

Referenzkarte zu GNU Emacs

(für Version 30)

Emacs starten

Um GNU Emacs 30 zu starten, geben Sie `emacs` ein.

Emacs verlassen

Emacs unterbrechen (unter X: minimieren) `C-z`
Emacs beenden `C-x C-c`

Dateien

Datei öffnen `C-x C-f`
Datei speichern `C-x C-s`
alle Dateien speichern `C-x s`
den Inhalt einer anderen Datei einfügen `C-x i`
diese Datei durch eine andere ersetzen `C-x C-v`
Puffer in neuer Datei speichern `C-x C-w`
Nur-Lesen-Modus des Puffers wechseln `C-x C-q`

Hilfe

Das Hilfesystem ist einfach zu bedienen. Drücken Sie `C-h` (oder `F1`). Neulinge drücken `C-h t`, um eine **Einführung** zu starten.

Hilfefenster entfernen `C-x 1`
Hilfefenster scrollen `C-M-v`
Befehle zeigen, die Zeichenkette enthalten `C-h a`
Tastenkombination beschreiben `C-h k`
eine Funktion beschreiben `C-h f`
modusspezifische Informationen `C-h m`

Fehlerbehandlung

Abbrechen eines Vorgangs `C-g`
Wiederherstellung von Dateien `M-x recover-session`
Änderungen rückgängig machen `C-x u, C-_ oder C-/`
Puffer in Ursprungszustand bringen `M-x revert-buffer`
Bildschirmanzeige in Ordnung bringen `C-l`

Inkrementelle Suche

Suche vorwärts `C-s`
Suche rückwärts `C-r`
Suche mit regulären Ausdrücken `C-M-s`
Rückwärtssuche mit regulären Ausdrücken `C-M-r`
früheren Suchausdruck auswählen `M-p`
späteren Suchausdruck auswählen `M-n`
inkrementelle Suche beenden `RET`
ein Suchzeichen zurückgehen `DEL`
Suche abbrechen `C-g`
Wiederholtes Drücken von `C-s` oder `C-r` sucht weitere Treffer.
Wenn Emacs sucht, unterbricht `C-g` nur die jeweils letzte Suche.

Cursor-Bewegung

Textteile überspringen	rückw.	vorw.
Zeichen	<code>C-b</code>	<code>C-f</code>
Wort	<code>M-b</code>	<code>M-f</code>
Zeile	<code>C-p</code>	<code>C-n</code>
zum Zeilenanfang (oder -ende) springen	<code>C-a</code>	<code>C-e</code>
Satz	<code>M-a</code>	<code>M-e</code>
Absatz	<code>M-{</code>	<code>M-}</code>
Seite	<code>C-x [</code>	<code>C-x]</code>
Lisp-s-expression	<code>C-M-b</code>	<code>C-M-f</code>
Funktion	<code>C-M-a</code>	<code>C-M-e</code>
zum Pufferanfang (oder -ende) springen	<code>M-<</code>	<code>M-></code>
eine Bildschirmseite herunter scrollen		<code>C-v</code>
eine Bildschirmseite hoch scrollen		<code>M-v</code>
nach links scrollen		<code>C-x <</code>
nach rechts scrollen		<code>C-x ></code>
Cursor-Zeile in die Bildschirmmitte scrollen	<code>C-u</code>	<code>C-l</code>

Löschen und Ausschneiden

Textteile ausschneiden	rückw.	vorw.
Zeichen (löschen, nicht ausschneiden)	<code>DEL</code>	<code>C-d</code>
Wort	<code>M-DEL</code>	<code>M-d</code>
Zeile (bis zum Ende)	<code>M-O C-k</code>	<code>C-k</code>
Satz	<code>C-x DEL</code>	<code>M-k</code>
Lisp-s-expression	<code>M-- C-M-k</code>	<code>C-M-k</code>
Bereich ausschneiden		<code>C-w</code>
Bereich in die Ablage kopieren		<code>M-w</code>
bis zum Zeichen ausschneiden		<code>M-z Zeichen</code>
zuletzt ausgeschnittenen Text einfügen		<code>C-y</code>
vorher ausgeschnittenen Text einfügen		<code>M-y</code>

Markieren

Marke setzen	<code>C-@ oder C-SPC</code>
Cursor und Marke austauschen	<code>C-x C-x</code>
Marke Argument Worte entfernt setzen	<code>M-@</code>
Absatz markieren	<code>M-h</code>
Seite markieren	<code>C-x C-p</code>
Lisp-s-expression markieren	<code>C-M-@</code>
Funktion markieren	<code>C-M-h</code>
den ganzen Puffer markieren	<code>C-x h</code>

Interaktives Ersetzen

Zeichenkette interaktiv ersetzen mit regulären Ausdrücken	<code>M-%</code>
	<code>C-M-%</code>
Mögliche Antworten in diesem Modus:	
dies ersetzen und zum nächsten gehen	<code>SPC</code>
dies ersetzen und nicht weitergehen	<code>,</code>
dies überspringen, zum nächsten gehen	<code>DEL</code>
alle verbleibenden Treffer ersetzen	<code>!</code>
zum vorherigen Treffer zurückgehen	<code>^</code>
interaktives Ersetzen beenden	<code>RET</code>
rekursives Editieren starten (<code>C-M-c</code> beendet)	<code>C-r</code>

