

# Vim for Muggles™

## The Cheatsheet

### CHEATSHEET CONVENTIONS

- **Key combination** – several keys pressed together, e.g. `<CTRL>+AN` means `<CTRL>+<SHIFT>+aN`.
- **Key sequence** – several keys or combinations pressed gradually, e.g.. `<CTRL>+wN` means a press of combination `<CTRL>+wN` and then a key `wN`.
- **Working mode** – it is displayed bottom right:
  - `foON` – Normal mode
  - `foOI` – Insert mode
  - `foOV` – Visual mode
  - `:foO` – Command line mode

### BASIC CONTROL

- Repeat command with typing number  $n \times$  of repetitions before command, e.g. `5xN` deletes 5 chars from cursor position. Option `:set showcmd` shows command sequence in bottom right.
- Help: `:help topic`, use `Tabox` for command completion.
- Similar commands uses same keys, e.g. `fN`/`FN`, `tN`/`TN` (see below). The difference is direction or range of operation.
- Repeat last:..... command `.N`, macro `@QN`
- **Undo**:..... `uN`  $\approx$  `:undo`, redo `<CTRL>+rN` / `:redo`
- Set or unset options with `:set option` a `:set nooption`. Alone `:set` displays current options.
- Shortening of options:..... `:set ignorecase`  $\approx$  `:se ic`
- More ex-commands together:..... `:command1 | command2`
- Comparison of two files:..... `vimdiff file1 file2`

### EDIT AND SAVE

- Edit:..... `:edit file`
- Open new empty file:..... `:enew`
- Save/append file:..... `:write file` / `:w >> file`
- Save/append lines  $n$  to  $m$ :..... `:n,mw file` / `:n,mw>> file`
- Save and quit:..... `:wq`, enforced `:wq!`
- Quit without save:..... `:q!`  $\approx$  `ZQN`
- Quit and save when file was changed:..... `:x`  $\approx$  `ZZN`
- Go to file under the cursor:..... `gfN`
- Change name of current file:..... `:f jméno`

### MARKING OF POSITION

- Insert invisible mark a:..... `maN`
- Jump to mark a:..... `'aN`
- Print marks:..... `:marks`
- Erase mark a:..... `:delmarks a`
- Process text from mark a to b:..... `:'a,'b příkaz`, e.g. `:'a,'b !sort` will use external command for sort

### INSERT SPECIAL SYMBOLS

- Display unprintable characters:..... `:set list`
- Insert ASCII char:..... `<CTRL>+v code<END>`
- Show ASCII/UTF-8 of char under cursor:..... `gaN` / `g8N`
- Suppress control character, e.g. `<TAB>` (`^I`):.. `<CTRL>+v<TAB>`
- Table of Unicode digraphs:..... `:digraphs`
- Insert digraphs:..... `<CTRL>+k K1 K2`
- Digraphs:  $\pm$   $+$   $^\circ$  DG  $\times$  \*X  $\div$  -:  $-$  N  $-$  M „: 9 “: 6
- Define digraphs .3 for „...“ (U+2026):..... `:dig .3 8230`

### SEARCH AND REPLACE

- Ignore case:..... `:set ignorecase`, opposite `:set noic`
- Repeat last search:..... `/N` / `?N`
- Ignore case:..... `/^cpatternN`
- Search for word under the cursor:.. `#N` forward, .. `*N` backward
- Search by ASCII code of „M“:..... `%d77`, `%x4d`, `%o115`
- Search Unicode character „α“ (U+03B1):..... `%u03b1`
- Substitution:..... `:substitute/from/to/flag`, abbr. `:s`
- Repeat last substitution:..... `:s`  $\approx$  `s//~/`
- The delimiter / can be any char instead \, " and |

- Flags:.. c – confirmation, g – for whole line, p – print, without flag changes only first occurrence
- Cancel highlight:..... `:nohlsearch`
- Search in more files:..... `:vimgrep /pattern/g f1 f2 ...fn`

