



- ✓ Vendor Independent
- ✓ Community Supported
- ✓ Respected Worldwide
- ✓ Distribution Neutral





# 1.103.8 Grundlagen der Dateibearbeitung mit vi

Beschreibung: Die Prüfungskandidaten müssen in der Lage sein, Textdateien mit vi zu bearbeiten. Dieses Lernziel beinhaltet Navigation im vi, die grundlegenden vi Modi, Einfügen, Bearbeiten, Löschen, Kopieren und Auffinden von Text.

Die wichtigsten Dateien, Bezeichnungen und Anwendungen:

**vi ( vim / gvim )**

/, ?

h,j,k,l

G, H, L

i, c, d, dd, p, o, a

ZZ, :w!, :q!, :e!

:!



**Bedeutung:** Der Editor vi ist auf jedem Unix / Linux-System vorhanden



## vi und **vi improved** (vim)

➔ vi ist ein Editor (wie emacs), der extrem viele Möglichkeiten bietet. Er ist sehr schnell, aber man muss zunächst Zeit investieren, um mit ihm angenehm arbeiten zu können. Dafür wird man jedoch mit einer hohen Arbeitsgeschwindigkeit entschädigt.

➔ Es gibt eine Reihe von vi-Clones. Der auf dem Screenshot gezeigte gvim kann entweder wie vi über die Tastatur bedient werden oder auch mit der Maus. (Vim gibt es übrigens auch für Windows.  
<http://www.vim.org>)

➔ Für die Prüfung ist die Bedienung mit der Maus nicht relevant!

```
" Configuration file for vim
set runtimepath=~/.vim,/etc/vim,/usr/share/vim/vimfiles,/usr/share/vim/addons,/usr/share/vim/vim63,/usr/share/vim/vimfiles,
s,/usr/share/vim/addons/after,~/.vim/after

" Normally we use vim-extensions. If you want true vi-compatibility
" remove change the following statements
set nocompatible      " Use Vim defaults instead of 100% vi compatibility
set backspace=indent,eol,start  " more powerful backspacing

" Now we set some defaults for the editor
" set autoindent      " always set autoindenting on
" set linebreak       " Don't wrap words by default
set textwidth=0       " Don't wrap lines by default
set nobackup          " Don't keep a backup file
set viminfo='20,\"50  " read/write a .viminfo file, don't store more than
                    " 50 lines of registers
set history=50        " keep 50 lines of command line history
set ruler             " show the cursor position all the time

" Suffixes that get lower priority when doing tab completion for filenames.
" These are files we are not likely to want to edit or read.
set suffixes=.bak,~, .swp,.o,.info,.aux,.log,.dvi,.bbl,.blg,.brf,.cb,.ind,.idx,.ilg,.inx,.out,.toc

" We know xterm-debian is a color terminal
if &term =~ "xterm-debian" || &term =~ "xterm-xfree86"
    set t_Co=16
    set t_Sf=^[[3%dm
    set t_Sb=^[[4%dm
endif

" Make p in Visual mode replace the selected text with the "" register.
"/etc/vim/vimrc" 78L, 2878C

1,1  Anfang
```



## Arbeitsmodi

### Kommando-Modus (auch Normaler Modus genannt)

- ➔ Nach dem Start befindet man sich im Kommando-Modus. Man kann zunächst keinen Text editieren. Alle Tasten sind mit Kommandos wie Ausschneiden, Einfügen und v.a.m. belegt.

### Kommandozeilen-Modus

- ➔ Indem man einen **:** (**Doppelpunkt**) tippt, begibt man sich in den Kommandozeilen-Modus. Hier können Aufgaben wie Speichern, Laden, Suchen und Ersetzen, ... durchgeführt werden.

### Einfüge-Modus

- ➔ Im Einfüge-Modus kann man wie gewohnt Text schreiben. Durch Drücken der **i-Taste** gelangt man vom Kommando-Modus in den Einfüge-Modus. Über die **Esc-Taste** gelangt man wieder in den Kommando-Modus.

### Visueller Modus

- ➔ Text markieren an dem Operationen vorgenommen werden sollen.



## Editierbefehle

Textobjekt	Ändern	Löschen	Kopieren
1 Wort	<code>cw</code>	<code>dw</code>	<code>yw</code>
2 Wörter, Satzzeichen zählen nicht	<code>2cw</code> oder <code>c2w</code>	<code>2dw</code> oder <code>d2w</code>	<code>2yw</code> oder <code>y2w</code>
3 Wörter rückwärts	<code>3cb</code> oder <code>c3b</code>	<code>3db</code> oder <code>d3b</code>	<code>3yb</code> oder <code>y3b</code>
1 Zeile	<code>cc</code>	<code>dd</code>	<code>yy</code> oder <code>Y</code>
Bis Zeilenanfang	<code>c0</code>	<code>d0</code>	<code>y0</code>
Bis Zeilenende	<code>c\$</code> oder <code>C</code>	<code>d\$</code> oder <code>D</code>	<code>y\$</code>
Einzelne Zeichen	<code>r</code>	<code>x</code> oder <code>X</code>	<code>yl</code> oder <code>yh</code>
5 Zeichen	<code>5s</code>	<code>5x</code>	<code>5yl</code>



## Bewegen in der Datei

Bewegung	Befehle
links, unten, oben, rechts	h, j, k, l
Zum ersten Zeichen der folgenden Zeile	+
Zum ersten Zeichen der vorangegangenen Zeile	-
An das Wortende (ohne/mit Satzzeichen)	e/E
Wortweise vorwärts (ohne/mit Satzzeichen)	w/W
Wortweise rückwärts (ohne/mit Satzzeichen)	b/B
An den Zeilenanfang	0
An das Zeilenende	\$
Zur letzten Zeile	G
Zur ersten Zeile (Window)	H
Zur letzten Zeile (Window)	L
Eine Bildschirmseite auf	Strg-b
Eine Bildschirmseite ab	Strg-f



