

- [Manual de PHP](#)
- [Referencia de funciones](#)
- [Otros servicios](#)
- [Network](#)
- [Funciones de red](#)

Change language: Spanish ▼

[Submit a Pull Request](#) [Report a Bug](#)

header

(PHP 4, PHP 5, PHP 7, PHP 8)

header — Enviar encabezado sin formato HTTP

Descripción ¶

header(string \$header, bool \$replace = **true**, int \$response_code = 0): void

header() es usado para enviar encabezados **HTTP** sin formato. Ver la especificación [» HTTP/1.1 specification](#) para más información sobre encabezados **HTTP**.

Recuerde que **header()** debe ser llamado antes de mostrar nada por pantalla, etiquetas HTML, líneas en blanco desde un fichero o desde PHP. Es un error muy común leer código con funciones como [include](#) o [require](#), u otro tipo de funciones de acceso de ficheros que incluyen espacios o líneas en blanco que se muestran antes de llamar a la función **header()**. Sucede el mismo problema cuando se utiliza un solo fichero PHP/HTML.

```
<html>
<?php
/* Esto producirá un error. Fíjese en el html
 * que se muestra antes que la llamada a header() */
header('Location: http://www.example.com/');
exit;
?>
```

Parámetros ¶

header

El encabezado en formato cadena.

Existen dos casos especiales en el uso de header. El primero el encabezado que empieza con la cadena "HTTP/" (las mayúsculas no son importantes), es utilizado para averiguar el código de status HTTP a enviar. Por ejemplo, si se tiene Apache configurado para usar un script en PHP para controlar las peticiones a ficheros no encontrados (usando la directiva ErrorDocument), querrá asegurarse de que el script genera el código de status que corresponde.

```
<?php
// Este ejemplo ilustra el caso especial "HTTP/"
// Alternativas mejores en cases de uso típicos incluyen:
// 1. header($_SERVER["SERVER_PROTOCOL"] . " 404 Not Found");
// (para sobrescribir el mensaje de estado HTTP para los clientes que todavía usan HTTP/1.0)
// 2. http_response_code(404); (para usar el mensaje defecto)
header("HTTP/1.0 404 Not Found");
?>
```

El segundo caso especial es el encabezado "Location:" No solamente envía el encabezado al navegador, sino que también devuelve el código de status (302) REDIRECT al navegador a no ser que el código de status 201 o 3xx ya haya sido enviado.

```
<?php
header("Location: http://www.example.com/"); /* Redirección del navegador */

/* Asegurándonos de que el código interior no será ejecutado cuando se realiza la redirección. */
exit;
?>
```

replace

El parámetro opcional replace indica cuando el encabezado debe reemplazar un encabezado previo similar o añadir un segundo encabezado del mismo tipo. Por defecto lo reemplazará, pero si se pasa **false** como segundo argumento se puede forzar múltiples encabezados del mismo tipo. Por ejemplo:

```
<?php
header('WWW-Authenticate: Negotiate');
header('WWW-Authenticate: NTLM', false);
?>
```

http_response_code

Fuerza el código de respuesta HTTP a un valor específico. Observe que este parámetro solamente tiene efecto si header no está vacío.

Valores devueltos ¶

No devuelve ningún valor.

Ejemplos ¶

Ejemplo #1 Diálogo de descarga

Si se quiere preguntar al usuario si quiere guardar los datos que se están enviando, como un fichero PDF generado, puede usarse el encabezado [» Content-Disposition](#) para proporcionar un nombre de fichero recomendado y forzar al navegador a mostrar el diálogo para guardar el fichero.

```
<?php
// Vamos a mostrar un PDF
header('Content-Type: application/pdf');

// Se llamará downloaded.pdf
header('Content-Disposition: attachment; filename="downloaded.pdf"');

// La fuente de PDF se encuentra en original.pdf
readfile('original.pdf');
?>
```

Ejemplo #2 Directivas caché

Scripts PHP normalmente generan contenido dinámico que no debe ser puesto en caché por el navegador cliente o por ningún proxy caché entre el servidor y el navegador cliente. En muchos proxies y clientes se puede forzar la desactivación de la caché con:

```
<?php
header("Cache-Control: no-cache, must-revalidate"); // HTTP/1.1
header("Expires: Sat, 26 Jul 1997 05:00:00 GMT"); // Fecha en el pasado
?>
```

Nota:

Seguramente encontrará que sus páginas no están en caché incluso si no se usan los encabezados mencionados más arriba. Existen varias opciones que los usuarios pueden cambiar en sus navegadores para cambiar el comportamiento por defecto de la caché. Al enviar los encabezados mencionados más arriba, se sobrescriban cualquiera de las opciones que intentan guardar en caché lo que muestre su script.

