

# GNU Emacs Referenzkarte

## Emacs Starten

Um GNU Emacs 19 zu starten, tippen Sie ein: **emacs**

Um eine Datei fuer Editieren zu laden, lesen Sie unten weiter.

## Emacs Verlassen

Emacs pausieren lassen **C-z**  
Emacs beenden **C-x C-c**

## Dateien

Datei **oeffnen** **C-x C-f**  
Datei **sichern** **C-x C-s**  
**alle** Dateien sichern **C-x s**  
den Inhalt einer anderen Datei **ein fuegen** **C-x i**  
diese Datei durch eine andere ersetzen **C-x C-v**  
Datei neu anlegen und speichern **C-x C-w**  
version control ein/auschecken **C-x C-q**

## Hilfe

Das Hilfesystem ist einfach zu bedienen. Tippen Sie **C-h** (oder **F1**). Neulinge tippen **C-h t** um ein **tutorial** zu starten.

Hilfe Fenster entfernen **C-x 1**  
Hilfe Fenster scrollen **C-M-v**  
apropos: zeigt alle Befehle mit dem Muster **C-h a**  
zeigt die Funktion mit dieser Tastenkomb. **C-h c**  
beschreibt die Funktion **C-h f**  
modusspezifische Information **C-h m**

## Was tun wenn nichts mehr geht

**Abbrechen** eines Vorgangs **C-g**  
**Wiederherstellung** von Dateien **M-x recover-file**  
**rueckgaengig** machen **C-x u**  
Puffer in Ursprungszustand bringen **M-x revert-buffer**  
Bildschirmanzeige in Ordnung bringen **C-l**

## Inkrementelle Suche

Suche vorwaerts **C-s**  
Suche rueckwaerts **C-r**  
Suche mit regularen Ausdruecken **C-M-s**  
Rueckwaertssuche mit reg. Ausdruecken **C-M-r**  
letzten Suchausdruck auswahlen **M-p**  
spaelteren Suchausdruck auswahlen **M-n**  
inkrementelle Suche beenden **RET**  
ein Suchzeichen zurueckgehen **DEL**  
Suche abbrechen **C-g**

Wiederholtes Druecken von **C-s** oder **C-r** sucht weitere Treffer.  
Wenn Emacs sucht, unterbricht **C-g** nur die jeweils letzte Suche.

© 1996 Free Software Foundation, Inc. Permissions on back. v2.1

## Cursor Bewegung

Textteile ueberspringen	rueckw.	vorw.
Zeichen	<b>C-b</b>	<b>C-f</b>
Wort	<b>M-b</b>	<b>M-f</b>
Zeile	<b>C-p</b>	<b>C-n</b>
zum Zeilenende springen	<b>C-a</b>	<b>C-e</b>
Satz	<b>M-a</b>	<b>M-e</b>
Paragraph	<b>M-{</b>	<b>M-}</b>
Seite	<b>C-x [</b>	<b>C-x ]</b>
Lisp-s-expression	<b>C-M-b</b>	<b>C-M-f</b>
Funktion	<b>C-M-a</b>	<b>C-M-e</b>
zum Pufferanfang (oder Ende)springen	<b>M-&lt;</b>	<b>M-&gt;</b>
eine Bildschirmseite runter scrollen	<b>C-v</b>	
eine Bildschirmseite hoch scrollen	<b>M-v</b>	
nach links scrollen	<b>C-x &lt;</b>	
nach rechts scrollen	<b>C-x &gt;</b>	
Cursorzeile in die Bildschirmmitte scrollen	<b>C-u C-l</b>	

## Loeschen

Textteile	rueckwaerts	vorwaerts
Zeichen (loeschen)	<b>DEL</b>	<b>C-d</b>
Wort	<b>M-DEL</b>	<b>M-d</b>
Zeile (bis zum Ende)	<b>M-0 C-k</b>	<b>C-k</b>
Satz	<b>C-x DEL</b>	<b>M-k</b>
Lisp-s-expression	<b>M-- C-M-k</b>	<b>C-M-k</b>
<b>Bereiche</b> loeschen		<b>C-w</b>
Bereich in die Ablage kopieren		<b>M-w</b>
Bis zum <i>Zeichen</i> loeschen		<b>M-z Zeichen</b>
letztes geloeschtes Zeichen einsetzen		<b>C-y</b>
eins davor einsetzen		<b>M-y</b>

## Markieren

Marke setzen	<b>C-@</b> or <b>C-SPC</b>
zwischen Cursor und Marke wechseln	<b>C-x C-x</b>
Marke <i>Argument</i> <b>Worte</b> entfernt setzen	<b>M-@</b>
<b>Paragraph</b> markieren	<b>M-h</b>
<b>Seite</b> markieren	<b>C-x C-p</b>
<b>Lisp-s-expression</b> markieren	<b>C-M-@</b>
<b>Funktion</b> markieren	<b>C-M-h</b>
den ganzen <b>Puffer</b> markieren	<b>C-x h</b>