### REGULAR EXPRESSIONS

- Begin/end of:..... line `^/$`, word `\</\>`, file `\\^/\\$`
- Last searched pattern/substitution:..... `&/~`
- Matches any character:..... `(dot)`
- Or/and:..... `|` / `&`
- Suppress next special symbol:..... `\.`, `\&`, `\$`, `\^`, `\\`, `\*`
- Atom:..... `\(pattern\)`
- $N^{\text{th}}$  reference:..... `\N`,..... first `\(pattern\)` = `\1`
- Ranges:..... `[abcd]`  $\approx$  `[a-d]`, `[a-zA-Z]`  $\approx$  `\a`, `[0-9]`  $\approx$  `\d`
- Ignored chars and ranges:..... `[^abcd]`  $\approx$  `[^a-d]`
- Zero or multiple z:..... `z*`  $\approx$  `z\{0,\}`
- One or multiple z:..... `z+`  $\approx$  `z\{1,\}`
- One or zero times:..... `z?`  $\approx$  `z\{0,1\}`
- Repetition z:..... `z\{n\}` accurate  $n \times$ , `z\{n,\}` minimal  $n \times$ , `z\{n,m\}` min.  $n$  and max.  $m$  times

### EDIT MULTIPLE FILES

#### BUFFERS

- Start vim:..... `vim file1 file2...`
- Display buffers:..... `:ls`  $\approx$  `:files` (numbered)
- Switch to next/previous file:..... `:next` / `:prev`
- Switch to first/last file:..... `:rewind`  $\approx$  `:first` / `:last`
- Perform mass *command* for all files:..... `:bufdo command`

#### WINDOWS

- New window:..... `:new`  $\approx$  `<CTRL>+wnN`
- Split window:..... `:split`  $\approx$  `<CTRL>+wsN`
- Vertically split window:..... `:vsplit`  $\approx$  `<CTRL>+wvN`
- Close window:..... `:close`  $\approx$  `<CTRL>+wcN`
- Close all except current:..... `:only`  $\approx$  `<CTRL>+woN`
- Jump to next window:..... `<CTRL>+wwN`
- Shrink/large window by  $n$  lines:.. `<CTRL>+wnN` / `<CTRL>+wnN`
- Shrink/large window by  $n$  columns: `<CTRL>+wnN` / `<CTRL>+wnN`
- Same size for all windows:..... `<CTRL>+wN`
- Mass:..... write `:wall`, quit `:qall`, `:wqall`
- Perform mass *command* for all windows:.... `:windo command`
- Open name under cursor in new window:..... `<CTRL>+wfN`

#### MOVING IN WINDOW

- Jump to window line:..... top `HN`, middle `MN`, last `LN`
- Jump to  $n^{\text{th}}$  line from top/bottom:..... `nHN` / `nLN`
- Set actual line to:..... top `z<END>N`, center `z.N`, bottom `zN`
- Scroll window up/down by:line `<CTRL>+eN` / `<CTRL>+yN`, page `<CTRL>+bN` / `<CTRL>+fN`, half-page `<CTRL>+uN` / `<CTRL>+dN`
- Redraw window without scrolling:..... `<CTRL>+lN`

#### TABS

- Command:..... `vim -p f1, f2 ...` open files in tabs
- Create/close tab:..... `:tabnew` / `:tabc`
- Close all, except current tab:..... `:tabo`
- Go to next/previous: `:tabn`  $\approx$  `<CTRL>+<PAGEDOWN>` / `:tabp`  $\approx$  `<CTRL>+<PAGEUP>`
- Move after the fold  $n$ :..... `:tabm n`
- Perform mass *command*:..... `:tabdo command`
- Change tab design:..... `:set tabline`
- Open name of file under cursor in new tab:..... `<CTRL>+wgfN`

### PROGRAMMER SUPPORT

- Run make:..... `:make`
- Jump to next/previous error:..... `:cprevious` / `:cnext`
- List all errors:..... `:clist`
- Add window with list of errors:..... `:copen`

# NORMAL MODE

## SYNTAX

- $[n+]$  **operator** + **motion**: ..... operate during motion  $n$  times
- $[n+]$  **operator** + **textobject**: ..... operate on text object  $n$  times
- $[n+]$  (**operator** \* 2): ..... operate on current line  $n$  times
- $[n+]$  **action**: ..... perform action  $n$  times

