

Vi Improved

© 2014 Martin Bruchanov, bruxy@regnet.cz

Basic control and conventions

- Key combination** – several keys pressed together, e.g. `CTRL+AE` means `CTRL+SHIFT+aE`.
- Key sequence** – several keys or combinations pressed gradually, e.g. `CTRL+wwE` means a press of combination `CTRL+W` and then a key `w`.
- Working mode** – it is displayed bottom right:
 - C – basic command mode, `ESCE`.
 - E – insert mode `INSC`, `iE`.
 - V – visual, used for highlighting of text blocks `vC`, `VC`.
 - ex – command line ex-mode `:C`.
- Repeat command with typing number $n \times$ of repetitions before command, e.g. `5xC` deletes 5 chars from cursor position. Option `:set showcmd` shows command sequence in bottom right.
- Help: `:help topic`, use `TabC` for command completion.
- Similar commands uses same keys, e.g. `fC`/`FC`, `tC`/`TC` (see below). The difference is direction or range of operation.
- Repeat last:..... command `..C`, macro `@C`.
- Undo**:..... `uC` \approx `:undo`, redo `CTRL+rC` / `:redo`.
- Set or unset options with `:set option` a `:set nooption`. Alone `:set` displays current options.
- Shortening of options:..... `:set ignorecase` \approx `:se ic`.
- Line addressing** of ex-commands:
 - `:xcommand`..... execute on line x
 - `:x,ycommand`..... from x to line y
 - `:$command`..... from actual position to the end
 - `:%command` \approx `:1,$command`..... for whole file
 - `:. ,+xcomm` \approx `x:comm`..... from cursor to next x lines
 - `:. ,-$-x`..... from current position to x . line from the end
 - `:/pattern1/, /pattern2/`..... from the first occurrence of line with `pattern1` to the first `pattern2`
- More ex-commands together:..... `:command1 | command2`
- Comparison of two files:..... `vimdiff file1 file2`

2. Edit and save

- Edit:..... `:edit file`
- Open new empty file:..... `:enew`
- Insert file to current position:..... `:read file`
- Insert output of system `command`:..... `:read !command`
- 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 `ZQC`
- Quit and save when file was changed:..... `:x` \approx `ZZC`
- Go to file under the cursors:..... `gfC`
- Change name of current file:..... `:f jméno`
- Insert before/after:..... char `iC`/`aC`, line..... `IC`/`AC`
- Insert new line above/below cursor:..... `OC`/`oC`
- Rewrite:char `rC`, word `cwC`, line `ccC`, text `RC` \approx `INSINSC`
- Delete char/line and continue edit:..... `sC`/`SC`

2.1. Moving through file

- Open file on 23rd line:..... `vim file +23`
- Cursor:..... `→C` \approx `lC`, `←C` \approx `hC`, `↑C` \approx `kC`, `↓C` \approx `jC`
- Paging:..... `CTRL+fC` \approx `PAGEUPE` / `CTRL+bC` \approx `PAGEDOWNE`
- Half-paging up/down:..... `CTRL+uC` / `CTRL+dC`
- Jump to n^{th} line:..... `:n` \approx `nGC`, the last `:$` \approx `GC`
- Jump to percentual line position:..... `n%C`
- Display current line number:..... `CTRL+gC`, `:set number`
- Display number of lines:..... `=`
- Jump to byte position:..... `:goto n`
- Jump in line before char a forward/backward:..... `taC`/`TaC`
- Jump in line to char a forward/backward:..... `faC`/`FaC`
- Repeat last line search forward/backward:..... `:c`/`:C`

- Go to beginning of next/previous word:..... `wC`/`bC`
- Go to end of a word:..... `eC`..... with interpunction `EC`
- Go to declaration:..... `gdC`
- Go to begin/end of paragraph:..... `{C`/`}C, block: [{C/]C`
- Go to begin/end of line:..... `HOMEC` \approx `^C` / `ENDC` \approx `$C`
- Go to next/previous sentence:..... `>C`/`<C`
- Go to next/previous line:..... `ENDC` \approx `+C` / `-C`
- Go to column:..... first `OC` \approx `lC`, n^{th} `n|C`
- Jump to pair bracket:..... `%C`
- Print position, number of lines and chars, etc.:..... `gCTRL+gC`

