

GNU Texinfo Reference Card

(for Texinfo version 4.13)

<http://www.gnu.org/software/texinfo/>

Texinfo document skeleton

Texinfo source files are plain text; standard extensions are ‘.texinfo’, ‘.texi’, and ‘.txi’. A Texinfo file must begin with lines like this:

```
\input texinfo
@setfilename info-file-name
@settitle name-of-manual

...the contents of the Texinfo document follow, ending with:
@bye
```

Texinfo @-commands

Beginning a Texinfo document

The commands in this section are given approximately in the order in which they conventionally appear.

@setfilename *info-file-name* Provide a name to be used for the output files. This command is essential for T_EX formatting as well, even though it produces no output. Should be the first command in the input.

@settitle *title* Specify document *title*.

Internationalization

@documentencoding *enc* Declare the input encoding to be *enc*; use right after **@setfilename**. Default is US-ASCII; other possibilities: UTF-8, ISO-8859-1, koi8-r, ...

@documentlanguage *LL*[_*CC*] Declare the current language as the two-character ISO-639 abbreviation *LL*, optionally followed by an underscore and a two-letter ISO 3166 two-letter country code (*CC*). Default is en_US.

@frenchspacing on/off Control spacing after punctuation.

Info directory specifications

@dircategory *category name* Specify an Info directory section for this file’s entries. Can be repeated. We try to follow the categories in the Free Software Directory.

@direntry Begin the Info directory menu entries for this file. Pair with **@end direntry**.

HTML document description

@documentdescription Set the document description text for HTML; default is just the title of the document. Pair with **@end documentdescription**.

Title pages

@shorttitlepage *title* Generate a minimal title page, generally for the first page of output in a long printed manual.

@center line-of-text Center *line-of-text*, usually for titles.

@titlefont {*text*} In a printed manual, print *text* in a larger than normal font. Ignored in Info.

@titlepage Begin the title page. Write the command on a line of its own, paired with **@end titlepage**. Ignored in Info.

@title *title* Within **@titlepage**, display *title* appropriately.

@subtitle *subtitle* Within **@titlepage**, display *subtitle*.

@author *author* Within **@titlepage**, display *author*.

Tables of contents

@shortcontents Print a short table of contents (chapter-level entries only). Not relevant to Info.

@summarycontents Synonym for **@shortcontents**.

@contents Print a complete table of contents. Has no effect in Info, which uses menus instead.

@setcontentsaftertitlepage Put the table of contents after the ‘@end titlepage’ even if the **@contents** is at the end.

@setshortcontentsaftertitlepage Place the short table of contents after the ‘@end titlepage’ command even if the **@shortcontents** command is at the end.

Nodes

@node *name, next, previous, up* Begin a new node. The *next*, *previous*, and *up* pointers are usually omitted in a normally-structured document.

@top *title* Mark the topmost **@node** in the file, which must be defined on the line immediately preceding **@top**. The *title* is formatted as a chapter-level heading. The entire top node, including the **@node** and **@top** lines, are normally enclosed with **@ifnottex ... @end ifnottex**.

@anchor {*name*} Define *name* as the current location, for use as a cross-reference target.

@novalidate Suppress validation of node references and omit creation of auxiliary files with T_EX. Use before **@setfilename**.

Chapter structuring

@lowersections Change subsequent chapters to sections, sections to subsections, and so on.

@raisesections Change subsequent sections to chapters, subsections to sections, and so on.

@part *title* Print part page, beginning a group of chapters; included in contents.

Numbered, included in contents

@chapter *title* Begin a numbered chapter; *title* appears in the table of contents.

@centerchap *title* Like **@chapter**, with chapter title centered.

Context-dependent, included in contents

@section *title* Begin a section within a chapter. Within **@chapter** and **@appendix**, the section is numbered; within **@unnumbered**, it is unnumbered.

@subsection *title* Begin a subsection within a section. Same context-dependent numbering as **@section**.

@subsubsection *title* Begin a numbered subsubsection within a subsection. Same context-dependent numbering as **@section**.

Unnumbered, included in contents

@unnumbered *title* Begin an unnumbered chapter.

