below	p paste after	^ first non-blank	\$ eol		
Q ex mode	S subst line	D delete to eol	F "back" find ch	G eof/ goto ln	H screen top	J join lines	K man	L screen bottom	M screen mid'l	% goto match	µ		
q record macro	s subst char	d delete	f find char	g extra cmds	h ←	j ↓	k ↑	l →	m set mark	ù	*		
> indent	W next WORD	X back- space	C change to eol	V visual lines	B prev WORD	N prev (find)	? find (rev.)	.	/ find	§			
< un- indent	w next word	x delete char	c change	v visual mode	b prev word	n next (find)	reverse , t/T/f/F	.	. ex cmd line

motion moves the cursor, or defines the range for an operator

command direct action command, if **red**, it enters insert mode

operator requires a motion afterwards, operates between cursor & destination

extra special functions, requires extra input

q commands with a dot need a char argument afterwards

bol = beginning of line, eol = end of line, mk = mark, yank = copy

words: quux(foo, bar, baz);

WORDS: quux(foo, bar, baz);

Main command line commands ('ex') Notes:

:w (save), :q (quit), :q! (quit w/o saving)
:e f (open file f),
:s/s/x/y/g (replace 'x' by 'y' filewide),
:h (help in vim), :new (new file in vim),

Other important commands:

CTRL-R: redo (vim),
CTRL-F/-B: page up/down,
CTRL-E/-Y: scroll line up/down,
CTRL-V: block-visual mode (vim only)

Visual mode:

Move around and type operator to act on selected region (vim only)

(1) use "x before a yank/paste/del command to use that register ('clipboard') (x=a..z, (e.g.: "ay\$ to copy rest of line to reg 'a')

(2) type in a number before any action to repeat it that number of times (e.g.: 2p, d2w, 5i, d4j)

(3) duplicate operator to act on current line (dd = delete line, >> = indent line)

(4) ZZ to save & quit, ZQ to quit w/o saving

(5) zt: scroll cursor to top,

zb: bottom, zz: center

(6) gg: top of file (vim only),

gf: open file under cursor (vim only)

:qw exit

:q ! exit without save

d number w delete [number] word

d \$ delete from cursor to eol

d d delete line X dd x cut line

x delete under cursor

a add text

i insert text

A insert at end of line I insert text at bol

o open line under cursor, and switch in insert mode

O open line above cursor, and switch in insert mode

r character replacement

ce input word replacement

c\$ input replacement until end of line

R input replacement until ECHAP

DÉPLACEMENT

e at the end of current word

\$ at the end of the line

w at the beginning of the next word

b at the beginning of the current word

O beginning of line

^ to the first non-white character of the line

() delete line

{ } delete paragraph

u undo

U undo all changes on the line

CTRL + R redo

g g go to beginning of file

G go to end of file

number G go to the number line

CTRL + f next page

CTRL + b previous page

: [number] go to the selected line

COPY TEXT

- place on the first letter of the word

push on V (visual mode)

- move to the desired selection

push on y (copy), move, then p

MARKERS (normal mode)

m + [lettre] put a marker

m + [lettre] go to this marker

SEARCH

/ [character] ENTER then n N

POSIX : [] ^ \$ /

ENTER select the current occurrence

PROGRAMMATION

= = indent line

g g = G re-indent code

K find document under cursor

PLUGINS

, I f open the file explorer

, I r open the file explorer from the current file

CTRL + n select next file / directory

CTRL + p select previous file / directory

CTRL + w parent directory

CTRL + e create a file by typing its name before

, I g opens the buffers explorer (buffer)

, c ou CTRL + P open a file on the disk

once the file first in the list do

▼ to navigate select auto-completion

C S [present character] [character to delete]

gg v G = automatic page indentation

SPLIT

:sp split active window horizontally

:close close active window (blue color)

:only keep the active window alone

:vsp split active window vertically