



vi-Training

Johannes Franken
<jfranken@jfranken.de>

Caution:

Please note: This document is available in german only. An english translation for this document is in progress

Contents

1. [Konzept](#)
2. [Erste Schritte](#)
3. [Navigation](#)
4. [Einfügen](#)
5. [Löschen](#)
6. [Undo/Redo](#)
7. [Copy/Paste](#)
8. [Ändern](#)
9. [Suchen](#)
10. [Ersetzen](#)
11. [Shell-Aufrufe](#)
12. [Fenster](#)
13. [Completion](#)
14. [Syntax Highlighting](#)
15. [Dateien vergleichen](#)

Konzept

- **vim** steht für "visual editor, improved"
- Entwicklung: ed, ex, vi, (vim, gvim)
- Konzept:
 - Datei
 - Cursor
 - Bearbeitungsfenster
- Bedienung über folgende Modi:
 - Bearbeitungsmodus (beginnen z.B. mit "i", beenden mit "[Esc])
 - Command (ein oder mehrere Zeichen, Beispiele: s.u.)
 - Commandline (beginnt mit ":", endet mit [Return])

Erste Schritte

[Strg]+g	status
:w Dateiname [Return]	speichern
:q [Return]	beenden
:q! [Return]	beenden ohne speichern
:wq [Return]	speichern und beenden (auch ZZ, :x)
:wq! [Return]	speichern und beenden, dabei ggf. überschrieben

Navigation

Cursortasten hkjl + - Enter	schrittweise
w W b B	wortweise
gg G 200G	Dateianfang Ende Zeile 200
0 \$	Zeilenanfang Zeilenende

Einfügen

i	am Cursor
I	am Zeilenanfang (falls Block markiert: vor dem Block)
o	über dem Cursor
O	unter (Zeile) dem Cursor
a	hinter dem Cursor
A	am Zeilenende

Löschen

x	löscht das Zeichen unter dem cursor (Entf.)
X	löscht das Zeichen VOR dem Cursor (Backspace)
30x	30 Zeichen
dw	bis zum Wortende
D	bis Zeilenende (entspr. d\$)
dd	aktuelle Zeile
2dd	zwei Zeilen
dG	bis Dateiene
d) d(bis Satzende ab Satzanfang
d} d{	bis Absatzende ab Absatzanfang

Löschungen landen im Puffer "*" (Zwischenablage)

Undo/Redo

u	u undo
[Strg]+r	redo
.	letztes Kommando wiederholen
200.	letztes Kommando 200 mal wiederholen

Copy/Paste

Text markieren:

[gehe zu Anfang] v [gehe zu Ende]	zeichenweise markieren
[gehe zu Anfang] V [gehe zu Ende]	zeilenweise markieren
[gehe zu Anfang] [Strg]+v [gehe zu Ende]	blockweise markieren

In Zwischenablage kopieren:

y	(yank) markierten Bereich
yy	ganze Zeile
y/j	alle Zeichen bis zum nächsten j

Zwischenablage einfügen:

p	paste nach/rechts von Cursor
P	paste vor/links von Cursor

Beispiele:

ddp	Zeilen vertauschen
xp	Zeichen vertauschen

Ändern

rj	ersetzt aktuelles Zeichen durch j
5rj	ersetzt nächste 5 Zeichen durch j
R	R wechselt in Überschreiben-Modus
C	C ersetzt bis Zeilenende (entspr. c\$)
cw	cw ersetzt bis Wortende
c) c(ersetzt bis Satzende ab Satzanfang
c} c{	ersetzt bis Absatzende ab Absatzanfang

Suchen

/xyz	nach unten suchen
?xyz	nach oben suchen
n N	zum nächsten Treffer zum vorherigen Treffer
* #	nächstes Vorkommen aktuelles Wort vorheriges Vorkommen
%	gehe zu Gegenklammer
:set hls [Return] :set nohls [Return]	Hervorhebungen (Highlightsearch) aktivieren deaktivieren
zt zb zz	Bearbeitungsfenster so verschieben, dass Cursor ganz oben (top) unten (bottom) Mitte (zenter)

Ersetzen

`:Bereich s/vorher/nacher/Optionen`

Listing: s Syntax

Bereich:

ohne oder .	nur aktuelle zeile
%	gesamte Datei
3,500	Zeilen 3 bis 500
v V [Strg]+v	nur innerhalb der Markierung

Optionen:

g	global (mehrmals pro Zeile)
c	confirm
i	ignore-Case

Beispiele:

<code>:%s/flasch/falsch/gci [Return]</code>	flasch durch falsch ersetzen
<code>:%s,/,\,g [Return]</code>	Slashes umdrehen

Shell-Aufrufe

<code>!:ls [Return]</code>	Verzeichnis auflisten
<code>v [Cursortasten] :!sort [Return]</code>	Markierte Zeilen sortieren
<code>:%!wc [Return]</code>	Alle Buchstaben, Wörter und Zeilen zählen
K	Manpage zu Wort unter Cursor aufrufen

Fenster

<code>vim -o Datei1 Datei2 ...</code>	Mehrer Fenster untereinander
<code>:split [Return]</code> <code>:vsplit [Return]</code>	Mehrere Bearbeitungsfenster derselben Datei untereinander, nebeneinander
<code>:split Dateiname [Return]</code> <code>:vsplit Dateiname [Return]</code>	Weitere Datei in neuem Bearbeitungsfenster öffnen
<code>[Strg]+w [Cursortaste]</code>	in anderes Bearbeitungsfenster wechseln
<code>[Strg]+w [Strg]+f</code>	öffne Datei unter Cursor in neuem Fenster
<code>[Strg]+w c</code> <code>:q</code>	close
<code>[Strg]+w o</code> <code>:only</code>	only one window
<code>[Strg]+w +</code> <code>[Strg]+w -</code> <code>[Strg]+w <</code> <code>[Strg]+w ></code>	vertikal verkleinern, vertikal vergrößern, horizontal verkleinern, horizontal vergrößern
<code>20[Strg]+w +</code>	aktuelles Bearbeitungsfenster um 20 Zeilen vergrößern

Completion

<code>[Strg]+x [Strg]+n</code>	next
<code>[Strg]+x [Strg]+p</code>	previous
<code>[Strg]+x [Strg]+f</code>	Filesystem
<code>[Strg]+n</code> <code>[Strg]+p</code>	Vorschlag weiter, zurück

Syntax Highlighting

:syntax on [Return]	Syntaxhighlighting aktivieren
:set bg=dark [Return] :set bg=light [Return]	Kontrastverbesserung für schwarzen, weissen Hintergrund
:set syntax=php [Return]	Syntaxhighlighting für Sprache "PHP" auswählen

Dateien vergleichen

vim -d Datei1 Datei2 ...	grafischer Diff
:diff	refresh
zo	open fold
zc	close fold