Appendix E. Quick Reference

This quick reference is arranged topically, in roughly the same order as the commands were treated in the text. Unfortunately, it's impossible to be both "quick" and thorough, particularly with an editor as large and comprehensive as GNU Emacs. We've tried to take a middle road between completeness and quickness; we'll confess that, if we've erred, we've erred on the side of quickness.

Table E-1. File-handling commands (Chapter 1)

Keystrokes	Command name	Action
C-x C-f File ⟶ Open File	find-file	Find file and read it in a new buffer.
C-x C-v	find-alternate-file	Read an alternate file, replacing the one read with C-x C-f.
C-x i File ⟶ Insert File	insert-file	Insert file at cursor position.
C-x C-s File ⟶ Save (current buffer)	save-buffer	Save file.
C-x C-w File ⟶ Save Buffer As	write-file	Write buffer contents to file.
C-x C-c File ⟶ Exit Emacs	save-buffers-kill-emacs	Exit Emacs.

Table E-2. Cursor movement commands (Chapter 2)

Keystrokes	Command name	Action
C-f	forward-char	Move *forward* one character (right).
C-b	backward-char	Move *backward* one character (left).
C-p	previous-line	Move to *previous* line (up).
C-n	next-line	Move to *next* line (down).
M-f	forward-word	Move one word *forward*.
M-b	backward-word	Move one word *backward*.
C-a	beginning-of-line	Move to beginning of line.
C-e	end-of-line	Move to *end* of line.
M-e	forward-sentence	Move forward one sentence.
M-a	backward-sentence	Move backward one sentence.
C-v	scroll-up	Move forward one screen.
M-v	scroll-down	Move backward one screen.
M-<	beginning-of-buffer	Move to beginning of file.
M->	end-of-buffer	Move to end of file.
(*none*)	goto-line	Go to line n of file.
(*none*)	goto-char	Go to character n of file.
M-n	digit-argument	Repeat the next command n times.
C-u n	universal-argument	Repeat the next command n times (four times if you omit n).

Table E-3. Deleting, yanking, region, and clipboard commands (Chapter 2)

Keystrokes	Command name	Action
C-d	delete-char	Delete character under cursor.

Del	delete-backward-char	Delete previous character.
M-d	**kill-word**	Delete next word.
M-Del	**backward-kill-word**	Delete previous word.
C-y *Edit* ➞ *Paste Most Recent*	**yank**	Restore what you've deleted.
C-w *Edit* ➞ *Cut*	**kill-region**	Delete a marked region (see next section).
C-@ *or* **C- Space**	**set-mark-command**	Mark the beginning (or end) of a region.
C-x C-x	**exchange-point-and-mark**	Exchange location of cursor and mark.
C-w	**kill-region**	Delete the region.
C-y	**yank**	Paste most recently killed or copied text.
M-w	**kill-ring-save**	Copy the region (so it can be pasted with **C-y**).
C-x h	**mark-whole-buffer**	Mark buffer.
M-y	**yank-pop**	After **C-y**, pastes earlier deletion.
(none)	**clipboard-kill-region**	Cut region and place both in kill ring and on system clipboard.
(none)	**clipboard-yank**	Paste text from clipboard.
(none)	**clipboard-kill-ring-save**	Copy text to clipboard.

Table E-4. Text filling and reformatting commands (Chapter 2)

Keystrokes	Command name	Action
(*none*) *Options* ➞ *Word Wrap in Text Modes*	**auto-fill-mode**	Toggle auto-fill mode, in which Emacs formats paragraphs as you type them.
M-q	**fill-paragraph**	Reformat paragraph.
(*none*) *Edit* ➞ *Fill*	**fill-region**	Reformat individual paragraphs within a region.

Table E-5. Stopping and undoing commands (Chapter 2)

Keystrokes	Command name	Action
C-g	**keyboard-quit**	Abort current command.
C-x u	**advertised-undo**	Undo last edit (can be done repeatedly).
C-_ *Edit* ➞ *Undo*	**undo**	Undo last edit.
(*none*)	**revert-buffer**	Restore buffer to the state it was in when the file was last saved (or auto-saved).

