

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.

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.

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 in a printed manual.

@detailmenu Mark the beginning of a detail menu.

Cross references

@xref {*node*, [*entry*]} a reference that stands for a command with punctuation mandatory.

@pxref {*node*, [*entry*]} **@xref**, but starts with *inside parentheses*.

@ref {*node*, [*entry*]} , but produces only *the* if followed by a punctuation mark.

@inforef {*node*, [*entry*]} an Info file for which a reference is made.

@url {*url*, [*displayed*]} a reference to a uniform resource locator.

@cite {*title*} Indicate a reference to a file; has no Info file; no punctuation.

@comment *text* A comment.

Marking words

@emph {*text*} Emphasize *text*, enclosing in asterisks.

@strong {*text*} Emphasize *text*, where possible; still *strong*.

@var {*metasyntactic*} Mark *metasyntactic* that stands for another word.

@dfn {*term*} Indicate a technical term.

@acronym {*acronym*} Mark *acronym* letters, such as ‘NASA’.

@abbr {*abbreviation*} Mark *abbreviation* ‘Comput.’.

@code {*sample-code*} Mark *sample-code* complete token, or a code fragment.

@file {*filename*} Mark *filename* in a directory, etc.

@command {*command*} Mark *command* such as `ls`.

@env {*environment-name*} Mark *environment-name*, such as `PATH`.

@option {*option-name*} Mark *option-name*, such as `-v` and `--help`.

@kbd {*keyboard-char*} Mark *keyboard-char* typed by users.

@key {*key-name*} Mark *key-name*, such as `Ctrl-C`.

@email {*address*, [*displayed*]} Mark *address*, such as `foo@bar.com`.

`@indicaturl {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`.

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 into to-index` into `to-index` with `@code`.

`@synindex from-index into to-index` into *to-index*; no space.

`@printindex index-name` Print index in two columns in a printed document.

Insertions with @

`@` Insert an at sign.

`@{` Insert a left brace.

`@}` Insert a right brace.

`@comma {}` Insert a comma. If the comma would be an end-of-sentence comma, it is normal, stretchable,

`@:` Tell T_EX to refrain from immediately preceding with a colon, as T_EX normally does.

`@.` Produce a period.

`@?` Produce a question mark.

`@!` Produce an exclamation mark.

`@dmn {dimension}` For T_EX to insert a thin space.

Braces are optional with the exception of `@dmn`.

`@" c` Generate an umlaut.

`@' c` Generate an acute accent.

`@, c` Generate a cedilla.

`@= c` Generate a macron.

`@~ c` Generate a circumflex.

`@‘ c` Generate a grave accent.

`@~ c` Generate a tilde.

`@dotaccent c` Generate a dotted accent.

`@H c` Generate a long dash.

`@ringaccent c` Generate a ringed accent.

`@tieaccent {cc}` Generate a tie accent, to be given in braces.

`@u c` Generate a breve.

`@ubaraccent c` Generate a barred accent.

`@udotaccent c` Generate a dotted accent.

`@v c` Generate a chevron.

`@AA {}, @aa {}` Generate an all caps font.

`@AE {}, @ae {}` Generate an old english font.

`@L {}, @l {}` Generate a long sans serif font.

`@O {}, @o {}` Generate an old english font.

`@OE {}, @oe {}` Generate an old english font.

`@exclamdown {}` Generate a down pointing exclamation mark.

`@questiondown {}` Generate a down pointing question mark.

`@ss {}` Generate the swiss font.

Other text characters and logos

`@dots {}` Generate an ellipsis, ‘...’.
`@enddots {}` Generate an end-of-sentence ellipsis, ‘...’ (more following space in $\mathrm{T}_{\mathrm{E}}\mathrm{X}$.)
`@bullet {}` Generate a large round dot, • (* in Info).
`@euro {}` Generate the Euro currency sign, €. [\[more\]](#)
`@pounds {}` Generate the pounds sterling currency sign, £.
`@minus {}` Generate a minus sign, ‘−’.
`@copyright {}` Generate the copyright symbol, ©.
`@registeredsymbol {}` Generate the registered symbol, ®.
`@TeX {}` Generate the $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ logo.
`@LaTeX {}` Generate the $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ logo.
`@today {}` Insert the current date, in ‘1 Jan 2000’ style.

Glyphs for code examples

`@equiv {}` Indicate equivalence of two forms, \equiv .
`@error {}` Indicate an error message, error .
`@expansion {}` Indicate the result of a macro expansion, \mapsto .
`@point {}` Indicate the position of point in a buffer, *.
`@print {}` Indicate printed output, \dashv .
`@result {}` Indicate the result of an expression, \Rightarrow .

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, \rightarrow .