@unnumberedsec *title* Begin an unnumbered section.

@unnumberedsubsec *title* Begin an unnumbered subsection.

@unnumberedsubsubsec *title* Begin an unnumbered subsubsection.

Lettered and numbered, included in contents

@appendix *title* Begin an appendix.

@appendixsec *title* Begin an appendix section within an appendix; **@appendixsection** is a synonym.

@appendixsubsec *title* Begin an appendix subsection;

@appendixsubsection is a synonym.

@appendixsubsubsec *title* Begin an appendix subsubsection;

@appendixsubsubsection is a synonym.

Unnumbered, not in contents, no new page

@chapheading *title* Print an unnumbered chapter-like heading.

@majorheading *title* Like **@chapheading**, but generate additional vertical whitespace before the heading.

@heading *title* Print an unnumbered section-like heading.

@subheading *title* Print an unnumbered subsection-like heading.

@subsubheading *title* Print an unnumbered subsubsection-like heading.

Menus

@menu Mark the beginning of a menu of nodes in Info. No effect in a printed manual. Pair with **@end menu**.

@detailmenu Mark the (optional) detailed node listing in a master menu.

Cross references

Within the Info system

@xref {*node*, [*entry*], [*node-title*], [*info-file*], [*manual*]} Makes a reference that starts with ‘See’ in a printed manual. Follow command with punctuation. Only the first argument is mandatory.

@pxref {*node*, [*entry*], [*node-title*], [*info-file*], [*manual*]} Like **@xref**, but starts with ‘see’ instead of ‘See’, and must be used inside parentheses.

@ref {*node*, [*entry*], [*node-title*], [*info-file*], [*manual*]} Like **@xref**, but produces only the bare reference without ‘See’ or ‘see’; must be followed by a punctuation mark.

@inforef {*node*, [*entry-name*], *info-file*} Make a cross reference to an Info file for which there is no printed manual.

Outside of Info

@url {*url*, [*displayed-text*], [*replacement*]} Make a hyperlink reference to a uniform resource locator. **@uref** is a synonym.

@cite {*title*} Indicate the name of a book or other reference that has no Info file; no hyperlink.

Marking words and phrases

Markup for regular text

@emph {*text*} Emphasize *text*, by using italics where possible, and enclosing in asterisks in Info.

@strong {*text*} Emphasize *text* even more strongly, by using bold where possible; still enclosed in asterisks in Info.

@var {*metasyntactic-variable*} Indicate a meta variable (a name that stands for another piece of text).

@dfn {*term*} Indicate the introductory or defining use of a technical term.

@acronym {*acronym*} Indicate an acronym written in all capital letters, such as ‘NASA’.

@abbr {*abbreviation*} Indicate a general abbreviation, such as ‘Comput.’

Markup for literal text

@code {*sample-code*} Indicate an expression, a syntactically complete token, or a program name. Unquoted in Info output.

@file {*filename*} Indicate the name of a file, buffer, node, directory, etc.

@command {*command-name*} Indicate a system command name, such as **ls**.

@env {*environment-variable*} Indicate an environment variable name, such as **PATH**.

@option {*option-name*} Indicate a command-line option, such as **-v** and **--help**.

@kbd {*keyboard-characters*} Indicate characters of input to be typed by users.

@key {*key-name*} Indicate the name of a key on a keyboard.

@email {*address*, [*displayed-text*]} Indicate an email address.

@indicateurl {*url*} Indicate a uniform resource locator, but without creating any hyperlink.

`@samp {text}` Indicate a sequence of literal characters, in general. Quoted in Info output.

`@verb {delimchar literal-text delimchar}` Output *literal-text*, delimited by the single character *delimchar*, exactly as is, including any whitespace or Texinfo special characters. Example: `@verb{|@ @|}`.

Explicit font selection

`@sc {text}` Set *text* in a small caps font in printed output, and uppercase in Info. No other font command has any effect in Info.

`@r {text}` Set *text* in the regular roman font.

`@i {text}` Set *text* in an italic font.

`@slanted {text}` Set *text* in a slanted (oblique) font if possible.

