

8 Vi

8.1 Die wichtigsten Vi-Befehle

~/.exrc ~/.vimrc ESC ESC :q! CR	Vi/Vim-Konfigurationsdatei [ex/vim resource] Kommando-Modus beenden (2x → piepst + ignoriert) Vi sicher verlassen (ohne Änderung an Datei)
vi [FILE] CR :wq CR :x CR ZZ :q CR :q! CR :w CR :w FILE CR	Datei FILE bzw. leere Datei (:w FILE <i>nötig!</i>) editieren Editierte Datei schreiben + Vi verlassen [write+quit/exit] Editierte Datei nicht schr. + Vi verlassen (!= <i>erzwingen!</i>) [quit] Editierten Text (auf Datei FILE) schreiben [write]
:e[!] FILE CR :f FILE CR	Datei FILE statt aktueller editieren (!= <i>erzwingen!</i>) [edit] Aktuelle Datei mit dem Namen FILE benennen [file]
:n CR :prev :N VIM! :args CR :rew CR	Nächste Datei editieren (falls mehrere angegeben) [next] Vorherige Datei editieren [previous/Next] Alle beim Aufruf des Vi angeg. Dateien anzeigen [arguments] Zur 1. beim Aufruf angegeb. Datei zurückspringen [rewind]
a TEXT... ESC A TEXT... ESC i TEXT... ESC I TEXT... ESC o TEXT... ESC O TEXT... ESC	Nach Cursorposition TEXT... einfügen [append] Am Zeilenende TEXT... einfügen [Append] Vor Cursorposition TEXT... einfügen [insert] Am Zeilenanfang TEXT... einfügen [Insert] Neue Zeile TEXT... unter aktueller Zeile einfügen [open] Neue Zeile TEXT... über aktueller Zeile einfügen [Open]
x X dd dw dMOVE	Ein Zeichen unter/vor Cursor löschen [crossout] Zeile/Wort/gemäß Bewegung MOVE löschen [delete]
rX R TEXT... ESC s TEXT... ESC cc cw cMOVE ESC yy yw yMOVE ESC p P	Ein Zeichen unter Cursor mit X übertippen (<i>kein ESC!</i>) [replace] Ab Cursorposition mit TEXT... übertippen [Replace] Ein Zeichen unter Cursor mit TEXT... übertippen [substitute] Zeile/Wort/gemäß Bewegung MOVE ändern [change] Zeile/Wort/gemäß Bew. MOVE in Zw.speicher kopieren [yank/copy] Zwischensp. nach/vor akt. Zeichen/Zeile einfügen [put]
h l j k w b e W B E 0 ^ \$ NN <Ctrl-F> <Ctrl-B> G lG NNG mX 'X `X	Zeichen links/rechts (auch <BACKSPACE>/<SPACE>) Zeile ab/auf (auch +/-<RETURN>) Wort vor/zurück/Wortende [word/backword/endword] WORT vor/zurück/WORTende (<i>inkl. Ziffern, Satzzeichen!</i>) Zum Zeilenanfang/1.Buchstaben/Zl.ende/Spalte NN springen Bildschirmseite auf/abblättern [forward/backward] Zum Dateiende/Dateianfang/Zeile NN springen [Go] Marke x (a-z) setzen, Zeile/Zeichen mit M. x anspringen [mark]
/SUCH CR ?SUCH CR :%s/SUCH/TEXT/gc CR	Text SUCH vorwärts/rückwärts suchen Text SUCH überall in Datei durch TEXT ersetzen (%=1, \$=alle Zeilen, s=substitute, g=global, c=confirm)
n N	Letzte Suche vor/zurück wiederholen [next]
u U <Ctrl-R> VIM! .	Letzte Änderung/alle Änd. einer Zeile zurücknehmen [undo] Letzte Änderung wiederherstellen (2x u im Vi) [redo] Letztes Edit-Kommando (Änderung) wiederholen [Punkt]

Mit `VIM!` markierte Befehle sind nur im Editor Vim möglich.