2.1.1. Marking of position

- Insert invisible mark a :..... `maC`
- Jump to mark a :..... `'aC`
- 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

2.2. Deleting, moving and copying of text

- Deletion automatically stores text in register (for moving)
- Delete:char `xC` \approx `DELC`, previous char `XC` \approx `d←C` \approx `dhC`, `CTRL+hE` \approx `BACKSPACEE`, word `dwC` \approx `CTRL+wE`, line `ddC`
- Delete up to pair bracket:..... `d%C`
- Delete to end/beginning of sentence:..... `d|C`/`d|C`
- Delete the rest of line/file:..... `DC` \approx `d$C` / `dGC`
- Delete from cursor position to beginning:..... `d_C` \approx `d^C`
- Join lines:..... `JC` add space,..... `gJC` without space
- Copy to register:..... line `YC` \approx `yyC`, word `ywC`
- Copy to the beginning/end of line:..... `yOC`/`y$C`
- More registers: press `"k...C` before command will save text to register k . E.g. `"kyyC` copies line into k
- Paste content of last used register or last deleted text: after `pC`, before `PC` cursor, `"kppC` pastes register k
- Special register: `%` – current file name, `/` – last search,...
- Show content of registers:..... `:register a`
- Swaps the next two chars/lines:..... `xpC` / `ddpC`
- ex-mode:..... delete `:d`, move `:m`, copy `:co` \approx `:t`
- `:10,20m0` – move lines 10–20 to the beginning (0)
- `:. ,+5d a` – delete next 5 lines and store them to reg. a

2.3. Formatting

- Set text width:..... `:set textwidth=n`
- Switch on line wrap:..... `:set wrap`
- Justify:..... `:center`, `:right`, `:left`
- Format line:..... `gqC`
- Format paragraph:..... `ggapC` jump to end, `gwapC`
- Reverse case of char:..... `~C` line `g~C`
- Line to upper/lower:..... `gUUC` / `VuC` \approx `guuC` / `VUC`
- Whole file to upper/lower case:..... `gggUGC` / `ggguGC`
- While line/file `ROT13` encoding:..... `g??C` / `ggg?GC`

2.3.1. Indentation

- Shift line:..... `>>C`, `CTRL+tE` of `:set shiftwidth=n`
- Shift line leftwards:..... `<<C`, `CTRL+dE`
- Tabbing:..... `TABE` \approx `CTRL+iE`
- Shift line same as before:..... `=C`
- Indent block relatively:..... `=aC`
- Refresh indentation of whole file:..... `gg=GC`

2.4. Complementation

- Complement word before/after cursor:..... `CTRL+pE` / `CTRL+nE`
- Name of file in working directory:..... `CTRL+x` `CTRL+fE`
- Name of directory:..... `CTRL+x` `CTRL+dE`
- Whole line:..... `CTRL+x` `CTRL+lE`
- Word from dictionary:..... `CTRL+x` `CTRL+kE`
- System command (reads `$PATH`):..... `CTRL+x` `CTRL+vE`
- Increase/decrease number under cursor:..... `CTRL+aC` / `CTRL+xC`
- Name of current file:..... `CTRL+rE`

2.4.1. Completion at ex-command line

- Insert content of register a :..... `CTRL+raC`
- Insert word under the cursor:..... `CTRL+r` `CTRL+aC`

- Cancel command:..... `CTRL+uC`

2.5. Insert special symbols

- Display unprintable characters:..... `:set list`
- Insert ASCII char:..... `CTRL+v codeENDE`
- Show ASCII/UTF-8 of char under cursor:..... `gaC` / `g8C`
- Suppress control character, e.g. `TAB` (`^I`):..... `CTRL+v TABE`
- Table of Unicode digraphs:..... `:digraphs`
- Insert digraphs:..... `CTRL+k K1 K2E`
- Digraphs: `±+- °DG ×*X ÷-: --N --M „:9 “:6`
- Define digraphs `.3` for „...“ (U+2026):..... `:dig .3 8230`

3. Search and replace

- Ignore case:..... `:set ignorecase`, opposite `:set noic`
- Search forward/backward:..... `/patternC` / `?patternC`
- Repeat last search:..... `/C` / `?C`
- Ignore case:..... `/\cpatternC`
- Jump to next searched place:..... `nC` forward, `NC` backward
- Search for word under the cursor:..... `#C` forward, `*C` backward
- Search by ASCII code of „M“:..... `\%d77, \%x4d, \%o0115`
- Search Unicode character „a“ (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`
- Execute command on line with: `:global/pattern/command`
- Execute where pattern not match:..... `:g!/pattern/command`

