



## ***Rocky Enterprise Linux 9.2 Manual Pages on command 'HTML::Formatter.3pm'***

**C:\>man HTML::Formatter.3pm**

HTML::Formatter(3pm)      User Contributed Perl Documentation      HTML::Formatter(3pm)

### NAME

HTML::Formatter - Base class for HTML formatters

### VERSION

version 2.12

### SYNOPSIS

```
use HTML::FormatSomething;

my $infile = "whatever.html";
my $outfile = "whatever.file";
open OUT, ">$outfile"
or die "Can't write-open $outfile: $!\n";
print OUT HTML::FormatSomething->format_file(
    $infile,
    'option1' => 'value1',
    'option2' => 'value2',
    ...
);
close(OUT);
```

### DESCRIPTION

HTML::Formatter is a base class for classes that take HTML and format it to some output format. When you take an object of such a base class and call

"\$formatter->format( \$tree )> with an HTML::TreeBuilder (or HTML::Element) object,

they return the appropriately formatted string for the input HTML.

HTML formatters are able to format a HTML syntax tree into various printable formats. Different formatters produce output for different output media. Common for all formatters are that they will return the formatted output when the `format()` method is called. The `format()` method takes a `HTML::Element` object (usually the `HTML::TreeBuilder` root object) as parameter.

## METHODS

`new`

```
my $formatter = FormatterClass->new(  
    option1 => value1, option2 => value2, ...  
);
```

This creates a new formatter object with the given options.

`format_file`

`format_from_file`

```
$string = FormatterClass->format_file(  
    $html_source,  
    option1 => value1, option2 => value2, ...  
);
```

Return a string consisting of the result of using the given class to format the given HTML file according to the given (optional) options. Internally it calls "`SomeClass->new( ... )->format( ... )`" on a new `HTML::TreeBuilder` object based on the given HTML file.

`format_string`

`format_from_string`

```
$string = FormatterClass->format_string(  
    $html_source,  
    option1 => value1, option2 => value2, ...  
);
```

Return a string consisting of the result of using the given class to format the given HTML source according to the given (optional) options. Internally it calls "`SomeClass->new( ... )->format( ... )`" on a new `HTML::TreeBuilder` object based on the given source.

`format`

```
my $render_string = $formatter->format( $html_tree_object );
```

This renders the given HTML object according to the options set for \$formatter.

After you've used a particular formatter object to format a particular HTML tree object, you probably should not use either again.

## SEE ALSO

The three specific formatters:-

HTML::FormatText

Format HTML into plain text

HTML::FormatPS

Format HTML into postscript

HTML::FormatRTF

Format HTML into Rich Text Format

Also the HTML manipulation libraries used - HTML::TreeBuilder, HTML::Element and

HTML::Tree

## INSTALLATION

See perlmodinstall for information and options on installing Perl modules.

## BUGS AND LIMITATIONS

You can make new bug reports, and view existing ones, through the web interface at <http://rt.cpan.org/Public/Dist/Display.html?Name=HTML-Format>.

## AVAILABILITY

The project homepage is <https://metacpan.org/release/HTML-Format>.

The latest version of this module is available from the Comprehensive Perl Archive Network (CPAN). Visit <http://www.perl.com/CPAN/> to find a CPAN site near you, or see <https://metacpan.org/module/HTML::Format/>.

## AUTHORS

? Nigel Metheringham <nigelm@cpan.org>

? Sean M Burke <sburke@cpan.org>

? Gisle Aas <gisle@ActiveState.com>

## COPYRIGHT AND LICENSE

This software is copyright (c) 2015 by Nigel Metheringham, 2002-2005 Sean M Burke, 1999-2002 Gisle Aas.

This is free software; you can redistribute it and/or modify it under the same terms as the Perl 5 programming language system itself.

