

# Kurzbeschreibung Editor *vi/vim*

## 1. Vorbemerkungen und Definitionen:

Das **aktuelle Zeichen** ist das Zeichen, wo der Cursor steht.

Die **aktuelle Zeile** ist die Zeile, in der der Cursor steht.

<code>&lt;ret&gt;</code>	bedeutet:	Betätigung der Taste "Return"
<code>&lt;Strg/f&gt;</code>	" :	" " Tasten "Strg" und "f" gleichzeitig
<code>[option]</code>	" :	wahlfreie Zusatzangabe
<i>Datei</i>	" :	vom Benutzer anzugeben (kursiv markiert)

Es gibt zwei Modi: **Befehlsmodus** und **Einfügemodus**.

Nach dem Aufruf befindet man sich im Befehlsmodus.

Mit einem Befehl zum Einfügen oder Ersetzen von Text wechselt man in den Einfügemodus (es werden keine Befehle erkannt!). Mit `<ESC>` beendet man den Einfügemodus und landet wieder im Befehlsmodus.

## 2. Aufruf und Beenden:

**Aufruf:** `vi(m) [-options] Dateiname-` Angaben zu *options* in der unter 5. angeführten Literatur

**Beenden:** `ZZ` oder `:wq<return>`

### 3.1 Bildschirmsteuerung / Umblättern

<code>&lt;Strg/d&gt;</code> , <code>&lt;Strg/u&gt;</code>	- eine halbe Bildschirmseite vor ( <u>down</u> ), zurück ( <u>up</u> )
<code>&lt;Strg/f&gt;</code> , <code>&lt;Strg/b&gt;</code>	- eine Bildschirmseite vor ( <u>forward</u> ), zurück ( <u>backward</u> )
<code>&lt;Strg/g&gt;</code>	- die aktuelle Zeilennummer anzeigen

### 3.2 Einfügen von Text (=Umschalten in den Einfügemodus, Beenden mit `<ESC>`)

<code>a</code>	- hinter dem Cursor einfügen ( <u>append</u> )	! Beenden mit <code>&lt;ESC&gt;</code>
<code>A</code>	- am Ende der Zeile anfügen	! Beenden mit <code>&lt;ESC&gt;</code>
<code>i</code>	- vor dem Cursor einfügen ( <u>insert</u> )	! Beenden mit <code>&lt;ESC&gt;</code>
<code>I</code>	- am Anfang der Zeile einfügen ( <u>Insert</u> )	! Beenden mit <code>&lt;ESC&gt;</code>
<code>&lt;ESC&gt;</code>	- Einfügemodus verlassen	

### 3.3 Löschen von Text (teils mit Ablage in den Zwischenspeicher)

<code>dw</code> , <code>dW</code>	- das aktuelle Wort (nur Buchstaben), (alle Zeichen) löschen ( <u>delete word</u> )
<code>ndw</code> , <code>ndW</code>	- <i>n</i> Worte (nur Buchstaben), (alle Zeichen) löschen
<code>dd</code> , <code>ndd</code>	- die aktuelle Zeile, <i>n</i> Zeilen löschen
<code>dnd</code>	- <i>n</i> Zeilen löschen und in den Zwischenspeicher legen
<code>d\$</code>	- die akt. Zeile ab Cursor bis Zeilenende löschen
<code>x</code> , <code>X</code>	- das aktuelle Zeichen löschen, das Zeichen vor dem Cursor löschen

### 3.4 Ändern/Ersetzen von Text

<code>cw</code> , <code>cW</code>	- das aktuelle Wort (nur Buchstaben), (alle Zeichen) ab Cursor ersetzen durch neue Eingabe ! Beenden mit <code>&lt;ESC&gt;</code>
<code>C</code> oder <code>c\$</code>	- die aktuelle Zeile ab Cursor bis Zeilenende ersetzen durch neue Eingabe ! Beenden mit <code>&lt;ESC&gt;</code>
<code>J</code> , <code>nJ</code>	- Zeilen verknüpfen ( <u>Join</u> aktuelle und nächste Zeile), aktuelle und <i>n-1</i> nächste Zeilen
<code>p</code> , <code>P</code>	- Inhalt des Zwischenspeichers hinter, vor der aktuellen Zeile einfügen ( <u>put</u> )
<code>r&lt;Neuzeichen&gt;</code>	- das aktuelle Zeichen ersetzen durch <code>&lt;Neuzeichen&gt;</code>
<code>Y</code> , <code>nY</code>	- die aktuelle Zeile, <i>n</i> Zeilen in den Zwischenspeicher laden ( <u>yank</u> )
<code>yw</code>	- das aktuelle Wort in den Zwischenspeicher laden ( <u>yank word</u> )
<code>~</code>	- Groß/Kleinschreibung des aktuellen Zeichens ändern