## Weitere Operationen

Operationen	Befehle
Text aus Register einfügen nachdem er vorher kopiert oder gelöscht wurde	(Paste) P über dem Cursor einfügen oder p unter den Cursor einfügen
Visueller Modus (visual mode) eignet sich gut zum Markieren	v



## Bearbeiten von Text

Editierarbeit	Befehl
Text an der aktuellen Position einfügen	i
Text am Zeilenanfang einfügen	I
Text an der aktuellen Position anfügen	a
Text am Zeilenende anfügen	A
Unterhalb des Cursors eine neue Zeile einfügen	o
Oberhalb des Cursors eine neue Zeile einfügen	O
Zeile löschen und Text ersetzen	S
Vorhandene Zeichen mit neuem Text überschreiben	R
Aktuelle und folgende Zeile zusammenführen	J
Groß-/Kleinschreibung umsetzen	~
Letzte Aktion wiederholen	.
Letzte Änderung zurücknehmen	u
Zeile in ihren ursprünglichen Zustand zurückversetzen	U





# Markierungen und Marken

## Markierungsbefehle

Befehl	Bedeutung
mx	Markiert aktuelle Position des Cursors mit x
'x	Bewegt den Cursor zu dem Zeichen, das mit x markiert wurde
''	Zurück zu der Stelle, an der der Cursor vor dem letzten Bewegungsbefehl stand
'	Zurück zum ersten Zeichen der Zeile, in der der Cursor vor dem letzten Bewegungsbefehl stand
:marks	Listet alle Marken auf (Kommando-Modus)

## Marken

Marken	Bedeutung
a-z	Lokal gültige Marken
A-Z	Global gültige Marken (Zusätzlich an den Dateinamen gebunden)
0-9	In diesen Marken werden die Cursorposition und der Dateiname beim Beenden der letzten VIM-Sitzung gespeichert

Um eine Markierung zu setzen, drücken Sie (im Kommandomodus) die Zeichenfolge mx, wobei x ein beliebiger Kleinbuchstabe ist. Um später zu dieser Markierung zu springen, genügt ein (wieder im Kommandomodus) eingegebenes 'x, wobei x wieder der gleiche Buchstabe ist. Sie können für Markierungen entweder Buchstaben (a-z und A-Z) oder ganze Wörter verwenden. Diese Markierungen können auch überall dort verwendet werden, wo Zeilenangaben auftreten.



## vi aufrufen / beenden

### Aufrufoptionen

Option	Bedeutung
<code>+n datei</code>	Öffnet datei bei der Zeilennummer <code>n</code>
<code>+ datei</code>	Öffnet datei bei der letzten Zeile
<code>+/muster datei</code>	Öffnet datei beim ersten Auftreten von <code>muster</code>
<code>+cmd</code>	Das Ex-Kommando wird nach dem Start ausgeführt

### Beenden

Option	Bedeutung
Änderungen speichern, Editor verlassen	<code>:wq</code> (auch <code>ZZ</code> oder <code>:x</code> )
Datei wurde nicht verändert, Editor verlassen	<code>:q</code>
Datei wurde verändert, Editor ohne speichern verlassen	<code>:q!</code>



# Übungen

- (1) Kopieren Sie die Passwort-Datei in Ihr Home-Verzeichnis. Öffnen Sie die Datei in vim. Setzen Sie eine Markierung auf die Zeile mit ihrem User und eine zweite auf root. Springen Sie zwischen diesen Markierungen hin und her.
- (2) Öffnen Sie die Passwort-Datei so, dass der Cursor sofort am Beginn der Zeile mit Ihrem User-Namen steht.
- (3) Löschen Sie die ersten 5 Zeilen mit einer Anweisung. Fügen Sie die gelöschten Zeilen am Dateiende wieder ein.
- (4) Suchen Sie den String „nobody“ im Text. Gehen Sie ans Wortende und löschen die restliche Zeile bis zum Ende. Machen Sie diesen Vorgang wieder Rückgängig.
- (5) Mit welcher Methode erzeugen Sie am schnellsten 23 mal die Zeile: „VI is NOT for Dummies“?
- (6) Fügen Sie am Ende der Datei das aktuelle Datum (date) ein ohne vi zu verlassen.
- (7) Aktivieren Sie den Visuellen Modus. Kopieren Sie die ersten 3 Zeilen und fügen Sie diese VOR der letzten Zeile wieder ein.



## Übungen (2)

- (1) Ersetzen Sie alle Vorkommen von „var“ im Text durch „VAR“. Damit
- (2) Was bewirkt folgende Anweisung:  
1,\$s/home/HOME/gc ? [Alternative:] %s/home/HOME/gc
- (3) Untersuchen Sie die vim-Konfigurationsdateien /etc/vim/vimrc und /etc/vim/gvimrc
- (4) ....



## Tipps & Links

- Es gibt ein Lernprogramm für vim. Aufruf: vimtutor
- [Best of VIM Tipps](#)
- [vim.org](http://vim.org)
- [Vim CookBook](#)
- [www.jerrywang.net/vi](http://www.jerrywang.net/vi) (Tutorial)
- **Anmerkung:** Vim war Gewinner des Readers' Choice Awards im Linux Journal in der Kategorie „Favorite Text Editor“ in den Jahren 2001 bis 2005 und erhielt 2000 den Slashdot Beanie Award als „Best Open Source Text Editor“ sowie 1999 den Linuxworld Editors' Choice Award.