`@b {text}` Set *text* in a bold font.

`@sansserif {text}` Set *text* in a sans serif font if possible.

`@t {text}` Set *text* in a fixed-width, typewriter-like font.

Quotation and example environments

‘Environments’ in Texinfo begin with an `@envname` on a line of its own and continue through the following lines of input until a corresponding `@end envname` is seen.

`@noindent` Omit the normal indentation when starting a new paragraph; often used after environments.

`@exdent line-of-text` Output *line-of-text* without any indentation.

`@cartouche` Highlight an example or quotation by drawing a box with rounded corners around it. Pair with `@end cartouche`. No effect in Info.

Displays in regular text fonts

`@format` Begin formatted text: do not fill text; retain normal margins and normal text font. Pair with `@end format`.

`@smallformat` Like `@format`, but select a smaller (text) font where possible. Pair with `@end smallformat`.

`@display` Begin displayed text: do not fill, and indent text; retain normal text font. Pair with `@end display`.

`@smalldisplay` Like `@display`, but select a smaller (text) font where possible. Pair with `@end smalldisplay`.

`@flushleft` Left justify lines: do not fill text, retain normal margins and normal text font. Pair with `@end flushleft`.

`@flushright` Right justify lines, leaving the left end ragged: do not fill text, retain normal margins and normal text font. Pair with `@end flushright`.

`@quotation` Narrow both margins to indicate text that is quoted from another work; do normal line breaking and retain normal font. Takes optional argument of prefix text. Pair with `@end quotation`.

`@raggedright` Left justify lines, leaving the right end ragged: fill text, retain normal text font. Pair with `@end raggedright`.

`@smallquotation` Like `@quotation`, but select a smaller (text) font where possible. Pair with `@end quotation`.

Displays using fixed-width fonts

`@example` Begin an example. Indent text, do not fill, and select fixed-width font. Pair with `@end example`.

`@smallexample` Like `@example`, but where possible, print text in a smaller font. Pair with `@end smallexample`.

`@lisp` Like `@example`, but specifically for an example of Lisp code. Pair with `@end lisp`.

`@smalllisp` Begin an example specifically of Lisp code; same effects as `@smallexample`. Pair with `@end smalllisp`.

`@verbatim` Output the text of the environment exactly as is, without indentation. Pair with `@end verbatim`.

Lists and tables

`@itemize mark-command` Begin an unordered list, using `@item` for each entry, which consist of indented paragraphs, with a mark such as `@bullet` inside the left margin at the beginning of each item. Pair with `@end itemize`.

`@enumerate [number-or-letter]` Begin a numbered list, using `@item` for each entry. Optionally, start list with *number-or-letter* and count from there; by default, number consecutively from ‘1’. Pair with `@end enumerate`.

`@table formatting-command` Begin a two-column table (description list), using `@item` for each entry. Write each first column entry on the same line as `@item`. *formatting-command* is applied to each first column entry. Pair with `@end table`.

`@ftable formatting-command` Like `@table`, but automatically enter each first column entry in the function index.

`@vtable formatting-command` Like `@table`, but automatically enter each first column entry in the variable index.

`@multitable column-width-spec` Begin multi-column table; start rows with `@item` or `@headitem`. Pair with `@end multitable`.

`@tab` Separate columns in a row of a multitable.

`@item` Begin an item for the list and table commands.

`@itemx` Used only with `@table`. Like `@item` but do not generate extra preceding vertical space. When several items have the same description, use `@item` for the first and `@itemx` for the rest.

`@headitem` Used only for multitable. Like `@item`, but indicates a heading row.

`@asis text` Prints *text* as-is; used as *formatting-command* for tables when no highlighting is desired.

Floating displays

`@float` Environment within which floating material is defined. Pair with `@end float`.

`@listoffloats` Produce a list of floats for a given float type.

`@caption` Define the full caption for a `@float` anywhere in the float environment.

`@shortcaption` Define the short caption for a `@float`, used in the list of floats.

`@image {filename, [width], [height], [alt], [ext]}` Include graphics image in external *filename* scaled to the given width and/or height, using alt text and looking for ‘*filename.ext*’ first. Can occur inline or by itself on the page.