### 3.5 Suchen von Zeichen/Zeichenketten

<code>/happy day</code>	- nach <i>happy day</i> vorwärts suchen
<code>n</code>	- nächstes Vorkommen suchen (Wiederholung des vorherigen Suchens)
<code>N</code>	- nächstes Vorkommen suchen mit Richtungsumkehr
<code>?happy day</code>	- nach <i>happy day</i> rückwärts suchen
<code>%</code>	- sucht Gegenstück von ( ) und [ ] - Paaren

### 3.6 Cursor positionieren

Cursortasten	- wie beschriftet (one Position) right/left/up/down
+, <ret>	- zum Anfang der nächsten Zeile
-	- zum Anfang der vorherigen Zeile
w, W	- zum Anfang des nächsten Wortes (nur Buchstaben), (alle Zeichen)
0 oder ^, \$	- zum Anfang, zum Ende der akt. Zeile
nG,	- zum Anfang der n. Zeile der Datei; Vorgabe: n=letzte Zeile
1G	- zum Anfang der 1. Zeile der Datei

### 3.7 Verschiedenes

<Strg/v><Strg/m>	- CTRL-Zeichen als String eingeben, hier STRG/M oder ^M = Carriage Return;
<Esc>	- Beenden einer Eingabe
u	- letzten Befehl zurück (undo)
U	- alle Änderungen in der akt. Zeile zurück (Undo)
<delete>	- bei der Eingabe 1 Zeichen zurück
<ctrl/w>	- bei der Eingabe 1 Wort zurück

### 3.8 ex - Befehle

Einleitung mit : ==> der Cursor springt in die Kommandozeile (letzte Bildschirmzeile)

:n	- zur n-ten Zeile springen
:\$	- zur letzten Zeile springen
:s/string1/string2/	- substitute; das 1. Vorkommen von <i>string1</i> in der akt. Zeile in <i>string2</i> ändern
:1,\$s/string1/string2/g	- ersetze alle Vorkommen von <i>string1</i> jeder Zeile durch <i>string2</i> , von der 1. bis letzten (= \$) Zeile (substitute)
:2,5d	- die 2. bis 5. Zeile löschen
:\$d	- die aktuelle bis letzte Zeile löschen
:.+1j	- die aktuelle und nächste Zeilen verbinden (Join)
:1,\$w file1	- write; erste bis letzte Zeile in die Datei <i>file1</i> schreiben

### 3.9 Setzen von Optionen

Ständig gewünschte Werte können in der Datei .login definiert werden z.B. **setenv EXINIT 'set wm=0'** oder in der neu anzulegenden Datei .exrc mit einem set-Befehl wie u.a. pro Zeile (ohne ":").

:set (no)option	- eine option setzen (rücksetzen); (beachte: nach <i>no</i> kein Leerzeichen!)
:set	- Anzeige der gesetzten Optionen
:set all	- Anzeige aller möglichen Optionen
:set number	- Zeilennummerierung ein(aus)schalten
:set showmode	- Rechts unten Anzeige "Input Mode" bei Einfügemodus
:set showmatch	- Anzeigen der zugehörigen "(" bei Eingabe einer ")"

### Lesen und Abspeichern von Dateien sowie Beenden

:r <i>statis.c</i>	- die Datei <i>statis.c</i> einlesen (read) hinter die aktuelle Zeile
:w	- den Pufferinhalt in die aktuelle Datei abspeichern (write)
:w <i>statis2.c</i>	- den Pufferinhalt in die Datei <i>statis2.c</i> abspeichern
:4,13w <i>hilf1.txt</i>	- die Zeilen 4 bis 13 in die Datei <i>hilf1.txt</i> kopieren
:wq	- die Änderungen in der aktuellen Datei speichern und vi beenden (write and quit)
:q!	- vi beenden ohne Abspeichern der Änderungen.

<b>Beispiel I.</b> advm1>vi <i>brief1.txt</i> <ret>	- vi starten, um die neue Datei <i>brief1.txt</i> zu erstellen;
i	- Eingabemodus beginnen;
<i>Sehr geehrte ...</i> ,<ret>	- Text zeilenweise eingeben
.	
Mit freundlichen Grüßen<ret>	- letzte Textzeile
<ESC>	- Einfügemodus beenden
:wq	- Datei schließen und vi beenden (write and quit)

<b>Beispiel II.</b> advm1> vi <i>liste.script</i> <ret>	- vi starten, um die vorhandene Datei <i>liste.script</i> zu ändern (Ausgabe der 1. Seite der Datei; der Cursor steht in der 1. Zeile, 1. Spalte)
:1,\$s/<Strg/v><Strg/m>//g<ret>	- alle Vorkommen von ^M (Control/M=Carriage Return) löschen
ZZ	- die Datei <i>liste.script</i> schließen