

ClickBlocks\Net\Response

General information

Inheritance	no
Child classes	no
Interfaces	no
Source	Framework/net/response.php

Class provides access to headers of the last HTTP-response of the server and allows to form the current server HTTP-response.

Public static methods

getInstance()

```
public static ClickBlocks\Net\Response getInstance()
```

Returns an instance of the class.

getMessage()

```
public static string getMessage(integer $status)
```

\$status	integer	status of the HTTP-response.
-----------------	---------	------------------------------

Returns status message of the server response by its status code. If such status code does not exist the method returns FALSE.

Public non-static properties

headers

```
public ClickBlocks\Net\Headers $headers
```

Object of class ClickBlocks\Net\Headers that provides access to headers of the server HTTP-response.

version

```
public string $version = '1.1'
```

Version of the HTTP-protocol.

status

```
public integer $status = 200
```

Status of the HTTP-response.

body

```
public string $body
```

Body of the HTTP-response.

Public non-static methods

markAsPrivate()

```
public void markAsPrivate()
```

Marks the response as "private". It makes the response ineligible for serving other clients.

markAsPublic()

```
public void markAsPublic()
```

Marks the response as "public". It makes the response eligible for serving other clients.

markAsNotModified()

```
public void markAsNotModified()
```

Modifies the response so that it conforms to the rules defined for a 304 status code. This sets the status, removes the body, and discards any headers that MUST NOT be included in 304 responses.

getContentType()

```
public String|array getContentType(boolean $withCharset = false)
```

<code>\$withCharset</code>	boolean	if TRUE the method returns an array of the following structure ['type' => ..., 'charset' => ...], otherwise only content type will be returned.
-----------------------------------	---------	---

Returns the content type and/or response charset.

setContentType()

```
public void setContentType(string $type, string $charset = null)
```

<code>\$type</code>	string	the response content type.
<code>\$charset</code>	string	the response charset.

Sets content type header. You can use content type alias instead of some HTTP headers.

getCharset()

```
public string getCharset()
```

Returns the response charset. If the "Content-Type" header with charset is not set the method returns NULL.

setCharset()

```
public void setCharset(string $charset = 'UTF-8')
```

<code>\$charset</code>	string	the response charset.
-------------------------------	--------	-----------------------

Sets the response charset.

getDate()

```
public DateTime getDate()
```

Returns the "Date" header of the response as a DateTime instance. If the "Date" header is not set or not parseable the method returns a DateTime object that represents the current time.

setDate()

```
public void setDate(DateTime|string $date = 'now')
```

\$date	string, DateTime	a DateTime instance or date string.
---------------	------------------	-------------------------------------

Sets the "Date" header value.

getExpires()

```
public DateTime getExpires()
```

Returns the value of the "Expires" header as a DateTime instance.

setExpires()

```
public void setExpires(string|DateTime $date = null)
```

\$date	string, DateTime	a DateTime instance, date string or NULL to remove the header.
---------------	------------------	--

Sets the "Expires" header value. Passing NULL as value will remove the header.

getLastModified()

```
public DateTime getLastModified()
```

Returns the Last-Modified HTTP header as a DateTime instance. The method returns FALSE if the header does not exist.

setLastModified()

```
public void setLastModified(string|DateTime $date = null)
```

\$date	string, DateTime	a DateTime instance, date string or NULL to remove the header.
---------------	------------------	--

Sets the "Last-Modified" HTTP header with a DateTime instance. Passing NULL as value will remove the header.

getAge()

```
public integer getAge()
```

Returns amount of time (in seconds) since the response (or its revalidation) was generated at the origin server.

getMaxAge()

```
public integer getMaxAge()
```

Returns the number of seconds after the time specified in the response's "Date" header when the response should no longer be considered fresh.

setMaxAge()

```
public void setMaxAge(integer $value)
```

\$value	integer	the number of seconds.
----------------	---------	------------------------

Sets the number of seconds after which the response should no longer be considered fresh. This methods sets the cache control "max-age" directive.

setSharedMaxAge()

```
public void setSharedMaxAge(integer $value)
```

\$value	integer	the number of seconds.
----------------	---------	------------------------

Sets the number of seconds after which the response should no longer be considered fresh by shared caches. This methods sets the cache control "s-maxage" directive.

getTTL()

```
public integer getTTL()
```

Returns the response's time-to-live in seconds.

setTTL()

```
public void setTTL(integer $value)
```

\$value	integer	the number of seconds.
----------------	---------	------------------------

Sets the response's time-to-live for shared caches. This method adjusts the cache control "s-maxage" directive.

setClientTTL()

```
public void setClientTTL(integer $value)
```

\$value	integer	the number of seconds.
----------------	---------	------------------------

Sets the response's time-to-live for private client caches. This method adjusts the cache control "max-age" directive.

getEtag()

```
public string getEtag()
```

Returns the literal value of the "ETag" HTTP header.

setEtag()

```
public setEtag(string $etag = null, boolean $weak = false)
```

\$etag	string	the ETag unique identifier or NULL to remove the header.
\$weak	boolean	determines whether you want a weak ETag or not.

Sets the "ETag" header value.

reset()

```
public void reset()
```

Initializes all properties of the class with values from PHP's super globals.

clean()

```
public void clean()
```

Removes all HTTP-headers.

cache()

```
public void cache(integer|boolean $expires)
```

\$expires	integer, boolean	response cache lifetime in seconds or FALSE if it is necessary to turn off the response caching.
------------------	------------------	--

Caches the server response for some time.

redirect()

```
public void redirect(string $url, integer $status = 302)
```

\$url	string	URL to redirect.
\$status	integer	status of the server response.

Redirects immediately to the specified URL. After calling this method, the script terminates.

stop()

```
public void stop(integer $status = 500, string $message = null)
```

\$status	integer	status of the server response.
\$message	string	content of the response body.

Terminates the script execution and sets the status code and message of the server response.

download()

```
public void download(string $path, string $filename = null, string $contentType = null, boolean $deleteAfterDownload = false)
```

\$path	string	the full path to the file.
\$filename	string	the name of the downloading file.
\$contentType	string	the content type of the downloading file.
\$deleteAfterDownload	boolean	determines whether the file should be deleted after download.

Downloads the given file. If the content type is not defined, it will be determined according to content of the file.

send()

```
public void send()
```

Sends HTTP-headers of the server response. Script execution is not terminated as a result of calling of this method.

Protected static properties

codes

```
protected static array $codes = [...]
```

Array of message which matches the codes of the server response.