3.1. 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^{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

4. Folding of text blocks (:help folding)

- Fold text:..... `zfC`, selected `zfvC`
- Jump to beginning/end of fold:..... `[zC` / `]zC`
- Open fold under the cursor:..... `zoC`, recursively `zOC`
- Open/close all:..... `zRC` / `zMC`
- Switch fold-unfold:..... `zaC`, recursively `zAC`
- Highlight folded parts:..... `:set foldcolumn=5`

5. Edit more files, windows and tabs

5.1. More files (edit 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`

5.2. Windows

- New window:..... `:new` \approx `CTRL+wnC`
- Split window:..... `:split` \approx `CTRL+wsC`
- Vertically split window:..... `:vsplit` \approx `CTRL+vwC`
- Close window:..... `:close` \approx `CTRL+wcC`
- Close all except current:..... `:only` \approx `CTRL+woC`
- Jump to next window:..... `CTRL+wwC`
- Shrink/large window by n lines:..... `CTRL+wn-C` / `CTRL+wn+C`

- Shrink/large window by n columns:..... `CTRL+wn<C` / `CTRL+wn>C`
- Same size for all windows:..... `CTRL+w=C`
- Mass:..... write `:wall`, quit `:qall`, `:wqall`
- Perform mass `command` for all windows:..... `:windo command`
- Open name under cursor in new window:..... `CTRL+wfC`

5.2.1. Moving in window

- Jump to window line:..... top `HC`, middle `MC`, last `LC`
- Jump to n^{th} line from top/bottom:..... `nHC` / `nLC`
- Set actual line to:..... top `zENDC`, center `z.C`, bottom `z-C`
- Scroll window up/down by:line `CTRL+eC` / `CTRL+yC`, page `CTRL+bC` / `CTRL+fC`, half-page `CTRL+uC` / `CTRL+dC`
- Redraw window without scrolling:..... `CTRL+lC`

5.3. 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+PAGEDOWNE` / `:tabp` \approx `CTRL+PAGEUPE`
- 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+wgfC`

6. Visual mode

- Select:..... chars `vC`, lines `VC`, cursor sets range
- Block text selection:..... `CTRL+VC`
- Change to upper/lower case:..... `U` / `u`
- Select:..... word `awC`, `()/{}block` `abC` / `aBC`, no brackets `ibC` / `iBC`, XML block with marks `atC`, inside marks `itC`
- Perform `command` in selected range:..... `:'<,'> command`

7. Abbreviations, key maps and macros

- Define abbreviation:..... `:abbreviate abbr full expansion`
- Abbreviations for insert/ex-mode:..... `:iabbrev` / `:cabbrev`
- List/delete abbreviations:..... `:ab` / `:unab abbreviation`
- Map key x :..... `:map x commands`
- Map by mode:..... `:map` `:imap` `:cmap` `:nmap` `:vmap`
- Map key for insert mode:..... `:map! x commands`
- List key mapping:..... `:map` / `:map!`
- Delete mapping:..... `:unmap x` / `:unmap! x`
- Write macro to register a :..... `qaC`
- End macro definition:..... `qC`
- Perform macro in register a :..... `@aC`
- Repeat last called macro:..... `@@C`
- Execute macro a as ex-command:..... `:@a`

8. Spell checking

- Switch on spelling:..... `:set spell spelllang=cs,en`
- Search next/previous mistype:..... `lsC` / `lSC`
- Add to dictionary a word under cursor:..... `zgC`
- Add temporary to dictionary:..... `zGC`
- Set word as bad:..... `zwC`, temporary `zWC`
- Suggest correctly spelled words:..... `z=C`
- Always take the first suggestion:..... `lz=C`

9. Programmer support

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

10. Other

- Start shell:..... `:shell`
- Open manpage of command under cursor:..... `KC`
- Open file under the cursor:..... `gfC`
- Save as color HTML:..... `:runtime! syntax/2html.vim`
- Wildcard of current file $\%$, e.g. `!:context %` process current file with `ConTeXt`