Example

- **c** **G**<sub>N</sub> ..... change from here to end of file
- **3** **dd**<sub>N</sub> ..... delete 3 lines
- **2** **gq** **ap**<sub>N</sub> ..... format 2 paragraphs
- **10** **J**<sub>N</sub> ..... join 10 lines

## ACTIONS

- Repeat last command: ..... **.**<sub>N</sub>
- Undo: ..... **u**<sub>N</sub> line **U**<sub>N</sub>
- Replace character by x: ..... **rx**<sub>N</sub>
- Enter replace mode: ..... **R**<sub>N</sub>
- Enter insert mode before/after current character: **i**<sub>N</sub> / **a**<sub>N</sub>
- Enter insert mode start/end of the line: **I**<sub>N</sub> / **A**<sub>N</sub>
- Add new line: ..... after **O**<sub>N</sub>, before **O**<sub>N</sub>
- Join lines: ..... **J**<sub>N</sub>, without spaces **gJ**<sub>N</sub>
- Substitute: ..... character **s**<sub>N</sub> line **S**<sub>N</sub>
- Paste (put) register: ..... **p**<sub>N</sub>
- Delete character: ..... **x**<sub>N</sub>, before cursor **X**<sub>N</sub>
- Change to end of line: ..... **C**<sub>N</sub>
- Increment/Decrement number: **<CTRL>+a**<sub>N</sub> / **<CTRL>+x**<sub>N</sub>

## OPERATORS

- Change: ..... **C**<sub>N</sub>, delete **d**<sub>N</sub>
- Copy (yank): ..... **y**<sub>N</sub>
- Filter through an external program: ..... **!**<sub>N</sub>
- Format: ..... **gq**<sub>N</sub>
- Shift left/right: ..... **<**<sub>N</sub> / **>**<sub>N</sub>
- Make lowercase/uppercase: ... **gu**<sub>N</sub> / **gU**<sub>N</sub>, swap case **g~**<sub>N</sub>

## MOTIONS

- Matching bracket/parenthesis: ..... **%**<sub>N</sub>
- Forward/Back. to start/end of word: **w**<sub>N</sub> / **b**<sub>N</sub> // **e**<sub>N</sub> / **ge**<sub>N</sub>
- Forward/Back. to start/end of WORD: **W**<sub>N</sub> / **B**<sub>N</sub> // **E**<sub>N</sub> / **gE**<sub>N</sub>
- Sentence forward/backwards: ..... **>**<sub>N</sub> / **<**<sub>N</sub>
- Paragraph forward/backwards: ..... **}**<sub>N</sub> / **{**<sub>N</sub>

### VERTICAL

- Goto line x: ..... **xG**<sub>N</sub>
- Goto file percent: ..... **x%**<sub>N</sub>
- Repeat last search in the same/reverse direction: **n**<sub>N</sub> / **N**<sub>N</sub>
- Start/End of file: ..... **gg**<sub>N</sub> / **G**<sub>N</sub>
- Lines up/down: ..... **k**<sub>N</sub> / **j**<sub>N</sub>
- Beginning of next/previous line: ..... **+n** / **-n**
- Search forward: ..... **/pattern**<sub>N</sub> / **?pattern**<sub>N</sub>
- Up/Down  $\frac{1}{2}$  page: ..... **<CTRL>+u**<sub>N</sub> / **<CTRL>+d**<sub>N</sub>
- Backwards/Forward a page: ... **<CTRL>+b**<sub>N</sub> / **<CTRL>+f**<sub>N</sub>
- Top/middle/bottom of visible text: ..... **H**<sub>N</sub> / **M**<sub>N</sub> / **L**<sub>N</sub>