Footnotes

`@footnote {text-of-footnote}` Produce a footnote: *text-of-footnote* is printed at the bottom of the page by T_EX, while Info and others may format at either (by default) the end of the current node, or in a separate node.

`@footnotestyle style` Specify footnote style: ‘end’ or ‘separate’.

Indices

`@cindex entry` Add entry to the index of concepts.

`@findex entry` Add entry to the index of functions.

`@vindex entry` Add entry to the index of variables.

`@kindex entry` Add entry to the index of keys.

`@pindex entry` Add entry to the index of programs.

`@tindex entry` Add entry to the index of data types.

`@defcodeindex newidx` Define a new index and its indexing command (namely, `@newidxindex`). Format entries in this index with `@code`.

`@defindex newidx` Define a new index and its indexing command. Format entries as regular text.

`@syncodeindex from-index to-index` Merge the contents of *from-index* into *to-index*, formatting the entries in *from-index* with `@code`.

`@synindex from-index to-index` Merge the contents of *from-index* into *to-index*; no special formatting.

`@printindex index-name` Generate an alphabetized index (using two columns in a printed manual).

Insertions within a paragraph

Characters special to Texinfo

`@@` Insert an at sign, ‘@’.

`@{` Produce a left brace, ‘{’.

`@}` Produce a right brace, ‘}’.

`@comma {}` Produce a comma character; needed only when a literal comma would be an argument separator, as in `@xref`.

`@whitespace` An `@` followed by a space, tab, or newline produces a normal, stretchable, interword space.

Spacing refinements

`@:` Tell T_PX to refrain from inserting extra whitespace after an immediately preceding period, question mark, exclamation mark, or colon, as T_EX normally would.

`@.` Produce a period that ends a sentence; typically needed after an end-of-sentence capital letter.

`@?` Produce a question mark that ends a sentence.

`@!` Produce an exclamation point that ends a sentence.

`@dmn {dimension}` Format a unit of measure, as in ‘12 pt’. Causes T_PX to insert a thin space before *dimension*. No effect in Info.

Accents

Braces are optional for the arguments to the accent commands, with the exception of `@tieaccent`.

`@" c` Generate an umlaut, ö.

`@' c` Generate an acute accent, ó.

`@, c` Generate a cedilla accent, ç.

`@= c` Generate a macron accent, ô.

`@^ c` Generate a circumflex (hat) accent, ô.

`@' c` Generate a grave accent, ó.

`@~ c` Generate a tilde accent, ô.

`@dotaccent c` Generate a dot (over) accent, ô.

`@H c` Generate a long Hungarian umlaut accent, ő.

`@ogonek c` Generate an ogonek diacritic, ą.

`@ringaccent c` Generate a ring accent, ô.

`@tieaccent {cc}` Generate a tie-after accent, oo; argument must be given in braces.

`@u c` Generate a breve accent, ô.

`@ubaraccent c` Generate an underbar accent, ɡ.

`@udotaccent c` Generate an underdot accent, ɡ.

`@v c` Generate a check (hacek) accent, ô.

Non-English characters

`@AA {}, @aa {}` Generate uppercase Å and lowercase å.

`@AE {}, @ae {}` Generate uppercase Æ and lowercase æ.

`@DH {}, @dh {}` Generate uppercase Ð and lowercase ð.

`@L {}, @l {}` Generate uppercase Ł and lowercase ł.

`@O {}, @o {}` Generate uppercase Ø and lowercase ø.

`@OE {}, @oe {}` Generate uppercase Œ and lowercase œ.

`@TH {}, @th {}` Generate uppercase Þ and lowercase þ.

`@exclamdown {}` Generate an upside-down exclamation point, ¡.

`@ordf {}, @ordm {}` Generate º and lowercase º.

`@questiondown {}` Generate an upside-down question mark, ¿.

`@ss {}` Generate the German sharp-S (es-zet) letter, ß.

Other text characters and logos

`@bullet {}` Produce a large round dot, • (*' in Info).

`@dots {}` Produce an ellipsis, '...'.