Table E-6. Search and replace commands (Chapter 3)

Keystrokes	Command name	Action
C-s *Edit* ➞ *Search* ➞ *Incremental Search* ➞ *Forward String*	**isearch-forward**	Start incremental search forward; follow by search string. Also, find next occurrence (forward) of search string.
C-r *Edit* ➞ *Search* ➞		

Keystrokes	Command name	Action
C-r *Edit* → *Search* → *Incremental Search* → *Backward String*	**isearch-backward**	Start incremental search backward; follow by search string. Also, find next occurrence (backward) of search string.

Table E-7. Regular expression search commands (Chapter 3)

Keystrokes	Command name	Action
C-M-s Enter *Edit* → *Search* → *Regexp Forward*	**re-search-forward**	Search for a regular expression forward.
C-M-r Enter *Edit* → *Search* → *Regexp Backwards*	**re-search-backward**	Search for a regular expression backward.
C-M-s *Edit* → *Search* → *Incremental Search* → *Forward Regexp*	**isearch-forward-regexp**	Search incrementally forward for a regular expression.
C-M-r *Edit* → *Search* → *Incremental Search* → *Backward Regexp*	**isearch-backward-regexp**	Search incrementally backward for a regular expression.
C-M-% *Edit* → *Replace* → *Replace Regexp*	**query-replace-regexp**	Query-replace a regular expression.

Table E-8. Spell-checking commands (Chapter 3)

Keystrokes	Command name	Action
(*none*) *Tools* → *Spell Checking* → *Spell-Check Buffer*	**ispell-buffer**	Check spelling of the buffer.
(*none*) *Tools* → *Spell Checking* → *Spell-Check Comments*	**ispell-comments-and-strings**	Checks spelling of comments and strings in a program.
(*none*) *Tools* → *Spell Checking* → *Automatic Spell-Checking (Flyspell)*	**flyspell-mode**	Enter the flyspell minor mode, in which incorrectly spelled words are highlighted.
(*none*)	**flyspell-buffer**	Spell-check the current buffer, underlining all misspelled words. Use middle mouse button to correct.

Table E-9. Buffer commands (Chapter 4)

Keystrokes	Command name	Action
C-x b *Buffers* → *Select Named Buffer*	**switch-to-buffer**	Move to the buffer specified.
C-x → *Buffers* → *Next Buffer*	**next-buffer**	Move to the next buffer in the buffer list.
C-x *Buffers* → *Previous Buffer*	**previous-buffer**	Move to the previous buffer in the buffer list.
C-x C-b *Buffers* → *List All Buffers*	**list-buffers**	Display the buffer list.
C-x k	**kill-buffer**	Delete the buffer specified.
(*none*)	**kill-some-buffers**	Ask about deleting each buffer.

Table E-10. Windows and frames (Chapter 4)

Keystrokes	Command name	Action
C-x 2 *File* → *Split Window*	**split-window-vertically**	Divide current window into two windows, one above the other.
C-x 3	**split-window-horizontally**	Divide current window into two side-by-side windows.

C-x o	other-window	Move to the other window; if there are several, move to the next window.
C-x 0	delete-window	Delete the current window.
C-x 1 *File* ➜ *Unsplit Windows*	delete-other-windows	Delete all windows but this one.
C-x 4 f	find-file-other-window	Find a file in the other window.
C-x 4 b	switch-to-buffer-other-window	Select a buffer in the other window.
(*none*) *Tools* ➜ *Compare (Ediff)* ➜ *This Window and Next Window*	compare-windows	Compare this window with the next window and show the first difference.
C-x 5 o *Buffers* ➜ *Frames*	other-frame	Move to other frame.
C-x 5 0 *File* ➜ *Delete Frame*	delete-frame	Delete current frame.
C-x 5 2 *File* ➜ *New Frame*	make-frame	Create a new frame on the current buffer.
C-x 5 f	find-file-other-frame	Find file in a new frame.
C-x 5 r	find-file-read-only-other-frame	Finds a file in a new frame, but it is read-only.
C-x 5 b	switch-to-buffer-other-frame	Make frame and display other buffer in it.

Table E-11. Shell mode commands (Chapter 5)

Keystrokes	Command name	Action
(*none*)	shell	Enter shell mode.
C-c C-c *Signals* ➜ *BREAK*	comint-interrupt-subjob	Interrupt current job; equivalent to C-c.
C-c C-z *Signals* ➜ *STOP*	comint-stop-subjob	Suspend or stop a job; C-z in Unix shells.
M-p *In/Out* ➜ *Previous Input*	comint-previous-input	Retrieve previous commands (can be repeated to find earlier commands).
M-n *In/Out* ➜ *Next Input*	comint-next-input	Retrieve subsequent commands (can be repeated to find more recent commands).
Enter	comint-send-input	Send input on current line.
Tab	comint-dynamic-complete	Complete current command, filename, or variable name.