### HORIZONTAL

- Character right/left: ..... **l**<sub>N</sub> / **h**<sub>N</sub>
- First/Last character: ..... **O**<sub>N</sub> / **\$**<sub>N</sub>
- First non-blank character: ..... **^**<sub>N</sub>
- Next/Last character x occurrence: ..... **fx**<sub>N</sub> / **Fx**<sub>N</sub>
- Before/After character x occurrence: ..... **tx**<sub>N</sub> / **Tx**<sub>N</sub>
- Repeat/Reverse last **f**<sub>N</sub>, **t**<sub>N</sub>, **F**<sub>N</sub>, **T**<sub>N</sub>: ..... **;**<sub>N</sub> / **;**<sub>N</sub>

## TEXT OBJECTS

- Word/WORD: ..... **aw**<sub>N</sub>, **iw**<sub>N</sub> / **aW**<sub>N</sub>, **iW**<sub>N</sub>
- Sentence: ..... **as**<sub>N</sub>, **is**<sub>N</sub>
- Paragraph: ..... **ap**<sub>N</sub>, **ip**<sub>N</sub>
- Enclosed text **[]**, **<>**, **()**, **{}** → **enc**: ..... **aenc**<sub>N</sub>, **aenc**<sub>N</sub>
- Quoted text **"**, **"**, **`** → **quot**: ..... **aquot**<sub>N</sub>, **iquot**<sub>N</sub>
- XML/HTML tag: ..... **at**<sub>N</sub>, **it**<sub>N</sub>

# INSERT & REPLACE MODES

- Insert previously inserted text: ..... **<CTRL>+a**<sub>i</sub>
- Delete word before the cursor: ..... **<CTRL>+w**<sub>i</sub>, all **<CTRL>+u**<sub>i</sub>
- Find next/previous keyword: ..... **<CTRL>+n**<sub>i</sub> / **<CTRL>+p**<sub>i</sub>
- Insert the contents of register x: ..... **<CTRL>+r**<sub>x</sub>
- Indent right/left current line: ..... **<CTRL>+t**<sub>i</sub> / **<CTRL>+d**<sub>i</sub>
- Insert character below/above the cursor: **<CTRL>+e**<sub>i</sub> / **<CTRL>+y**<sub>i</sub>

- Enter CTRL-X mode (completion): ..... **<CTRL>+x**<sub>i</sub>
  - Whole lines: ..... **<CTRL>+l**<sub>i</sub>
  - Keywords in the current file: ..... **<CTRL>+n**<sub>i</sub>
  - Keywords in 'dictionary': ..... **<CTRL>+k**<sub>i</sub>
  - Vim command-line: ..... **<CTRL>+v**<sub>i</sub>
  - File names: ..... **<CTRL>+f**<sub>i</sub>

# VISUAL MODE

## OPERATORS

Normal Mode operators apply to selection. Additionally:

- Block insert/append: ..... ( **I**<sub>V</sub> / **A**<sub>V</sub> ) text **<Esc>**<sub>i</sub>
- Perform ex command in selected range: **;**<sub>V</sub> → **'<**, **'>** command

## CHANGING SELECTION

Selection can be modified using Normal Mode motions and text objects.

- Changing the visual area top-left/bottom-right: ..... **o**<sub>V</sub> / **O**<sub>V</sub>

# COMMAND LINE MODE

## SYNTAX

- **:xcmd** ..... execute on line  $x$
- **:x,ycmd** ..... from  $x$  to line  $y$
- **:\$cmd** ..... from actual position to the end
- **:%cmd** ≈ **:1,\$cmd** ..... for whole file
- **..,+xcmd** ≈ **x:cmd** ..... from cursor to next  $x$  lines
- **..,\$-xcmd** ..... from current position to  $x$ . line from the end
- **:/pattern<sub>1</sub>/,/pattern<sub>2</sub>/cmd** ..... from the first occurrence of line with  $pattern_1$  to the first  $pattern_2$

## COMMANDS

- **:global/pattern/[cmd]**: ..... run cmd on matching lines
- **:read [name]**: ..... insert file below the cursor
- **:read !{cmd}**: ..... execute shell cmd and insert output
- **:substitute/pattern/string/[flags]**: ..... replace matches
- **[ :copy / :move / :put ] + addr:** ..... the lines below  $addr$
- **:center** / **:right** / **:left**: ..... align text
- **:delete [x]**: ..... delete lines [into register  $x$ ]