`@enddots {}` Produce an end-of-sentence ellipsis, '...' (more following space in \TeX .)

`@euro {}` Produce the Euro currency sign, €.

`@pounds {}` Produce the pounds sterling currency sign, £.

`@textdegree {}` Produce the degree symbol, °.

`@copyright {}` Produce the copyright symbol, ©.

`@registeredsymbol {}` Produce the registered symbol, ®.

`@TeX {}` Produce the \TeX logo.

`@LaTeX {}` Produce the \LaTeX logo.

`@today {}` Produce the current date, in '1 Jan 2000' style.

`@guillemetleft{} @guillemetright{} @guillemotleft{} @guillemotright`

`@guilsinglleft{} @guilsinglright{} @quotedblbase{} @quotedblleft @quotedblright{} @quoteleft{} @quoteright{} @quotesinglbase{} Produce various quotation marks: « » « » < > „ “ ” ‘ ’ , , .`

Glyphs for code examples

`@equiv {}` Indicate equivalence of two forms, ≡.

`@error {}` Indicate an error message, `error` .

`@expansion {}` Indicate the result of a macro expansion, →.

`@point {}` Indicate the position of point in a buffer, *.

`@print {}` Indicate printed output, ↵.

`@result {}` Indicate the result of an expression, ⇒.

GUI sequences

`@clicksequence {action @click{} action ...}` Represent a sequence of GUI actions or clicks.

`@click {}` Represent a single 'click' in a `@clicksequence`.

`@clickstyle {cmd}` Execute `@cmd` for each `@click`; default is `@arrow`. Empty braces usually following `@cmd` are omitted here.

`@arrow {}` Generate a right arrow glyph, →.

Math

`@math {math}` Format the given math (inline); \ becomes an escape character, and all \TeX math is available.

`@minus {}` Produce a minus sign, '−'.

`@\` Inside `@math`, generate a backslash, \.

`@geq {}` Generate a greater-than-or-equal sign, ≥.

`@leq {}` Generate a less-than-or-equal sign, ≤.

Making and preventing breaks

`@*` Force a line break.

`@/` Produces no output, but allows a line break.

`@-` Insert a discretionary hyphenation point.

`@hyphenation {hy-phen-a-ted words}` Explicitly define hyphenation points.

`@tie {}` Generate a normal interword space at which a line break is not allowed.

`@w {text}` Disallow line breaks within *text*.

`@refill` Does nothing (previously refilled and indented the preceding paragraph for Info).

Vertical space

`@sp n` Skip *n* blank lines.

`@page` Start a new page in a printed manual. Ignored in other formats.

`@need mils` Start a new page in a printed manual if fewer than *mils* thousandths of an inch remain on the current page. Ignored in other formats.

`@group` Disallow page breaks in following text. Pair with `@end group`. Ignored in other formats.

`@vskip amount` In a printed manual, insert whitespace *amount*. Ignored in other formats.

Definition commands

The ...*x* variant of each command is analogous to `@itemx`, leaving no extra space above. All definition commands must be written entirely on one line. Multiword arguments must be enclosed in braces. Descriptive text follows, ended by '`@end defcmdname`'.

Function-like definitions, untyped

`@defn[x]` *category name arguments* Format a description for a function, interactive command, or similar entity that may take arguments. Its arguments are the *category* of entity being described, the *name* of this particular entity, and the entity's *arguments*, if any.

`@defun[x]` *name arguments* Format a description for a function; equivalent to '`@defn Function ...`'.

`@defmac[x]` *name arguments* Format a description for a macro; equivalent to '`@defn Macro ...`'.

`@defspec[x]` *name arguments* Format a description for a special form; equivalent to '`@defn {Special Form} ...`'.

Function-like definitions, typed

`@deftypefn[x]` *category data-type name arguments* Like `@defn`, but for typed entities.

`@deftypefun[x]` *data-type name arguments* Like `@defun`, but for typed entities; equivalent to '`@deftypefn Function ...`'.

Variable-like definitions, untyped

`@defvr[x]` *category name* Format a description for any kind of variable. Its arguments are the *category* of the entity and the *name* of this particular entity.