Adicionalmente, las opciones de configuración de [session_cache_limiter\(\)](#) y `session.cache_limiter` pueden ser usadas para generar automáticamente encabezados caché cuando se están usando sesiones.

Notas ¶

Nota:

Las cabeceras sólo serán accesibles y mostradas si se utiliza un SAPI que lo soporte.

Nota:

Se puede usar búfer de salida para evitar este problema, procesando en el servidor todo el contenido a mostrar en el navegador antes de ser enviado. Se puede hacer esto usando [ob_start\(\)](#) y [ob_end_flush\(\)](#) en el script, o definiendo la directiva `output_buffering` en el *php.ini* o en los ficheros de configuración del servidor.

Nota:

El encabezado con el status HTTP siempre será el primero a ser enviado al cliente, sin tener en cuenta si está usando **header()** con el status en primer lugar o no. El status puede ser sobrescrito pasando a **header()** un nuevo status en cualquier momento a no ser que los encabezados HTTP ya hayan sido enviados.

Nota:

Existe un bug en Microsoft Internet Explorer 4.01 que hace que no funcione. No existe una solución. También hay un bug en Microsoft Internet Explorer 5.5 que también sucede lo mismo, puede ser resuelto actualizando a Service Pack 2 o superior.

Nota:

HTTP/1.1 requiere una URI absoluta como argumento para [» Location](#): incluyendo el esquema, nombre del host y ruta absoluta, pero algunos clientes aceptan también URIs relativas. Se puede usar [\\$_SERVER\['HTTP_HOST'\]](#), [\\$_SERVER\['PHP_SELF'\]](#) y [dirname\(\)](#) para crear una URI absoluta desde una relativa:

```
<?php
/* Redirecciona a una página diferente en el mismo directorio el cual se hizo la petición */
$host = $_SERVER['HTTP_HOST'];
$uri = rtrim(dirname($_SERVER['PHP_SELF']), '/\\');
$extra = 'mypage.php';
header("Location: http://$host$uri/$extra");
exit;
?>
```

Nota:

El ID de sesión no es pasado con el encabezado Location incluso si [session.use_trans_sid](#) está habilitado. Debe ser pasado manualmente usando la constante **SID**.

Ver también [¶](#)

- [headers_sent\(\)](#) - Comprueba si o donde han enviado cabeceras
- [setcookie\(\)](#) - Enviar una cookie
- [http_response_code\(\)](#) - Obtener o establecer el código de respuesta HTTP
- [header_remove\(\)](#) - Elimina encabezados previamente establecidos
- La sección [Autenticación HTTP](#)

[+ add a note](#)

User Contributed Notes 32 notes

[up](#)

[down](#)

239

[mjt at jpeto dot net ¶](#)

13 years ago

I strongly recommend, that you use

```
header($_SERVER["SERVER_PROTOCOL"]." 404 Not Found");
```

instead of

```
header("HTTP/1.1 404 Not Found");
```

I had big troubles with an Apache/2.0.59 (Unix) answering in HTTP/1.0 while I (accidentally) added a "HTTP/1.1 200 Ok" - Header.

Most of the pages were displayed correct, but on some of them apache added weird content to it:

A 4-digits HexCode on top of the page (before any output of my php script), seems to be some kind of checksum, because it changes from page to page and browser to browser. (same code for same page and browser)

"0" at the bottom of the page (after the complete output of my php script)

It took me quite a while to find out about the wrong protocol in the HTTP-header.

[up](#)

[down](#)

153

[Marcel G ¶](#)

12 years ago

Several times this one is asked on the net but an answer could not be found in the docs on php.net ...

If you want to redirect an user and tell him he will be redirected, e. g. "You will be redirected in about 5 secs. If not, click here." you cannot use `header('Location: ...')` as you can't sent any output before the headers are sent.