Mehrere Fenster

Wenn zwei Befehle angezeigt werden, ist der zweite ein ähnlicher für einen Rahmen statt eines Fensters.

alle anderen Fenster schließen	<code>C-x 1</code>	<code>C-x 5 1</code>
Fenster vertikal teilen	<code>C-x 2</code>	<code>C-x 5 2</code>
dieses Fenster schließen	<code>C-x 0</code>	<code>C-x 5 0</code>
Fenster horizontal teilen		<code>C-x 3</code>
das andere Fenster scrollen		<code>C-M-v</code>
in anderes Fenster wechseln	<code>C-x o</code>	<code>C-x 5 o</code>
Puffer in and. Fenster auswählen	<code>C-x 4 b</code>	<code>C-x 5 b</code>
Puffer in anderem Fenster anzeigen	<code>C-x 4 C-o</code>	<code>C-x 5 C-o</code>
Datei in anderem Fenster öffnen	<code>C-x 4 f</code>	<code>C-x 5 f</code>
Datei in anderem Fenster anzeigen	<code>C-x 4 r</code>	<code>C-x 5 r</code>
Dired in anderem Fenster starten	<code>C-x 4 d</code>	<code>C-x 5 d</code>
Tag in anderem Fenster finden	<code>C-x 4 .</code>	<code>C-x 5 .</code>
Fenster vergrößern		<code>C-x ^</code>
Fenster verengen		<code>C-x {</code>
Fenster verbreitern		<code>C-x }</code>

Formatierung

Zeile (modusabhängig) einrücken	<code>TAB</code>
Bereich (modusabhängig) einrücken	<code>C-M-\</code>
Lisp-s-expression (modusabh.) einrücken	<code>C-M-q</code>
Bereich Argument Spalten einrücken	<code>C-x TAB</code>
Zeilenumbruch nach Cursor einfügen	<code>C-o</code>
Zeilenrest vertikal nach unten verschieben	<code>C-M-o</code>
Leerzeilen um Cursor-Position löschen	<code>C-x C-o</code>
Zeile mit voriger verbinden (Arg. nächste)	<code>M-^</code>
alle Leerzeichen um Cursor-Position löschen	<code>M-\</code>
genau ein Leerzeichen an Cursor-Position	<code>M-SPC</code>
Absatz auffüllen	<code>M-q</code>
Spalte für Umbruch auf Argument setzen	<code>C-x f</code>
Präfix für jede Zeile setzen	<code>C-x .</code>

Groß- und Kleinschreibung

Wort in Großbuchstaben	<code>M-u</code>
Wort in Kleinbuchstaben	<code>M-l</code>
Wort mit großem Anfangsbuchstaben	<code>M-c</code>
Bereich in Großbuchstaben	<code>C-x C-u</code>
Bereich in Kleinbuchstaben	<code>C-x C-l</code>

Der Minipuffer

Die folgenden Tastenkombinationen gelten im Minipuffer:

so weit wie möglich ergänzen	<code>TAB</code>
ein Wort ergänzen	<code>SPC</code>
ergänzen und ausführen	<code>RET</code>
mögliche Ergänzungen zeigen	<code>?</code>
letzte Eingabe zurückholen	<code>M-p</code>
später Eingabe zurückholen	<code>M-n</code>
reg. Ausd. rückwärts in History suchen	<code>M-r</code>
reg. Ausd. vorwärts in History suchen	<code>M-s</code>
Befehl abbrechen	<code>C-g</code>

Drücken Sie `C-x ESC ESC`, um den letzten Befehl zu bearbeiten und zu wiederholen, der im Minipuffer ausgeführt wurde. Drücken Sie `F10`, um die Menüzeile im Minipuffer zu aktivieren.

Referenzkarte zu GNU Emacs

Puffer

anderen Puffer auswählen
Liste aller Puffer anzeigen
einen Puffer schließen

C-x b
C-x C-b
C-x k

Vertauschen

Zeichen vertauschen
Worte vertauschen
Zeilen vertauschen
Lisp-s-expressions vertauschen

C-t
M-t
C-x C-t
C-M-t

Rechtschreibprüfung

aktueller Wort überprüfen
alle Wörter im Bereich überprüfen
gesamten Puffer überprüfen

M-\$
M-x ispell-region
M-x ispell-buffer

Tags

Tag finden (Definition)
neue Tagsdatei angeben
regulären Ausdruck in Dateien suchen
interakt. Ersetzen in allen Dateien

M-.
M-x visit-tags-table
M-x tags-search
M-x tags-query-replace

Shells

Shellbefehl ausführen
Shellbefehl für Bereich ausführen
Bereich durch Shellbefehl filtern
eine Shell im Fenster *shell* starten