`@defvar[x]` *name* Format a description for a variable; equivalent to '`@defvr Variable ...`'.

`@defopt[x]` *name* Format a description for a user option; equivalent to '`@defvar {User Option} ...`'.

Variable-like definitions, typed

`@deftypevr[x]` *category data-type name* Like `@defvr`, but for typed entities.

`@deftypevar[x]` *data-type name* Like `@defvar`, but for typed entities; equivalent to '`@deftypevr Variable ...`'.

Types

`@deftp[x]` *category name attributes* Format a description for a data type. Its arguments are the *category*, the *name* of the type (e.g., 'int'), and then the *attributes* of objects of that type.

Object-oriented entities, untyped

`@defop[x]` *category class name arguments* Format a description for an operation in object-oriented programming. Its arguments are the *category* of the variable, the *class* to which it belongs, its *name*, and any *arguments*.

`@defmethod[x]` *class name arguments* Format a description for method *name*; equivalent to '`@defop {Method on} ...`'.

`@defcv[x]` *category class name* Format a description for a variable associated with a class in object-oriented programming.

`@defivar[x]` *class name* Format a description for an instance variable *name* in object-oriented programming; equivalent to '`@defcv {Instance Variable of} ...`'.

Object-oriented entities, typed

`@deftypeop[x]` *category class data-type name arguments* Format a description for a typed operation in object-oriented programming.

`@deftypemethod[x]` *class data-type name arguments* Format a description for a typed method in object-oriented programming; equivalent to '`@deftypeop {Method on} ...`'.

`@deftypecv[x]` *category class datatype name* Like `@defcv`, but for typed class variables.

`@deftypeivar[x]` *class data-type name* Like `@defivar`, but for typed instance variables.

Conditionally (in)visible text

`@comment comment` Begin a comment in Texinfo. The rest of the line is not visible in any output.

`@c comment` Synonym for `@comment`. DEL also starts a comment.

`@ignore` Begin text that will not appear in any output. Pair with `@end ignore`.

Output formats

`@ifdocbook` Begin text to appear only in Docbook output. Pair with `@end ifdocbook`.

`@ifhtml` Begin text to appear only in HTML output. Pair with `@end ifhtml`.

`@ifinfo` Begin text to appear in both Info and (for historical compatibility) plain text output. Pair with `@end ifinfo`.

`@ifplaintext` Begin text to appear only in plain text output. Pair with `@end ifplaintext`.

`@iftex` Begin text to appear only in the \TeX output. Pair with `@end iftex`.

`@ifxml` Begin text that will appear only in the XML output. Pair with `@end ifxml`.

`@ifnotdocbook @ifnothtml @ifnotplaintext @ifnottex @ifnotxml` Begin text to be ignored in one output format but not the others: `@ifnothtml` text is omitted from HTML output, etc. Pair with the corresponding `@end ifnotformat`.

`@ifnotininfo` Like the other `@ifnot...` commands, but omit from plain text output as well as Info. Pair with `@end ifnotininfo`.

Raw formatter text

`@docbook` Enter Docbook completely. Pair with `@end docbook`.

`@html` Enter HTML completely. Pair with `@end html`.

`@tex` Enter \TeX completely (e.g., \ becomes an escape character). Pair with `@end tex`.

`@xml` Enter XML completely. Pair with `@end xml`.

Document variables

`@set name [string]` Define the Texinfo variable *name*, optionally to the value *string*.

`@clear name` Undefine the Texinfo variable *name*.

`@value {name}` Insert the value of the Texinfo variable *name*.

`@ifset name` If the Texinfo variable *name* is set, format the following text. Pair with `@end ifset`.

`@ifclear flag` If the Texinfo variable *name* is not set, format the following text. Pair with `@end ifclear`.

Defining new Texinfo commands

`@alias new=existing` Make the command `@new` a synonym for the existing command `@existing`.

`@macro macroname {params}` Define a new Texinfo command `@macroname`. Pair with `@end macro`.

`@defineenclose newcmd, before, after` Can only be used within `@ifinfo`; create a new command that encloses its argument in the *before* and *after* strings.

