

NAME

pod2text - Convert POD data to formatted ASCII text

SYNOPSIS

```
pod2text [-aclostu] [--code] [-i indent] [-q quotes] [--stderr] [-w width] [input [output ...]]
```

```
pod2text -h
```

DESCRIPTION

pod2text is a front-end for Pod::Text and its subclasses. It uses them to generate formatted ASCII text from POD source. It can optionally use either termcap sequences or ANSI color escape sequences to format the text.

input is the file to read for POD source (the POD can be embedded in code). If *input* isn't given, it defaults to `STDIN`. *output*, if given, is the file to which to write the formatted output. If *output* isn't given, the formatted output is written to `STDOUT`. Several POD files can be processed in the same **pod2text** invocation (saving module load and compile times) by providing multiple pairs of *input* and *output* files on the command line.

OPTIONS

-a, --alt

Use an alternate output format that, among other things, uses a different heading style and marks `=item` entries with a colon in the left margin.

--code

Include any non-POD text from the input file in the output as well. Useful for viewing code documented with POD blocks with the POD rendered and the code left intact.

-c, --color

Format the output with ANSI color escape sequences. Using this option requires that `Term::ANSIColor` be installed on your system.

-i *indent*, --indent=*indent*

Set the number of spaces to indent regular text, and the default indentation for `=over` blocks. Defaults to 4 spaces if this option isn't given.

-h, --help

Print out usage information and exit.

-l, --loose

Print a blank line after a `=head1` heading. Normally, no blank line is printed after `=head1`, although one is still printed after `=head2`, because this is the expected formatting for manual pages; if you're formatting arbitrary text documents, using this option is recommended.

-m *width*, --left-margin=*width*, --margin=*width*

The width of the left margin in spaces. Defaults to 0. This is the margin for all text, including headings, not the amount by which regular text is indented; for the latter, see `-i` option.

-o, --overstrike

Format the output with overstrike printing. Bold text is rendered as character, backspace, character. Italics and file names are rendered as underscore, backspace, character. Many paggers, such as **less**, know how to convert this to bold or underlined text.

-q *quotes*, --quotes=*quotes*

Sets the quote marks used to surround `C<>` text to *quotes*. If *quotes* is a single character, it is used as both the left and right quote; if *quotes* is two characters, the first character is used as

the left quote and the second as the right quoted; and if *quotes* is four characters, the first two are used as the left quote and the second two as the right quote.

quotes may also be set to the special value *none*, in which case no quote marks are added around `C<>` text.

-s, --sentence

Assume each sentence ends with two spaces and try to preserve that spacing. Without this option, all consecutive whitespace in non-verbatim paragraphs is compressed into a single space.

--stderr

By default, **pod2text** puts any errors detected in the POD input in a POD ERRORS section in the output manual page. If **--stderr** is given, errors are sent to standard error instead and the POD ERRORS section is suppressed.

-t, --termcap

Try to determine the width of the screen and the bold and underline sequences for the terminal from termcap, and use that information in formatting the output. Output will be wrapped at two columns less than the width of your terminal device. Using this option requires that your system have a termcap file somewhere where Term::Cap can find it and requires that your system support termios. With this option, the output of **pod2text** will contain terminal control sequences for your current terminal type.

-u, --utf8

By default, **pod2text** tries to use the same output encoding as its input encoding (to be backward-compatible with older versions). This option says to instead force the output encoding to UTF-8.

Be aware that, when using this option, the input encoding of your POD source must be properly declared unless it is US-ASCII or Latin-1. POD input without an `=encoding` command will be assumed to be in Latin-1, and if it's actually in UTF-8, the output will be double-encoded. See *perlpod(1)* for more information on the `=encoding` command.

-w, --width=*width*, -*width*

The column at which to wrap text on the right-hand side. Defaults to 76, unless **-t** is given, in which case it's two columns less than the width of your terminal device.

DIAGNOSTICS

If **pod2text** fails with errors, see *Pod::Text* and *Pod::Simple* for information about what those errors might mean. Internally, it can also produce the following diagnostics:

-c (`--color`) requires Term::ANSIColor be installed

(F) **-c** or **--color** were given, but Term::ANSIColor could not be loaded.

Unknown option: %s

(F) An unknown command line option was given.

In addition, other *Getopt::Long* error messages may result from invalid command-line options.

ENVIRONMENT

COLUMNS

If **-t** is given, **pod2text** will take the current width of your screen from this environment variable, if available. It overrides terminal width information in TERMCAP.

TERMCAP

If **-t** is given, **pod2text** will use the contents of this environment variable if available to

determine the correct formatting sequences for your current terminal device.

SEE ALSO

Pod::Text, *Pod::Text::Color*, *Pod::Text::Overstrike*, *Pod::Text::Termcap*, *Pod::Simple*, *perlpod(1)*

The current version of this script is always available from its web site at <http://www.eyrie.org/~eagle/software/podlators/>. It is also part of the Perl core distribution as of 5.6.0.

AUTHOR

Russ Allbery <rra@stanford.edu>.

COPYRIGHT AND LICENSE

Copyright 1999, 2000, 2001, 2004, 2006, 2008 Russ Allbery <rra@stanford.edu>.

This program is free software; you may redistribute it and/or modify it under the same terms as Perl itself.