M-!
M-|
C-u M-|
M-x shell

Rechtecke

Rechteck in Register kopieren
Rechteck ausschneiden
Rechteck einfügen
Rechteck öffnen, Text nach rechts
Rechteck mit Leerzeichen überschreiben
Präfix vor jede Zeile setzen

C-x r r
C-x r k
C-x r y
C-x r o
C-x r c
C-x r t

Abkürzungen

globale Abkürzung hinzufügen
modusabhängige Abkürzung hinzufügen
globalen Ersetzungstext für Abk. definieren
modusabh. Ersetzungstext für Abk. def.
Abkürzung explizit ausschreiben

C-x a g
C-x a l
C-x a i g
C-x a i l
C-x a e

letztes Wort dynamisch ausschreiben

M-/

Reguläre Ausdrücke

jedes einzelne Zeichen außer Zeilenumbruch
null oder mehr Wiederholungen
eine oder mehr Wiederholungen
null oder eine Wiederholung
Spezialzeichen c maskieren
Alternative („oder“)
Gruppe
gleicher Text wie nte Gruppe
Anfang oder Ende eines Wortes
weder Anfang noch Ende eines Wortes

. (Punkt)
*
+
?
\c
\l
\(... \)
\n
\b
\B

Einheit	passt am Anf.	passt am Ende
Zeile	^	\$
Wort	\<	\>
Puffer	\`	\'
Zeichenklasse	passt auf diese	passt auf and.
Explizite Menge	[...]	[^ ...]
Wortsyntax-Zeichen	\w	\W
Zeichen mit Syntax c	\sc	\Sc

Internationale Zeichensätze

Hauptsprache einstellen
Alle Eingabemethoden anzeigen
Eingabemethode in oder außer Kraft setzen
Kodierung für nächsten Befehl setzen
Alle Kodierungen anzeigen
bevorzugte Kodierung auswählen

C-x RET 1
M-x list-input-methods
C-\
C-x RET c
M-x list-coding-systems
M-x prefer-coding-system

Info

Info-Betrachter starten
Funktion oder Variable in Info finden
Bewegung innerhalb eines Knotens:

C-h i
C-h S

vorwärts scrollen
rückwärts scrollen
zum Anfang eines Knotens

SPC
DEL
. (Punkt)

Bewegung zwischen Knoten:

n

nächster Knoten
vorheriger Knoten
nach oben
Menüeintrag über Namen auswählen
nen Menüeintrag auswählen (1-9)
Querverweise folgen (zurück mit 1)
zurück zum letzten gesehenen Knoten
zurück zum Verzeichnisknoten
zum Anfangsknoten der aktuellen Datei
beliebigen Knoten über Namen auswählen

p

u
m
f
l
d
t
g

Sonstiges:

h
i
s
q

Einführung in Info starten
Begriff in den Indizes suchen
nach regulären Ausdrücken suchen
Info verlassen

Register

Region in Register speichern
Registerinhalt in Puffer einfügen
Cursor-Position in Register speichern
zu abgespeicherter Position springen

C-x r s
C-x r i
C-x r SPC
C-x r j

Tastaturmakros

Makrodefinition starten
Makrodefinition beenden
zuletzt definiertes Makro ausführen
an letztes Makro anhängen
letztes Makro benennen
Lispcode für Makro in Puffer einfügen

C-x ()
C-x)
C-x e
C-u C-x ()
M-x name-last-kbd-macro
M-x insert-kbd-macro

Befehle für Emacs-Lisp

Lisp-s-expression vor Cursor auswerten
aktuelle Definition auswerten
Bereich auswerten
Lisp-Ausdruck im Minipuffer auswerten
Datei aus Standardverzeichnis laden

C-x C-e
C-M-x
M-x eval-region
M-:
M-x load-library

Einfache Anpassungen

Variablen und Faces anpassen
Tastenkombinationen definieren (Beispiel):
(global-set-key (kbd "C-c g") 'search-forward)
(global-set-key (kbd "M-#") 'query-replace-regexp)

Eigene Befehle schreiben

```
(defun Befehlsname (Argumente)
  "Dokumentation"
  (interactive "Vorlage"
  Rumpf))
Ein Beispiel:
(defun diese-Zeile-zum-Fensteranfang (Zeile)
  "Zeile an Cursor-Position zum Fensteranfang bewegen.
Mit ARGUMENT, Cursor in Zeile ARGUMENT bewegen."
  (interactive "P")
  (recenter (if (null Zeile)
    0
    (prefix-numeric-value Zeile))))
```

Die Spezifikation zu interactive gibt an, wie die Argumente gelesen werden, wenn die Funktion interaktiv aufgerufen wird. Für nähere Details geben Sie C-h f interactive ein.

Copyright © 2025 Free Software Foundation, Inc.
Für GNU Emacs Version 30
Entworfen von Stephen Gildea

Released under the terms of the GNU General Public License version 3 or later.

For more Emacs documentation, and the TeX source for this card, see the Emacs distribution, or <https://www.gnu.org/software/emacs>