So, either you have to use the HTML meta refresh thingy or you use the following:

```
<?php
    header( "refresh:5;url=wherever.php" );
    echo 'You\'ll be redirected in about 5 secs. If not, click <a href="wherever.php">here</a>.';
?>
```

Hth someone

[up](#)
[down](#)

84

[Dylan at WeDefy dot com](#)

15 years ago

A quick way to make redirects permanent or temporary is to make use of the `$http_response_code` parameter in `header()`.

```
<?php
// 301 Moved Permanently
header("Location: /foo.php",TRUE,301);

// 302 Found
header("Location: /foo.php",TRUE,302);
header("Location: /foo.php");

// 303 See Other
header("Location: /foo.php",TRUE,303);

// 307 Temporary Redirect
header("Location: /foo.php",TRUE,307);
?>
```

The HTTP status code changes the way browsers and robots handle redirects, so if you are using `header(Location:)` it's a good idea to set the status code at the same time. Browsers typically re-request a 307 page every time, cache a 302 page for the session, and cache a 301 page for longer, or even indefinitely. Search engines typically transfer "page rank" to the new location for 301 redirects, but not for 302, 303 or!! 307. If the status code is not specified, `header('Location:')` defaults to 302.

[up](#)
[down](#)

41

[*mandor at mandor dot net*](#)**16 years ago**

When using PHP to output an image, it won't be cached by the client so if you don't want them to download the image each time they reload the page, you will need to emulate part of the HTTP protocol.

Here's how:

```
<?php

// Test image.
$fn = '/test/foo.png';

// Getting headers sent by the client.
$headers = apache_request_headers();

// Checking if the client is validating his cache and if it is current.
if (isset($headers['If-Modified-Since']) && (strtotime($headers['If-Modified-Since']) == filemtime($fn))) {
    // Client's cache IS current, so we just respond '304 Not Modified'.
    header('Last-Modified: '.gmdate('D, d M Y H:i:s', filemtime($fn)).' GMT', true, 304);
} else {
    // Image not cached or cache outdated, we respond '200 OK' and output the image.
    header('Last-Modified: '.gmdate('D, d M Y H:i:s', filemtime($fn)).' GMT', true, 200);
    header('Content-Length: '.filesize($fn));
    header('Content-Type: image/png');
    print file_get_contents($fn);
}

?>
```

That way foo.png will be properly cached by the client and you'll save bandwidth. :)

[up](#)
[down](#)

6

[*php at ober-mail dot de*](#)**3 years ago**

Since PHP 5.4, the function `http_response_code()` can be used to set the response code instead of using the `header()` function, which requires to also set the correct protocol version (which can lead to problems, as seen in other comments).

[up](#)

[down](#)

25

[bebertjean at yahoo dot fr ¶](#)**14 years ago**

If using the 'header' function for the downloading of files, especially if you're passing the filename as a variable, remember to surround the filename with double quotes, otherwise you'll have problems in Firefox as soon as there's a space in the filename.

So instead of typing:

```
<?php
    header("Content-Disposition: attachment; filename=" . basename($filename));
?>
```

you should type:

```
<?php
    header("Content-Disposition: attachment; filename=\"\" . basename($filename) . "\"");
?>
```

If you don't do this then when the user clicks on the link for a file named "Example file with spaces.txt", then Firefox's Save As dialog box will give it the name "Example", and it will have no extension.

See the page called "Filenames_with_spaces_are_truncated_upon_download" at

<http://kb.mozillazine.org/> for more information. (Sorry, the site won't let me post such a long link...)

[up](#)[down](#)

6

[tim at sharpwebdevelopment dot com ¶](#)**4 years ago**

The header call can be misleading to novice php users.

when "header call" is stated, it refers the the top leftmost position of the file and not the "header()" function itself.

"<?php" opening tag must be placed before anything else, even whitespace.

[up](#)[down](#)

6

[nospam at nospam dot com ¶](#)**6 years ago**

```
<?php
```



```
// Response codes behaviors when using
header('Location: /target.php', true, $code) to forward user to another page:

$code = 301;
// Use when the old page has been "permanently moved and any future requests should be sent to the target page instead. PageRank may be
transferred."

$code = 302; (default)
// "Temporary redirect so page is only cached if indicated by a Cache-Control or Expires header field."

$code = 303;
// "This method exists primarily to allow the output of a POST-activated script to redirect the user agent to a selected resource. The new URI
is not a substitute reference for the originally requested resource and is not cached."

$code = 307;
// Beware that when used after a form is submitted using POST, it would carry over the posted values to the next page, such if target.php
contains a form processing script, it will process the submitted info again!

// In other words, use 301 if permanent, 302 if temporary, and 303 if a results page from a submitted form.
// Maybe use 307 if a form processing script has moved.
```

?>

[up](#)

[down](#)

7

[yjf_victor](#)