Math

`@math {math}` Format the given math (inline); \backslash becomes an escape character, and all $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ math is available.
`@\` Inside `@math`, generate a backslash, \backslash .
`@geq {}` Generate a greater-than-or-equal sign, \geq .
`@leq {}` Generate a less-than-or-equal sign, \leq .

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 is on the following lines, ended by ‘`@end defcmdname`’.

Function-like definitions, untyped

`@deffn[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 ‘`@deffn Function ...`’.
`@defmac[x] name arguments` Format a description for a macro; equivalent to ‘`@deffn Macro ...`’.
`@defspeg[x] name arguments` Format a description for a special form; equivalent to ‘`@deffn {Special Form} ...`’.

Function-like definitions, typed

`@deftypefn[x] category data-type name arguments` Like `@deffn`, 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] category class data-type name arguments` Format a description for a typed operation in object-oriented programming; equivalent to ‘`@deftypemethod {Method on} ...`’.
`@deftypecv[x] category class data-type name` Format a description for a typed instance variable associated with a class in object-oriented programming.
`@deftypeivar[x] class data-type name` Format a description for a typed instance variable associated with a class in object-oriented programming.

Conditionally

`@comment comment text` *comment text* is not visible in the output.
`@c comment` Synonym for `@comment`.
`@ignore` Begin text that will be ignored.
`@end ignore` End ignored text.

`@ifdocbook` Begin text that will be included with `@end ifdocbook` if the output is Docbook.
`@ifhtml` Begin text that will be included with `@end ifhtml` if the output is HTML.
`@ifinfo` Begin text that will be included with `@end ifinfo` if the output is Info compatibility) plain text.
`@ifplaintext` Begin text that will be included with `@end ifplaintext` if the output is plain text.
`@iftex` Begin text that will be included with `@end iftex` if the output is TeX.
`@ifxml` Begin text that will be included with `@end ifxml` if the output is XML.
`@ifnotdocbook`, `@ifnothtml`, `@ifnotinfo`, `@ifnotplaintext`, `@ifnottex`, `@ifnotxml` Like the others: `@ifnot` followed by the entity type. Pair with the corresponding `@end`.
`@ifnotinfo` Like the others: `@ifnot` followed by the entity type. Pair with the corresponding `@end`.
`@ifnotplaintext` Like the others: `@ifnot` followed by the entity type. Pair with the corresponding `@end`.
`@ifnottex` Like the others: `@ifnot` followed by the entity type. Pair with the corresponding `@end`.
`@ifnotxml` Like the others: `@ifnot` followed by the entity type. Pair with the corresponding `@end`.

`@docbook` Enter Docbook mode.
`@html` Enter HTML mode.
`@tex` Enter $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ mode.
`@plaintext` Enter plain text mode.
`@xml` Enter XML mode.

`@set name [string]` Set the value of the variable *name* to the value *string*.
`@clear name` Undefine the variable *name*.
`@value {name}` Insert the value of the variable *name*.
`@ifset name` If the variable *name* is set, begin the following text. Pair with `@endifset`.
`@ifclear flag` If the variable *flag* is not set, begin the following text. Pair with `@endifclear`.

Defining new ?

`@alias new=existing` Define *new* as a synonym for the existing command *existing*.
`@macro macroname arguments` Define a macro *macroname* with *arguments*.
`@definfoenclose new=existing` Define *new* as a synonym for the existing command *existing*.
`@ifinfo`; create a new section in the Info file. Pair with `@endifinfo`.
`@before and after` Define a new section in the Info file. Pair with `@end before and after`.

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 T_EX

`@finalout` Tell T_EX to omit the large black rectangles beside over-wide lines.

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.
`@afourwide` 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.
`@paragraphindent indent` Indent paragraphs by *indent* number of spaces (perhaps 0); preserve source file indentation if *indent* is the string *asis*.
`@exampleindent indent` Indent example-like environments by *indent* number of spaces (perhaps 0).
`@kbdinputstyle style` Specify when `@kbd` should use a font distinct from `@code`.

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

There are two main ways to read Texinfo documents: using Emacs, and the standalone `info` program.

q	Quit Info al
H	Invoke the l
Up	Move up on
Down	Move down
DEL	Scroll backw
SPC	Scroll forwa
Home	Go to the b
End	Go to the en
TAB	Skip to the
RET	Follow the h
1	Go back to
[Go to the p
]	Go to the n
P	Go to the p
n	Go to the m
u	Go up one l
t	Go to the t
d	Go to the m
1...9	Pick the fir
0	Pick the las
m	Pick a menu
r	Follow a cro
g	Go to a nod
s	Search forw
{	Search for p
}	Search for n
i	Search for a
	select the no
C-g	Cancel the c

The Texinfo reference manual (and at <http://www.gnu.org/doc/inf>)

Copyright © 2009 Free Software Foundation, Inc.

Copying and distributing copies, or making modifications, are permitted, provided the copyright notice is preserved.

For printed copies of this manual, contact the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.