File inclusion

`@include filename` Read the contents of Texinfo source file *filename*.
`@verbatiminclude filename` Incorporate the contents of *filename* in the output, exactly as is (in the fixed-width font).

Formatting and headers/footers for TeX

`@allowcodebreaks true|false` Control breaks at - and _ in TeX; default true.
`@finalout` Omit large black rectangles at overfull lines in TeX output.
`@fonttextsize 10|11` Change size of main body font in TeX.

Paper sizes

The default paper size is US letter-size, 8x11"xx.
`@smallbook` Format for a 7x9.25"xx trim size.
`@afourpaper`, `@afivepaper` Format for A4 and A5 paper sizes.
`@afourlatex` A variant for A4 with more whitespace.
`@fourwide` Format for A4 in landscape.
`@pagesizes [width][, height]` Format for the explicitly given page dimensions, *width* by *height*.

Page headers and footers

`@oddfooting [left] @| [center] @| [right]`
`@oddheading [left] @| [center] @| [right]` Specify page footings resp. headings for odd-numbered (right-hand) pages.
`@evenfooting [left] @| [center] @| [right]`
`@evenheading [left] @| [center] @| [right]` Specify page footings resp. headings for even-numbered (left-hand) pages.
`@everyfooting [left] @| [center] @| [right]`
`@everyheading [left] @| [center] @| [right]` Specify page footings resp. headings for every page.
`@thischapter`, `@thischaptername`, `@thischapternum`, `@thisfile`, `@thispage`, `@thistitle` These commands are only allowed in a heading or footing. They stand for, respectively, the number and name of the current chapter (in the format ‘Chapter 1: Title’), the current chapter name only (‘Title’), the current chapter number only (‘1’), the filename, the current page number, and the title of the document.

Document preferences

It is usually best to omit any of these settings from the Texinfo document, thus allowing users to override the defaults when processing the document according to their needs.
`@setchapternewpage on|off|odd` Specify whether chapters start on new pages, and if so, whether on odd-numbered (right-hand) new pages; also set the page headings accordingly.
`@headings on|off|single|double` Turn page headings on or off, specify single-sided or double-sided page headings for printing.
`@exampleindent indent` Indent example-like environments by *indent* number of spaces (perhaps 0).
`@firstparagraphindent insert|none` Indent first paragraphs after section headings, or not.
`@paragraphindent indent` Indent paragraphs by *indent* number of spaces (perhaps 0); preserve source file indentation if *indent* is the string *asis*.
`@kbdinputstyle code|distinct|example` Specify font for `@kbd`: normal `@code` font, a distinct (slanted) typewriter font, or a distinct font only within `@example` and similar.
`@urefbreakstyle after|before|none` Specify how `@uref` should break at special characters; default `after`.

Ending a Texinfo document

`@bye` End reading the Texinfo source. Any following text in the file is ignored.

Editing Texinfo sources in Emacs

The Texinfo major mode in GNU Emacs is for editing Texinfo source files. To enter Texinfo mode, visit a source file with one of the supported extensions. This list of mode commands is from Emacs 22.1.