8.2 Weitere wichtige Vi-Befehle

<code>%</code> <code>~</code> <code>J</code> <code><< >></code> <code>xp dwwP ddp</code> <code>v V <Ctrl-V> VIM!</code> <code>:e# <Ctrl-6></code>	Passende Klammer zu { [(< >)] } suchen Groß/Kleinschreibung von aktuellem Zeichen vertauschen Nachfolgende Zeile an aktuelle anhängen [join] Aktuelle Zeile einrücken/ausrücken (Option <code>shiftwidth</code>) Zwei Zeichen/Worte/Zeilen vertauschen Block zeichen/zeilen/spaltenorientiert markieren [visual] Vorherige Datei editieren
<code>"XyMOVE</code> <code>"XdMOVE</code> <code>"XP</code> <code>"Xp</code>	Von <code>MOVE</code> ausgewählten Text in Puffer <code>x</code> (a-z) kopieren [yank] Von <code>MOVE</code> ausgewählten Text in Puffer <code>x</code> (a-z) verschieben [del] Inhalt von Puffer <code>x</code> (a-z) vor Cursor/aktueller Zeile einfügen [put] Inhalt von Puffer <code>x</code> (a-z) nach Cursor/aktueller Zeile einfügen
<code>:map MACRO TEXT</code> <code>:unmap MACRO</code> <code>:map</code>	<code>MACRO</code> mit Ersatztext <code>TEXT</code> für Kommando-Modus definieren <code>MACRO</code> mit Ersatztext <code>TEXT</code> für Kommando-Modus löschen Alle Makros für Kommando-Modus anzeigen
<code>:map! MACRO TEX</code> <code>:unmap! MACRO</code> <code>:map!</code>	<code>MACRO</code> mit Ersatztext <code>TEXT</code> für Eingabe-Modus definieren <code>MACRO</code> mit Ersatztext <code>TEXT</code> für Eingabe-Modus löschen Alle Makros für Eingabe-Modus anzeigen
<code>:ab ABBREV TEXT</code> <code>:unab ABBREV</code> <code>:ab</code>	Abkürzung <code>ABBREV</code> für Eingabe-Modus definieren (<i>Leerzeichen!</i>) Abkürzung <code>ABBREV</code> für Eingabe-Modus löschen Alle Abkürzung für Eingabe-Modus anzeigen

8.3 Wichtige allgemeine Vi-Optionen

<code>errorbells</code> <code>ignorecase</code> <code>list</code> <code>number</code> <code>report=NN</code> <code>showmode</code> <code>wrapmargin=NN</code> <code>wrapscan</code>	Bei Eingabefehler piepsen Groß/Kleinschreibung bei der Suche ignorieren Zeilenenden durch <code>\$</code> und Tabulatoren durch <code>^I</code> anzeigen Zeilen numeriert anzeigen Änderungen ab <code>NN</code> Zeilen in Statuszeile anz. (<i>1=ab 1 Zeile, 0 geht n.!</i>) Vi-Modus in Statuszeile anzeigen (<code>INSERT</code> , <code>APPEND</code> , ...) Automatischer Umbruch <code>NN</code> Zeichen vor Zeilenende (<i>0=kein Umbruch!</i>) Suche über Dateiende/anfang hinweg fortsetzen
--	--

Optionen mit `:set OPTION` setzen und mit `:set noOPTION` (*zusammengeschrieben!*) löschen. In Konfigurations-Datei `.exrc/.vimrc` den Doppelpunkt vor `set` weglassen. Kommentare in `.exrc/.vimrc` durch `"` einleiten (*keine Leerzeilen erlaubt!*).

8.4 Wichtige Vi-Optionen für Programmierer

autoindent	Automatisch einrücken (mit <Ctrl-D/T> ein/ausrücken)
shiftwidth=NN	Breite NN für nachträgliches Einrücken setzen (<< >>)
showmatch	Passende öffnende Klammer bei Eingabe der schließenden anzeigen
tabstop=NN	Tabulatorbreite auf NN Zeichen setzen

8.5 Wichtige Vim-Optionen

nocompatible	Nicht im Vi-Kompatibilitätsmodus arbeiten
ruler	Koordinatenanzeige (Zeile + Spalte) in Statuszeile [Lineal]
:syntax on	Syntaxcoloring an (abhängig von Datei-Extension)
wrap	Zu lange Zeilen umgebrochen darstellen
incsearch	Inkrementell suche [incremental]
hlsearch	Zur Suche passende Textteile markieren [highlight]
showcmd	Unvollständige Kommandos in Statuszeile anzeigen [command]
ff=unix	Dateiformat (dos, unix, mac) [fileformat]

8.6 Nützliche Vi-Makros

K	!}fmt -80 -u^M	Absatz (bis Leerzeile) auf 80 Zeichen Breite formatieren
D	:%s/ *\$//^M	Alle Leerzeichen am Zeilenende entfernen
Ctrl-W	/\<	Suche nach Wort einleiten (danach Wort eintippen)
Ctrl-W	\>/^M	Suche nach Wort durchführen (<i>mit :map! definieren!</i>)
+	:n^M	Zur nächsten Datei springen (ohne Abspeichern)
#	:w^M:n^M	Zur nächsten Datei springen (aktuelle vorher speichern)
-	:prev^M VIM!	Zur vorherigen Datei springen (ohne Abspeichern)

Mit :map NAME MACRO definieren, ^M ist einzugeben durch <Ctrl-V> <RETURN> oder im Vim durch <CR>.