Table E-12. Dired commands (Chapter 5)

Keystrokes	Command name	Action
C-x d *File* ➜ *Open Directory*	dired	Start Dired.
C *Operate* ➜ *Copy to*	dired-do-copy	Copy file.
d *Mark* ➜ *Flag*	dired-flag-file-deletion	Flag for deletion.
D *Operate* ➜ *Delete*	dired-do-delete	Query for immediate deletion.

Keystrokes	Command name	Action
f	**dired-advertised-find-file**	Find (so you can edit).
g *Immediate* ➡ *Refresh*	**revert-buffer**	Reread the directory from disk.
m or *** m** *Mark* ➡ *Mark*	**dired-mark**	Mark with *****.
Q *Operate* ➡ *Query Replace in Files*	**dired-do-query-replace**	Query replace string in marked files.
R *Operate* ➡ *Rename to*	**dired-do-rename**	Rename file.
s	**dired-sort-toggle-or-edit**	Sort the Dired display by date or by filename (toggles between these).
t *Mark* ➡ *Toggle Marks*	**dired-toggle-marks**	Toggle marks on files and directories; pressing **t** once marks all unmarked files and directories; pressing **t** again restores original marks.
u *Mark* ➡ *Unmark*	**dired-unmark**	Remove mark.
+ *Immediate* ➡ *Create Directory*	**dired-create-directory**	Create a directory.
* ! or **M-Del** *Mark* ➡ *Unmark All*	**dired-unmark-all-files**	Remove all marks from all files.

Table E-13. Macro commands (Chapter 6)

Keystrokes	Command name	Action
C-x (	**kmacro-start-macro**	Start macro definition.
F3	**kmacro-start-macro-or-insert-counter**	Start macro definition. If pressed while defining a macro, insert a counter.
C-x)	**kmacro-end-macro**	End macro definition.
F4	**kmacro-end-or-call-macro**	End macro definition (if definition is in progress) or invoke last keyboard macro.
C-x e	**kmacro-end-and-call-macro**	Execute last keyboard macro defined. Can type **e** to repeat macro.
C-x C-k n	**name-last-kbd-macro**	Name the last macro you created (before saving it).
C-x C-k e	**edit-kbd-macro**	Edit a keyboard macro by typing C-x e for the last keyboard macro defined, M-x for a named macro, C-h l for lossage, or keystrokes for a macro bound to a key.
C-x C-k Enter	**kmacro-edit-macro**	Edit the last keyboard macro.

Table E-14. Outline mode commands (Chapter 7)

Keystrokes	Command name	Action
(none)	outline-mode	Toggle outline mode
C-c C-t *Hide* ➡ *Hide Body*	hide-body	Hide all body lines.

C-c C-a *Show* ⟶ *Show All*		show-all	Show everything that's hidden.

Table E-15. Compilation mode commands ([Chapter 9](#))

Keystrokes	Command name	Action
C-x `	**next-error**	Move to the next error message and visit the corresponding source code.
M-n	**compilation-next-error**	Move to the next error message.
M-p	**compilation-previous-error**	Move to the previous error message.
C-c C-c	**compilation-goto-error**	Visit the source code for the current error message.

Table E-16. Basic indentation commands ([Chapter 7](#) and [Chapter 9](#))

Keystrokes	Command name	Action
C-M-	**indent-region**	Indent each line between the cursor and mark.
M-m	**back-to-indentation**	Move to the first nonblank character on the line.
M-^	**delete-indentation**	Join this line to the previous one.

Table E-17. C motion commands ([Chapter 9](#))

Keystrokes	Command name	Action
M-a	**c-beginning-of-statement**	Move to the beginning of the current statement.
M-e	**c-end-of-statement**	Move to the end of the current statement.
M-q	**c-fill-paragraph**	If in comment, fill the paragraph, preserving indentations and decorations.
C-M-a	**beginning-of-defun**	Move to the beginning of the body of the function surrounding the point.
C-M-e	**end-of-defun**	Move to the end of the function.
C-M-h	**c-mark-function**	Put the cursor at the beginning of the function, the mark at the end.
C-c C-q	**c-indent-defun**	Indent the entire function according to indentation style.
C-c C-u	**c-up-conditional**	Move to the beginning of the current preprocessor conditional.
C-c C-p	**c-backward-conditional**	Move to the previous preprocessor conditional.
C-c C-n	**c-forward-conditional**	Move to the next preprocessor conditional.

