

Table of Contents

package TWiki::Request.....	1
ClassMethod new ([\${initializer}).....	1
ObjectMethod action () -> \$action.....	1
ObjectMethod method ([\${method}] -> \$method.....	1
ObjectMethod request_method ([\${method}] -> \$method.....	1
ObjectMethod pathInfo ([\${path}] -> \$path.....	1
ObjectMethod protocol () -> \$protocol.....	2
ObjectMethod uri ([\${uri}] -> \$uri.....	2
ObjectMethod queryString () -> \$query_string.....	2
ObjectMethod url ([-full=>1,.....	2
ObjectMethod secure ([\${secure}] -> \$secure.....	2
ObjectMethod remoteAddress ([\${ip}] -> \$ip.....	2
ObjectMethod remoteUser ([\${userName}] -> \$userName.....	2
ObjectMethod serverPort ([\${userName}] -> \$userName.....	2
ObjectMethod queryParam ([-name=>\$name,-value=>\$value].....	3
ObjectMethod bodyParam ([-name=>\$name,-value=>\$value].....	3
ObjectMethod param ([-name=>\$name,-value=>\$value].....	3
ObjectMethod cookie (\$name[, \$value, \$path, \$secure, \$expires]) -> \$value.....	3
ObjectMethod delete (@paramNames).....	4
ObjectMethod deleteAll ().....	4
ObjectMethod header ([-name=>\$name,-value=>\$value].....	4
ObjectMethod save (\$fh).....	4
ObjectMethod load (\$fh).....	4
ObjectMethod upload (\$name) -> \$handle.....	4
ObjectMethod uploadInfo (\$fname) -> \$headers.....	5
ObjectMethod tmpFileName (\$fname) -> \$tmpFileName.....	5
ObjectMethod uploads ([\%uploads]) -> \$hashref.....	5
ObjectMethod http ([\${header}] -> \$valueDEPRECATED.....	5
ObjectMethod https ([\${name}] -> \$value \$secureDEPRECATED.....	5
ObjectMethod userAgent () -> \$userAgent;.....	5
ObjectMethod referer ().....	5

package TWiki::Request

Class to encapsulate request data.

Fields:

- `action` action requested (view, edit, save, ...)
- `cookies` hashref whose keys are cookie names and values are CGI::Cookie objects
- `headers` hashref whose keys are header name
- `request_method` request method (GET, HEAD, POST)
- `param` hashref of parameters, both query and body ones
- `param_list` arrayref with parameter names in received order
- `path_info` path_info of request (eg. /WebName/TopciName)
- `remote_address` Client's IP address
- `remote_user` Remote HTTP authenticated user
- `secure` Boolean value about use of encryption
- `server_port` Port that the webserver listens on
- `uploads` hashref whose keys are parameter name of uploaded files
- `uri` the request uri

ClassMethod new ([\$initializer])

Constructs a TWiki::Request object.

- `$initializer` - may be a filehandle or hashref.
 - ◆ If it's a filehandle, it'll be used to reload the TWiki::Request object. See `save` method. Note: Restore only parameters
 - ◆ It can be a hashref whose keys are parameter names. Values may be arrayref's to multivalued parameters. Same note as above.

ObjectMethod action () -> \$action

Gets/Sets action requested (view, edit, save, ...)

ObjectMethod method ([\$method]) -> \$method

DEPRECATED. Sets/Gets request method (GET, HEAD, POST). Same as `request_method()` method.

ObjectMethod request_method ([\$method]) -> \$method

Sets/Gets request method (GET, HEAD, POST).

ObjectMethod pathInfo ([\$path]) -> \$path

Sets/Gets request path info.

Called without parameters returns current pathInfo.

There is a `path_info()` alias for compatibility with CGI.

ObjectMethod protocol () -> \$protocol

Returns 'https' if secure connection. 'http' otherwise.

ObjectMethod uri ([\$uri]) -> \$uri

Gets/Sets request uri.

ObjectMethod queryString () -> \$query_string

Returns query_string part of request uri, if any.

query_string() alias provided for compatibility with CGI.

ObjectMethod url ([-full=>1,

-base => 1, -absolute => 1, -relative => 1, -path => 1, -query => 1]) -> \$url

Returns many url info.

- If called without parameters or with -full => 1 returns full url, e.g. http://twiki.org/cgi-bin/view
- If called with -base => 1 returns base url, e.g. http://twiki.org
- -absolute => 1 returns absolute action path, e.g. /cgi-bin/view
- -relative => 1 returns relative action path, e.g. view
- -path => 1, -query => 1 also includes path info and query string respectively

Reasonably compatible with CGI corresponding method. Doesn't support -rewrite. See Item5914.

ObjectMethod secure ([\$secure]) -> \$secure

Gets/Sets connection's secure flag.

ObjectMethod remoteAddress ([\$ip]) -> \$ip

Gets/Sets client IP address.

remote_addr() alias for compatibility with CGI.

ObjectMethod remoteUser ([\$userName]) -> \$userName

Gets/Sets remote user's name.

remote_user() alias for compatibility with CGI.

ObjectMethod serverPort ([\$userName]) -> \$userName

Gets/Sets server user's name.

server_port() alias for compatibility with CGI.