## Interaktives Ersetzen

Zeichenkette interaktiv ersetzen	<b>M-%</b>
mit regularen Ausdruecken	<b>M-x query-replace-regexp</b>
Moegliche Antworten in diesem Modus:	
dies <b>ersetzen</b> und zum naechsten gehen	<b>SPC</b>
dies ersetzen	<b>,</b>
dies <b>ueberspringen</b> , zum naechsten gehen	<b>DEL</b>
alle verbleibenden Treffer ersetzen	<b>!</b>
eine Ersetzung <b>rueckgaengig</b> machen	<b>~</b>
interaktiven Modus <b>verlassen</b>	<b>RET</b>
rekursiven Modus starten ( <b>C-M-c</b> verlassen)	<b>C-r</b>

## Mehrere Fenster

Die zweite Tastenk. bezieht sich immer auf das andere Fenster:

alle anderen Fenster in den Hintergrund	C-x 1	
Fenster vertikal teilen	C-x 2	C-x 5 2
dieses Fenster loeschen	C-x 0	C-x 5 0
Fenster horizontal teilen	C-x 3	
das andere Fenster scrollen	C-M-v	
ins andere Fenster wechseln	C-x o	C-x 5 o
Puffer in ein anderes Fenster bringen	C-x 4 b	C-x 5 b
Puffer in einem anderen Fenster darstellen	C-x 4 C-o	C-x 5 C-o
Datei in ein anderes Fenster oeffnen	C-x 4 f	C-x 5 f
Datei im Ansichtmodus in anderem Fenster oeffnen	C-x 4 r	C-x 5 r
Dired in einem anderen Fenster oeffnen	C-x 4 d	C-x 5 d
Tag in einem anderen Fenster finden	C-x 4 .	C-x 5 .
Fenster vergroessern	C-x ^	
Fenster schmaeler machen	C-x {	
Fenster breiter machen	C-x }	

## Formattierung

<b>Zeile</b> (modusabhaengig) einruecken	TAB
<b>Bereich</b> (modusabh.) einruecken	C-M-\
<b>Lisp-s-expression</b> (modusabh.) einruecken	C-M-q
Bereich <i>Argument</i> Spalten einruecken	C-x TAB
Zeilenumbruch nach Cursor einfuegen	C-o
Zeilenrest vertikal nach unten verschieben	C-M-o
Leerzeilen um Cursorposition loeschen	C-x C-o
Zeile mit letzter verbinden (Arg. naechster)	M-^
Leerzeichen an Cursorposition loeschen	M-\
ein Leerzeichen an Cursorposition setzen	M-SPC
Paragraph auffuellen	M-q
Fuell Spalte setzen	C-x f
Praefix setzen fuer jede Zeile	C-x .
Zeichendarstellung setzen	M-g

## Gross-Kleinbuchstaben

Wort in Grossbuchstaben	M-u
Wort in Kleinbuchstaben	M-l
Word mit grossen Anfangsbuchstaben	M-c
Bereich in Grossbuchstaben	C-x C-u
Bereich in Kleinbuchstaben	C-x C-l

## Der Minipuffer

Die folgenden Tastenkombination gelten im Minipuffer:

so viel wie moeglich ergaenzen	TAB
ein Wort ergaenzen	SPC
ergaenzen und ausfuehren	RET
moegliche Ergaenzungen zeigen	?
letzte Eingabe wiederanzeigen	M-p
spaelere Eingabe wiederanzeigen	M-n
reg. Ausd. rueckwaerts in History suchen	M-r
reg. Ausd. vorwaerts in History suchen	M-s
Vorgang unterbrechen	C-g

Tippen Sie **C-x ESC ESC** um den letzten Befehl zu editieren und zu wiederholen der im Minipuffer ausgefuehrt wurde.

# GNU Emacs Referenzkarte

## Puffer

anderen Puffer auswahlen	C-x b
alle Puffer anzeigen	C-x C-b
Puffer loeschen	C-x k

## Vertauschen

<b>Zeichen</b> vertauschen	C-t
<b>Worte</b> vertauschen	M-t
<b>Zeilen</b> vertauschen	C-x C-t
<b>Lisp-s-expressions</b> vertauschen	C-M-t
C-x 5 .	

## Rechtschreibkorrektur

Ueberpruefe aktuelles Wort	M-\$
Ueberpruefe alle Woerter in Bereich	M-x ispell-region
Ueberpruefe den gesamten Bereich	M-x ispell-buffer

## Tags

Tag finden (Definition)	M-.
Naechstes Vorkommen von Tag finden	C-u M-.
Neue Tags Datei angeben	M-x visit-tags-table
Regulaere Ausdruck Suche in Dateien	M-x tags-search
Interakt. Ersetzen in allen Dateien	M-x tags-query-replace
Letzte Tag Suche oder Ersetzen nochmal	M-,

## Shells

Shell Kommando ausfuehren	M-!
Shell Kommando fuer bereich ausfuehren	M-
Bereich durch Shell Kommando filtern	C-u M-
Shell im Fenster <i>*shell*</i> starten	M-x shell

## Rechtecke

Kopiere Rechteck in Register	C-x r r
Loesche Rechteck	C-x r k
Rechteck einsetzen	C-x r y
Rechteck aufmachen, Text nach rechts	C-x r o
Rechteck mit Leerzeichen ueberschreiben	C-x r c
Praefix vor jede Zeile setzen	C-x r t