Table E-18. SQL mode commands ([Chapter 9](#))

Keystrokes	Command name	Action
C-c C-c	sql-send-paragraph	Send the paragraph the cursor is on. A paragraph is defined by the particular database client. For the **sql-mysql** process, for example, a paragraph begins with a statement like select or update and ends with a semicolon. Any number of lines can intervene.
C-c C-r	sql-send-region	Send the marked region.
C-c C-b	sql-send-buffer	Send the entire buffer.

Table E-19. Lisp commands ([Chapter 9](#))

Keystrokes	Command name	Action
C-M-b	**backward-sexp**	Move backward by one S-expression.
C-M-f	**forward-sexp**	Move forward by one S-expression.
C-M-t	**transpose-sexps**	Transpose the two S-expressions around the cursor.
C-M-@	**mark-sexp**	Set mark to the end of the current S-expression; set the cursor to the beginning.
C-M-k	**kill-sexp**	Delete the S-expression following the cursor.
(*none*)	**backward-kill-sexp**	Delete the S-expression preceding the cursor.
C-M-n	**forward-list**	Move forward by one list.
C-M-p	**backward-list**	Move backward by one list.
C-M-d	**down-list**	Move forward and down one parenthesis level.
(*none*)	**up-list**	Move forward out of one parenthesis level.
C-M-u	**backward-up-list**	Move backward out of one parenthesis level.
C-M-a	**beginning-of-defun**	Move to the beginning of the current function.
C-M-e	**end-of-defun**	Move to the end of the current function.
C-M-h	**mark-defun**	Put the cursor at the beginning of the function, put the mark at the end.

Table E-20. VC commands (Chapter 12)

Keystrokes	Command name	Action
C-x v v	**vc-next-action**	Go to the next logical version control state.
C-x v d	**vc-directory**	Show all registered files beneath a directory.
C-x v =	**vc-diff**	Generate a version difference report.
C-x v u	**vc-revert-buffer**	Throw away changes since the last checked-in revision.
C-x v ~	**vc-version-other-window**	Retrieve a given revision in another window.
C-x v l	**vc-print-log**	Display a file's change comments and history.
C-x v i	**vc-register**	Register a file for version control.
C-x v h	**vc-insert-headers**	Insert version control headers in a file.
C-x v r	**vc-retrieve-snapshot**	Check out a named project snapshot.
C-x v s	**vc-create-snapshot**	Create a named project snapshot.
C-x v c	**vc-cancel-version**	Throw away a saved revision.
C-x v a	**vc-update-change-log**	Update a GNU-style ChangeLog file.

Table E-21. Ediff commands (Chapter 12)

Keystrokes	Command name	Action
Space or **n**	**ediff-next-difference**	Move to the next difference between the files.
Del or **p**	**ediff-previous-difference**	Move to the preceding difference between the files.
j	**ediff-jump-to-difference**	Go to the difference specified as a numeric prefix argument.
a	**ediff-copy-A-to-B**	Copy the version of the current difference found in buffer A to buffer B.
b	**ediff-copy-B-to-A**	Copy the version of the current difference found in buffer B to buffer A.

r a or r b	ediff-restore-diff	Restore the current difference in buffer A (or B) to the way it was before copying from the other buffer.
A or B	ediff-toggle-read-only	Switch the specified buffer into (or out of) read-only mode.
g a or g b	ediff-jump-to-difference-at-point	Recenter the comparison buffers on the difference nearest to your current location (point) in the specified buffer.
!	ediff-update-diffs	Recalculate and redisplay the highlighted regions; useful if you've manually made extensive changes to a buffer.
w a or w b	ediff-save-buffer	Save the specified buffer to disk.
z	ediff-suspend	Close the Ediff control window, but leave the session active so you can resume it later.
q	ediff-quit	Close the Ediff window and end this comparison session.