ObjectMethod protocol () -> \$protocol

ObjectMethod queryParam

([-name=>\$name, -value=>\$value |

-name => \$name, -values => [\$v1, \$v2, ...] | \$name, \$v1, \$v2, ... | name, [\$v1, \$v2, ...]]) ->
 @paramNames | @values | \$firstValue

This method is used by engines, during its prepare phase. Should not be used anywhere else. Since bodyParam must exist and it has different semantics from param method, this one exists for symmetry, and could be modified in the future, so it could be possible to get query and body parameters independently.

ObjectMethod bodyParam

([-name=>\$name, -value=>\$value |

-name => \$name, -values => [\$v1, \$v2, ...] | \$name, \$v1, \$v2, ... | name, [\$v1, \$v2, ...]]) ->
 @paramNames | @values | \$firstValue

Adds parameters passed within request body. It keeps previous values, but places new ones first. Should be called only by engines. Otherwise use param() method.

ObjectMethod param ([-name=>\$name, -value=>\$value |

-name => \$name, -values => [\$v1, \$v2, ...] | \$name, \$v1, \$v2, ... | name, [\$v1, \$v2, ...]]) ->
 @paramNames | @values | \$firstValue

- Called without parameters returns all parameter names
- Called only with parameter name or with -name => 'name'
 - ◆ In list context returns all associated values (maybe empty list)
 - ◆ In scalar context returns first value (maybe undef)
- Called with name and list of values or with -name => 'name', -value => 'value' or -name => 'name', -values => [...] sets parameter value

Reasonably compatible with CGI.

ObjectMethod cookie

(\$name [, \$value, \$path, \$secure, \$expires]) ->
 \$value

- If called without parameters returns a list of cookie names.
- If called only with \$name parameter returns value of cookie with that name or undef if it doesn't exist.
- If called with defined \$value and other parameters returns a CGI::Cookie object created with those parameters. Doesn't store this new created cookie within request object. This way for compatibility with CGI.

ObjectMethod cookies(\%cookies) -> \$hashref

Gets/Sets cookies hashref. Keys are cookie names and values CGI::Cookie objects.

ObjectMethod delete (@paramNames)

Deletes parameters from request.

Delete() alias provided for compatibility with CGI

ObjectMethod deleteAll ()

Deletes all parameter name and value(s).

delete_all() alias provided for compatibility with CGI.

ObjectMethod header

([-name=>\$name, -value=>\$value |

-name => \$name, -values => [\$v1, \$v2, ...] | \$name, \$v1, \$v2, ... | name, [\$v1, \$v2, ...]) ->
@paramNames | @values | \$firstValue

Gets/Sets a header field:

- Called without parameters returns all header field names
- Called only with header field name or with -name => 'name'
 - ◆ In list context returns all associated values (maybe empty list)
 - ◆ In scalar context returns the first value (maybe undef)
- Called with name and list of values or with -name => 'name', -value => 'value' or -name => 'name', -values => [...] sets header field value

Not compatible with CGI, since CGI correspondent is a response write method. CGI scripts obtain headers from %ENV or http method. %ENV is not available and must be replaced by calls to this and other methods of this class. http is provided for compatibility, but is deprecated. Use this one instead.

Calls to CGI header method must be replaced by calls to TWiki::Response header method.

ObjectMethod save (\$fh)

Saves object state to filehandle. Object may be loaded latter passing \$fh to new constructor or by calling load().

ObjectMethod load (\$fh)

Loads object state from filehandle, probably created with a previous save().

ObjectMethod upload (\$name) -> \$handle

Called with file name parameter returns an open filehandle to uploaded file.

- If called without parameters returns a list filenames, app trying to upload

ObjectMethod uploadInfo (\$fname) -> \$headers

Returns a hashref to information about uploaded files as sent by browser.

ObjectMethod tmpFileName (\$fname) -> \$tmpFileName

Returns the name of temporarily created file to store uploaded \$fname.

\$fname may be obtained by calling `param()` with form field name.

ObjectMethod uploads ([\%uploads]) -> \$hashref

Gets/Sets request uploads field. Keys are uploaded file names, as sent by browser, and values are TWiki::Request::Upload objects.

ObjectMethod http ([\$header]) -> \$valueDEPRECATED

Called without parameters returns a list of all available header field names.

Given a field name returns value associated.

`http('HTTP_USER_AGENT');` `http('User-Agent')` and `http('User_Agent')` are equivalent.

Please, use `header()` instead. Present only for compatibility with CGI.

ObjectMethod https ([\$name]) -> \$value || \$secureDEPRECATED

Similar to `http()` method above. Called with no parameters returns secure flag.

Please, use `header()` and `secure()` instead. Present only for compatibility with CGI.

ObjectMethod userAgent () -> \$userAgent;

Convenience method to get User-Agent string.

`user_agent()` alias provided for compatibility with CGI.

ObjectMethod referer ()

Convenience method to get Referer uri.

This topic: TWiki > TWikiRequestDotPm
Topic revision: r1 - 2010-05-29 - TWikiContributor



Copyright © 1999-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback

Note: Please contribute updates to this topic on TWiki.org at TWiki:TWiki.TWikiRequestDotPm.