<code>texinfo-show-structure</code>	C-c C-s
<code>up-list</code>	C-c]
<code>up-list</code>	C-c }
<code>texinfo-insert-quote</code>	"
<code>texinfo-insert-@item</code>	M-RET
<code>texinfo-insert-block</code>	C-c C-o
<code>texinfo-insert-braces</code>	C-c {
<code>texinfo-start-menu-description</code>	C-c C-c C-d
<code>texinfo-insert-@emph</code>	C-c C-c C-e
<code>texinfo-insert-@strong</code>	C-c C-c C-s
<code>texinfo-insert-@code</code>	C-c C-c c
<code>texinfo-insert-@dfn</code>	C-c C-c d
<code>texinfo-insert-@end</code>	C-c C-c e
<code>texinfo-insert-@file</code>	C-c C-c f
<code>texinfo-insert-@item</code>	C-c C-c i
<code>texinfo-insert-@kbd</code>	C-c C-c k
<code>texinfo-insert-@email</code>	C-c C-c m
<code>texinfo-insert-@node</code>	C-c C-c n
<code>texinfo-insert-@noindent</code>	C-c C-c o
<code>texinfo-insert-@quotation</code>	C-c C-c q
<code>texinfo-insert-@samp</code>	C-c C-c s
<code>texinfo-insert-@table</code>	C-c C-c t
<code>texinfo-insert-@uref</code>	C-c C-c u
<code>texinfo-insert-@var</code>	C-c C-c v
<code>texinfo-insert-@example</code>	C-c C-c x
<code>texinfo-all-menus-update</code>	C-c C-u C-a
<code>texinfo-every-node-update</code>	C-c C-u C-e
<code>texinfo-make-menu</code>	C-c C-u RET
<code>texinfo-update-node</code>	C-c C-u C-n
<code>texinfo-master-menu</code>	C-c C-u m
<code>texinfo-format-buffer</code>	C-c C-e C-b
<code>texinfo-format-region</code>	C-c C-e C-r
<code>makeinfo-buffer</code>	C-c RET C-b
<code>kill-compilation</code>	C-c RET C-k
<code>makeinfo-recenter-compilation-buffer</code>	C-c RET C-l
<code>makeinfo-region</code>	C-c RET C-r
<code>ispell-complete-word</code>	M-TAB
<code>texinfo-tex-buffer</code>	C-c C-t C-b
<code>texinfo-delete-from-print-queue</code>	C-c C-t C-d
<code>texinfo-texindex</code>	C-c C-t TAB
<code>tex-kill-job</code>	C-c C-t C-k
<code>tex-recenter-output-buffer</code>	C-c C-t C-l
<code>texinfo-tex-print</code>	C-c C-t C-p
<code>tex-show-print-queue</code>	C-c C-t C-q
<code>texinfo-tex-region</code>	C-c C-t C-r
<code>texinfo-tex-view</code>	C-c C-t C-v
<code>texinfo-quit-job</code>	C-c C-t C-x

Reading Info files

There are two main Info readers: `M-x info (C-h i)` in GNU Emacs, and the standalone `info` program. These basic keystrokes are for standalone Info, but most work in Emacs Info as well.

<code>q</code>	Quit Info altogether.
<code>H</code>	Invoke the Info tutorial.
<code>Up</code>	Move up one line.
<code>Down</code>	Move down one line.
<code>DEL</code>	Scroll backward one screenful.
<code>SPC</code>	Scroll forward one screenful.
<code>Home</code>	Go to the beginning of this node.
<code>End</code>	Go to the end of this node.
<code>TAB</code>	Skip to the next hypertext link.
<code>RET</code>	Follow the hypertext link under the cursor.
<code>1</code>	Go back to the last node seen in this window.
<code>[</code>	Go to the previous node in the document.
<code>]</code>	Go to the next node in the document.
<code>p</code>	Go to the previous node on this level.
<code>n</code>	Go to the next node on this level.
<code>u</code>	Go up one level.
<code>t</code>	Go to the top node of this document.
<code>d</code>	Go to the main ‘directory’ node.
<code>1...9</code>	Pick the first...ninth item in this node’s menu.
<code>0</code>	Pick the last item in this node’s menu.
<code>m</code>	Pick a menu item specified by name.
<code>r</code>	Follow a cross reference specified by name.
<code>g</code>	Go to a node specified by name.
<code>s</code>	Search forward for a specified string.
<code>{</code>	Search for previous occurrence.
<code>}</code>	Search for next occurrence.
<code>i</code>	Search for a specified string in the index, and select the node referenced by the first entry found.
<code>C-g</code>	Cancel the current operation.

The Texinfo reference manual is available via `info texinfo` (and at <http://www.gnu.org/software/texinfo/manual>).

Copyright © 2008 Free Software Foundation, Inc.
for Texinfo version 4.13, 2008
by Robert D. Solimeno

Copying and distribution of this material, with or without modification, are permitted in any medium without royalty provided the copyright notice and this notice are preserved.

For printed copies of the Texinfo manual, visit <http://gnupress.org> or write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.