



Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'HTTP::Body.3pm'

\$ man HTTP::Body.3pm

HTTP::Body(3pm) User Contributed Perl Documentation HTTP::Body(3pm)

NAME

HTTP::Body - HTTP Body Parser

SYNOPSIS

```
use HTTP::Body;

sub handler : method {

    my ( $class, $r ) = @_;

    my $content_type = $r->headers_in->get('Content-Type');

    my $content_length = $r->headers_in->get('Content-Length');

    my $body = HTTP::Body->new( $content_type, $content_length );

    my $length = $content_length;

    while ( $length ) {

        $r->read( my $buffer, ( $length < 8192 ) ? $length : 8192 );

        $length -= length($buffer);

        $body->add($buffer);

    }

    my $uploads = $body->upload;    # hashref

    my $params = $body->param;    # hashref

    my $param_order = $body->param_order # arrayref

    my $body = $body->body;    # IO::Handle

}
```

DESCRIPTION

HTTP::Body parses chunks of HTTP POST data and supports application/octet-stream,

application/json, application/x-www-form-urlencoded, and multipart/form-data.

Chunked bodies are supported by not passing a length value to new().

It is currently used by Catalyst to parse POST bodies.

NOTES

When parsing multipart bodies, temporary files are created to store any uploaded files.

You must delete these temporary files yourself after processing them, or set

`$body->cleanup(1)` to automatically delete them at DESTROY-time.

METHODS

`new` Constructor. Takes content type and content length as parameters, returns a `HTTP::Body` object.

`add` Add string to internal buffer. Will call `spin` unless `done`. returns length before adding self.

`body`
accessor for the body.

`chunked`
Returns 1 if the request is chunked.

`cleanup`
Set to 1 to enable automatic deletion of temporary files at DESTROY-time.

`content_length`
Returns the content-length for the body data if known. Returns -1 if the request is chunked.

`content_type`
Returns the content-type of the body data.

`init`
return self.

`length`
Returns the total length of data we expect to read if known. In the case of a chunked request, returns the amount of data read so far.

`trailing_headers`
If a chunked request body had trailing headers, `trailing_headers` will return an `HTTP::Headers` object populated with those headers.

`spin`
Abstract method to spin the io handle.

state

Returns the current state of the parser.

param

Get/set body parameters.

upload

Get/set file uploads.

part_data

Just like 'param' but gives you a hash of the full data associated with the part in a multipart type POST/PUT. Example:

```
{
  data => "test",
  done => 1,
  headers => {
    "Content-Disposition" => "form-data; name=\"arg2\"",
    "Content-Type" => "text/plain"
  },
  name => "arg2",
  size => 4
}
```

tmpdir

Specify a different path for temporary files. Defaults to the system temporary path.

param_order

Returns the array ref of the param keys in the order how they appeared on the body

SUPPORT

Since its original creation this module has been taken over by the Catalyst development team. If you want to contribute patches, these will be your primary contact points:

IRC:

Join #catalyst-dev on irc.perl.org.

Mailing Lists:

<http://lists.scsys.co.uk/cgi-bin/mailman/listinfo/catalyst-dev>

AUTHOR

Christian Hansen, "chansen@cpan.org"

Sebastian Riedel, "sri@cpan.org"

Andy Grundman, "andy@hybridized.org"

CONTRIBUTORS

Simon Elliott "cpan@papercreatures.com"

Kent Fredric <kentnl@cpan.org>

Christian Walde

Torsten Raudssus <torsten@raudssus.de>

LICENSE

This library is free software. You can redistribute it and/or modify it under the same terms as perl itself.

perl v5.32.0

2021-01-02

HTTP::Body(3pm)