## Abkuerzungen

globale Abkuerzung hinzufuegen	C-x a g
modusabhaengige Abkuerzung hinzufuegen	C-x a l
globale Expansion fuer Abk. definieren	C-x a i g
modusabhaengige Abkuerzung definieren	C-x a i l
explizites Expandieren	C-x a e
letztes Wort dynamisch expandieren	M-/

## Regulaere Ausdruecke

jedes Zeichen ausser Zeilenumbruch	.	(Punkt)
Null oder mehr Wiederholungen	*	
Eine oder mehr Wiederholungen	+	
Null oder eine Wiederholung	?	
jedes Zeichen in der Menge	[ ... ]	
jedes Zeichen nicht in der Menge	[^ ... ]	
Zeilenanfang	^	
Zeilenende	\$	
spezielles Zeichen maskieren <i>c</i>	\c	
Alternative ("oder")	\	
Gruppe	\( ... \)	
<i>n</i> te Gruppe	\n	
Pufferanfang	\'	
Pufferende	\'	
Wortzwischenraum	\b	
Weder Anfang noch Ende eines Wortes	\B	
Wortanfang	\<	
Wortende	\>	
jedes Wort-Syntax Zeichen	\w	
jedes Nicht-Wort-Syntax Zeichen	\W	
Zeichen mit Syntax <i>c</i>	\sc	
Zeichen nicht mit Syntax <i>c</i>	\Sc	

## Register

Region in Register speichern	C-x r s
Register Inhalt in Puffer einfuegen	C-x r i
Cursorposition in Register speichern	C-x r SPC
Springe zur abgespeicherten Position	C-x r j

## Info

Info starten	C-h i	
Bewegung innerhalb eines Knotens:		
vorwaerts scrollen	SPC	
rueckwaerts scrollen	DEL	
zum Anfang eines Knotens	.	(dot)
Bewegung zwischen Knoten:		
<b>naechster</b> Knoten	n	
<b>vorheriger</b> Knoten	p	
nach <b>oben</b>	u	
Menue Element ueber Namen auswaelen	m	
<i>n</i> ten Menueeintrag auswaelen (1-9)	n	
Kreuzverweis folgen (zurueck mit 1)	f	
zurueck zum letzten gesehenen Knoten	l	
zurueck zum Verzeichnisknoten	d	
Knoten ueber Namen auswaelen	g	
Sonstige:		
Info <b>Tutorial</b> starten	h	
Info Befehle zeigen	?	
Info <b>verlassen</b>	q	
Knoten nach reg. Ausd. durchsuchen	M-s	

## Tastatur Makros

Tastatur Makro Definition <b>starten</b>	C-x (
Tastatur Makro Definition <b>beenden</b>	C-x )
zuletzt definiertes Tast. Makro <b>ausfuehren</b>	C-x e
an letztes Tastatur Makro anhaengen	C-u C-x (
letztes Tastatur Makro benennen	M-x name-last-kbd-macro
Lisp Definition in Puffer einfuegen	M-x insert-kbd-macro

## Kommandos fuer Emacs Lisp

Lisp-s-expression vor Cursor laden	C-x C-e
aktuelle <b>Definition</b> auswerten	C-M-x
<b>Bereich</b> auswerten	M-x eval-region
gesamten <b>Puffer</b> auswerten	M-x eval-current-buffer
Lisp Ausdruck im Minipuffer auswerten	M-:
letztes Minipufferkommando auswerten	C-x ESC ESC
Emacs Lisp Datei lesen und auswerten	M-x load-file
aus Standard Systemverzeichnis laden	M-x load-library

## Einfaches Konfigurieren

Ein Beispiel dafuer, wie man Tastenkombinationen definiert:

```
(global-set-key "\C-cg" 'goto-line)
(global-set-key "\C-x\C-k" 'kill-region)
(global-set-key "\M-#" 'query-replace-regexp)
```

So weist man in Emacs Lisp einer Variablen Werte zu:

```
(setq backup-by-copying-when-linked t)
```

## Selbst Kommandos schreiben

```
(defun Commando-Name (args)
  "Documentation"
  (interactive "template")
  body)
```

Ein Beispiel:

```
(defun diese-Zeile-zum-Fensteranfang (Zeile)
  "Zeile an Cursorposition zum Fensteranfang bewegen"
  Mit numerischem Argument n, zur Zeile n
  Mit negativem Argument zum Fensterende
  (interactive "P")
  (recenter (if (null Zeile)
                0
                (prefix-numeric-value Zeile))))
```

Das Argument fuer **interactive** ist eine Zeichenkette, die spezifiziert, wie die Argumente bereitgestellt werden, wenn die Funktion interaktiv aufgerufen wird. C-h f **interactive** fuer mehr Informationen.

Copyright © 1996 Free Software Foundation, Inc.  
designed by Stephen Gildea, March 1996 v2.1  
for GNU Emacs version 19 on Unix systems

Permission is granted to make and distribute copies of this card provided the copyright notice and this permission notice are preserved on all copies.