

# GNU Emacs Survival Card

for version 30

In the following, **C-z** means hit the ‘z’ key while holding down the *Ctrl* key. **M-z** means hit the ‘z’ key while hitting the *Meta* (labeled *Alt* on some keyboards) or after hitting *Esc* key.

## Running Emacs

To enter GNU Emacs, just type its name: **emacs**. Emacs divides the frame into several areas: menu line, buffer area with the edited text, mode line describing the buffer in the window above it, and a minibuffer/echo area in the last line.

<b>C-x C-c</b>	quit Emacs
<b>C-x C-f</b>	edit file; this command uses the minibuffer to read the file name; use this to create new files by entering the name of the new file
<b>C-x C-s</b>	save the file
<b>C-x k</b>	kill a buffer
<b>C-g</b>	in most context: cancel, stop, abort partially typed or executing command
<b>C-x u</b>	undo

## Moving About

<b>C-l</b>	scroll current line to center of window
<b>C-x b</b>	switch to another buffer
<b>M-&lt;</b>	move to beginning of buffer
<b>M-&gt;</b>	move to end of buffer
<b>M-g M-g</b>	go to a given line number

## Multiple Windows

<b>C-x 0</b>	remove the current window from the display
<b>C-x 1</b>	make active window the only window
<b>C-x 2</b>	split window horizontally
<b>C-x 3</b>	split window vertically
<b>C-x o</b>	move to other window

## Regions

Emacs defines a ‘region’ as the space between the *mark* and the *point*. A mark is set with **C-space**. The point is at the cursor position.

<b>M-h</b>	mark entire paragraph
<b>C-x h</b>	mark entire buffer

## Killing and Copying

<b>C-w</b>	kill region
<b>M-w</b>	copy region to kill-ring
<b>C-k</b>	kill from the cursor all the way to the end of the line
<b>M-DEL</b>	kill word
<b>C-y</b>	yank back the last kill ( <b>C-w C-y</b> combination could be used to move text around)
<b>M-y</b>	replace last yank with previous kill

## Searching

<b>C-s</b>	search for a string
<b>C-r</b>	search for a string backwards
<b>RET</b>	quit searching
<b>M-C-s</b>	regular expression search
<b>M-C-r</b>	reverse regular expression search

Use **C-s** or **C-r** again to repeat the search in either direction.

## Tags

Tags tables files record locations of function and procedure definitions, global variables, data types and anything else convenient. To create a tags table file, type ‘**etags input\_files**’ as a shell command.

<b>M-.</b>	find a definition
<b>M-,</b>	pop back to where <b>M-.</b> was last invoked
<b>M-x tags-query-replace</b>	run query-replace on all files recorded in tags table

## Compiling

<b>M-x compile</b>	compile code in active window
<b>C-c C-c</b>	go to the next compiler error, when in the compile window or
<b>C-x ‘</b>	when in the window with source code

## Dired, the Directory Editor

<b>C-x d</b>	invoke Dired
<b>d</b>	flag this file for deletion
<b>~</b>	flag all backup files for deletion
<b>u</b>	remove deletion flag
<b>x</b>	delete the files flagged for deletion
<b>C</b>	copy file
<b>g</b>	update the Dired buffer
<b>f</b>	visit the file described on the current line
<b>s</b>	switch between alphabetical date/time order

## Reading and Sending Mail

<b>M-x rmail</b>	start reading mail
<b>q</b>	quit reading mail
<b>h</b>	show headers
<b>d</b>	mark the current message for deletion
<b>x</b>	remove all messages marked for deletion
<b>C-x m</b>	begin composing a message
<b>C-c C-c</b>	send the message and switch to another buffer
<b>C-c C-f C-c</b>	move to the ‘Cc’ header field, creating one if there is none

## Miscellaneous

<b>M-q</b>	fill paragraph
<b>M-/</b>	expand previous word dynamically
<b>C-z</b>	iconify (suspend) Emacs when running it under X or shell, respectively
<b>M-x revert-buffer</b>	replace the text being edited with the text of the file on disk

## Query Replace

**M-%** interactively search and replace  
**M-C-%** using regular expressions

Valid responses in query-replace mode are

**SPC** replace this one, go on to next  
**,** replace this one, don't move  
**DEL** skip to next without replacing  
**!** replace all remaining matches  
**^** back up to the previous match  
**RET** exit query-replace  
**C-r** enter recursive edit (**M-C-c** to exit)

## Regular Expressions

**.** (dot) any single character except a newline  
**\*** zero or more repeats  
**+** one or more repeats  
**?** zero or one repeat  
**[...]** denotes a class of character to match  
**[^...]** negates the class  
**\c** quote characters otherwise having a special meaning in regular expressions  
**...\|...\|...** matches one of the alternatives ("or")  
**\( ... \)** groups a series of pattern elements to a single element  
  
**\n** same text as *n*th group  
**^** matches at line beginning  
**\$** matches at line end  
**\w** matches word-syntax character  
**\W** matches non-word-syntax character  
**\<** matches at word beginning  
**\>** matches at word end  
**\b** matches at word break  
**\B** matches at non-word break

## Registers

**C-x r s** save region in register  
**C-x r i** insert register contents into buffer  
**C-x r SPC** save value of point in register  
**C-x r j** jump to point saved in register

## Rectangles

**C-x r r** copy rectangle to register  
**C-x r k** kill rectangle  
**C-x r y** yank rectangle  
**C-x r t** prefix each line with a string  
**C-x r o** open rectangle, shifting text right  
**C-x r c** blank out rectangle

## Shells

**M-x shell** start a shell within Emacs  
**M-!** execute a shell command  
**M-|** run a shell command on the region  
**C-u M-|** filter region through a shell command

## Spelling Check

**M-\$** check spelling of word at the cursor  
**M-x ispell-region** check spelling of all words in region  
**M-x ispell-buffer** check spelling of entire buffer

## International Character Sets

**C-x RET C-\** select and activate input method for the current buffer  
**C-\** enable or disable input method  
**M-x list-input-methods** show all input methods  
**M-x set-language-environment** specify principal language  
**C-x RET c** set coding system for next command  
**M-x find-file-literally** visit file with no conversion of any kind  
**M-x list-coding-systems** show all coding systems  
**M-x prefer-coding-system** choose preferred coding system

## Keyboard Macros

**C-x (** start defining a keyboard macro  
**C-x )** end keyboard macro definition  
**C-x e** execute last-defined keyboard macro  
**C-u C-x (** append to last keyboard macro  
**M-x name-last-kbd-macro** name last keyboard macro

## Simple Customization

**M-x customize** customize variables and faces

## Getting Help

Emacs does command completion for you. Typing **M-x tab** or *space* gives a list of Emacs commands.

**C-h** Emacs help  
**C-h t** run the Emacs tutorial  
**C-h i** enter Info, the documentation browser  
**C-h a** show commands matching a string (apropos)  
**C-h k** display documentation of the function invoked by keystroke

Emacs gets into different *modes*, each of which customizes Emacs for editing text of a particular sort. The mode line contains names of the current modes, in parentheses.

**C-h m** get mode-specific information