

NAME

Pod::Html - module to convert pod files to HTML

SYNOPSIS

```
use Pod::Html;
pod2html([options]);
```

DESCRIPTION

Converts files from pod format (see *perlpod*) to HTML format. It can automatically generate indexes and cross-references, and it keeps a cache of things it knows how to cross-reference.

FUNCTIONS

pod2html

```
pod2html("pod2html",
         "--podpath=lib:ext:pod:vms",
         "--podroot=/usr/src/perl",
         "--htmlroot=/perl/nmanual",
         "--libpods=perlfunc:perlguts:perlvar:perlrun:perlop",
         "--recurse",
         "--infile=foo.pod",
         "--outfile=/perl/nmanual/foo.html");
```

pod2html takes the following arguments:

backlink

```
--backlink="Back to Top"
```

Adds "Back to Top" links in front of every `head1` heading (except for the first). By default, no backlinks are generated.

cachedir

```
--cachedir=name
```

Creates the item and directory caches in the given directory.

CSS

```
--css=stylesheet
```

Specify the URL of a cascading style sheet. Also disables all HTML/CSS `style` attributes that are output by default (to avoid conflicts).

flush

```
--flush
```

Flushes the item and directory caches.

header

```
--header
--noheader
```

Creates header and footer blocks containing the text of the `NAME` section. By default, no headers are generated.

help

`--help`

Displays the usage message.

`hiddendirs`

`--hiddendirs`
`--nohiddendirs`

Include hidden directories in the search for POD's in podpath if recurse is set. The default is not to traverse any directory whose name begins with `..`. See *podpath* and *recurse*.

[This option is for backward compatibility only. It's hard to imagine that one would usefully create a module with a name component beginning with `..`.]

`htmldir`

`--htmldir=name`

Sets the directory in which the resulting HTML file is placed. This is used to generate relative links to other files. Not passing this causes all links to be absolute, since this is the value that tells Pod::Html the root of the documentation tree.

`htmlroot`

`--htmlroot=name`

Sets the base URL for the HTML files. When cross-references are made, the HTML root is prepended to the URL.

`index`

`--index`
`--noindex`

Generate an index at the top of the HTML file. This is the default behaviour.

`infile`

`--infile=name`

Specify the pod file to convert. Input is taken from STDIN if no infile is specified.

`libpods`

`--libpods=name:...:name`

List of page names (eg, "perlfunc") which contain linkable `=items`.

`netscape`

`--netscape`
`--nonetscape`

Deprecated, has no effect. For backwards compatibility only.

`outfile`

`--outfile=name`

Specify the HTML file to create. Output goes to STDOUT if no outfile is specified.

`podpath`

`--podpath=name:...:name`

Specify which subdirectories of the podroot contain pod files whose HTML converted forms can be linked to in cross references.

podroot

`--podroot=name`

Specify the base directory for finding library pods.

quiet

`--quiet`

`--noquiet`

Don't display *mostly harmless* warning messages. These messages will be displayed by default. But this is not the same as `verbose` mode.

recurse

`--recurse`

`--norecurse`

Recurse into subdirectories specified in podpath (default behaviour).

title

`--title=title`

Specify the title of the resulting HTML file.

verbose

`--verbose`

`--noverbose`

Display progress messages. By default, they won't be displayed.

htmlify

```
htmlify($heading);
```

Converts a pod section specification to a suitable section specification for HTML. Note that we keep spaces and special characters except " , ? (Netscape problem) and the hyphen (writer's problem...).

anchorify

```
anchorify(@heading);
```

Similar to `htmlify()`, but turns non-alphanumerics into underscores. Note that `anchorify()` is not exported by default.

ENVIRONMENT

Uses `$Config{pod2html}` to setup default options.

AUTHOR

Tom Christiansen, <tchrist@perl.com>.

SEE ALSO

perlpod

COPYRIGHT

This program is distributed under the Artistic License.