Table E-22. CUA mode commands (Chapter 13)

Keystrokes	Command name	Action
C-c C-x C-x	cua-exchange-point-and-mark	Exchange location of cursor and mark.
C-x *or* C-w *or* S-Delete	kill-region	Delete the region.
C-v *or* C-y *or* S-Insert	cua-paste	Paste most recently killed or copied text.
C-c	copy-region-as-kill	Copy the region.
M-v	cua-repeat-replace-region	After highlighting and replacing a string, find the next string and replace it the same way.
M-y	cua-paste-pop	After **C-v**, pastes earlier deletion.
C-z *or* C-x u	cua-undo	Undoes the last change.
C-x C-z	iconify-frame	Minimize the current frame (what **C-z** does outside CUA mode).

Table E-23. Help commands (Chapter 14)

Keystrokes	Command name	Question answered
C-h k *Help* ⟶ *Describe* ⟶ *Describe Key*	describe-key	What command does this keystroke sequence run, and what does it do?
C-h f *Help* ⟶ *Describe* ⟶ *Describe Function*	describe-function	What does this function do?
C-h v *Help* ⟶ *Describe* ⟶ *Describe Variable*	describe-variable	What does this variable mean, and what is its value?
C-h m *Help* ⟶ *Describe* ⟶ *Describe Buffer Modes*	describe-mode	Tell me about the modes the current buffer is in.
C-h b *Help* ⟶ *Describe* ⟶ *List Key Bindings*	describe-bindings	What are all the key bindings for this buffer?
C-h a *Help* ⟶ *Search Documentation* ⟶ *Find Commands by Name*	apropos-command	What commands include this word?
(*none*) *Help* ⟶ *Search Documentation* ⟶ *Find Options by Name*	apropos-variable	What variables include this regular expression?
(*none*) *Help* ⟶ *Search Documentation* ⟶ *Find Any Object by Name*	apropos	What functions and variables involve this regular expression?

Table E-24. Documentation help commands (Chapter 14)

Keystrokes	Command name	Action
C-h t *Help* → *Emacs Tutorial*	**help-with-tutorial**	Run the Emacs tutorial.
C-h i	**info**	Start the Info documentation reader. If prefaced with **C-u**, reads an Info file of your choice.
C-h r *Help* → *Read the Emacs Manual*	**info-emacs-manual**	Open the Emacs manual.
C-h K *Help* → *More Manuals* → *Find Key in Manual*	**Info-goto-emacs-key-command-node**	Start Info documentation reader at the node that discusses this key sequence.
(*none*) *Help* → *Search Documentation* → *Look Up Subject in User Manual*	**emacs-index-search**	Search the index of the Emacs user manual.
C-h p	**finder-by-keyword**	Invoke a menu that lets you get information about Emacs Lisp packages available on your system.

Table E-25. Important modes

Mode	Function
fundamental-mode	The default mode
text-mode	Major mode for writing text (Chapter 2)
outline-mode	Major mode for writing outlines (Chapter 7)
picture-mode	Major mode for creating ASCII drawings using the keyboard (Chapter 7)
html-mode	Major mode for writing HTML (Chapter 8)
sgml-mode	Major mode for writing SGML and XML (Chapter 8)
latex-mode	Major mode for formatting files for TEX and LATEX (Chapter 8)
c-mode, c++-mode	Major mode for writing C and C++, and Java programs (Chapter 9)
java-mode	Major mode for writing Java programs (Chapter 9)
perl-mode, cperl-mode	Major modes for writing Perl programs (Chapter 9)
sql-mode	Major mode for interacting with databases using SQL (Chapter 9)
emacs-lisp-mode	Major mode for writing Emacs Lisp functions (Chapter 9 and Chapter 11)
lisp-mode	Major mode for writing Lisp programs (Chapter 9 and Chapter 11)
lisp-interaction-mode	Major mode for writing and evaluating Lisp expressions (Chapter 9 andChapter 11)
auto-fill-mode	Minor mode that enables word wrap (Chapter 2)
overwrite-mode	Minor mode that replaces characters as you type instead of inserting them (Chapter 2)
flyspell-mode	Minor mode for flyspell spell-checker (Chapter 3)
flyspell-prog-mode	Minor mode for spell-checking programs with flyspell (Chapter 3)
abbrev-mode	Minor mode for word abbreviations (Chapter 3)
artist-mode	Minor mode for creating ASCII drawings using the mouse (Chapter 7)
font-lock-mode	Minor mode for highlighting text in colors and fonts (Chapter 9)
vc-mode	Minor mode for using version control systems (Chapter 12)