6 years ago

According to the RFC 6226 (<https://tools.ietf.org/html/rfc6226>), the only way to send Content-Disposition Header with encoding is:

```
Content-Disposition: attachment;
    filename*= UTF-8''%e2%82%ac%20rates
```

for backward compatibility, what should be sent is:

```
Content-Disposition: attachment;
    filename="EURO rates";
    filename*=utf-8''%e2%82%ac%20rates
```

As a result, we should use

```
<?php
$filename = '中文文件名.exe'; // a filename in Chinese characters

$contentDispositionField = 'Content-Disposition: attachment; '
    . sprintf('filename="%s"; ', rawurlencode($filename))
    . sprintf("filename*=utf-8''%s", rawurlencode($filename));

header('Content-Type: application/octet-stream');

header($contentDispositionField);

readfile('file_to_download.exe');
?>
```

I have tested the code in IE6-10, firefox and Chrome.

[up](#)

[down](#)

10

[sk89q_¶](#)

14 years ago

You can use HTTP's etags and last modified dates to ensure that you're not sending the browser data it already has cached.

```
<?php
$last_modified_time = filemtime($file);
$etag = md5_file($file);

header("Last-Modified: ".gmdate("D, d M Y H:i:s", $last_modified_time)." GMT");
header("Etag: $etag");

if (@strtotime($_SERVER['HTTP_IF_MODIFIED_SINCE']) == $last_modified_time ||
    trim($_SERVER['HTTP_IF_NONE_MATCH']) == $etag) {
    header("HTTP/1.1 304 Not Modified");
    exit;
}
?>
```

[up](#)

[down](#)

4

[David ¶](#)

5 years ago

It seems the note saying the URI must be absolute is obsolete. Found on https://en.wikipedia.org/wiki/HTTP_location

«An obsolete version of the HTTP 1.1 specifications (IETF RFC 2616) required a complete absolute URI for redirection.[2] The IETF HTTP working group found that the most popular web browsers tolerate the passing of a relative URL[3] and, consequently, the updated HTTP 1.1 specifications (IETF RFC 7231) relaxed the original constraint, allowing the use of relative URLs in Location headers.»

[up](#)

[down](#)

8

[ben at indietorrent dot org ¶](#)

10 years ago

Be aware that sending binary files to the user-agent (browser) over an encrypted connection (SSL/TLS) will fail in IE (Internet Explorer) versions 5, 6, 7, and 8 if any of the following headers is included:

Cache-control:no-store

Cache-control:no-cache

See: <http://support.microsoft.com/kb/323308>

Workaround: do not send those headers.

Also, be aware that IE versions 5, 6, 7, and 8 double-compress already-compressed files and do not reverse the process correctly, so ZIP files and similar are corrupted on download.

Workaround: disable compression (beyond text/html) for these particular versions of IE, e.g., using Apache's "BrowserMatch" directive. The following example disables compression in all versions of IE:

```
BrowserMatch ".*MSIE.*" gzip-only-text/html
```

[up](#)

[down](#)

6

[dev at omikrosys dot com ¶](#)

13 years ago

Just to inform you all, do not get confused between Content-Transfer-Encoding and Content-Encoding

Content-Transfer-Encoding specifies the encoding used to transfer the data within the HTTP protocol, like raw binary or base64. (binary is more compact than base64. base64 having 33% overhead).

Eg Use:- header('Content-Transfer-Encoding: binary');

Content-Encoding is used to apply things like gzip compression to the content/data.

Eg Use:- `header('Content-Encoding: gzip');`

[up](#)

[down](#)

2

[chris at ocproducts dot com ¶](#)

5 years ago

Note that 'session_start' may overwrite your custom cache headers.

To remedy this you need to call:

```
session_cache_limiter('');
```

...after you set your custom cache headers. It will tell the PHP session code to not do any cache header changes of its own.

[up](#)

[down](#)

8

[shutout2730 at yahoo dot com ¶](#)

14 years ago

It is important to note that headers are actually sent when the first byte is output to the browser. If you are replacing headers in your scripts, this means that the placement of echo/print statements and output buffers may actually impact which headers are sent. In the case of redirects, if you forget to terminate your script after sending the header, adding a buffer or sending a character may change which page your users are sent to.

This redirects to 2.html since the second header replaces the first.

```
<?php
header("location: 1.html");
header("location: 2.html"); //replaces 1.html
?>
```

This redirects to 1.html since the header is sent as soon as the echo happens. You also won't see any "headers already sent" errors because the browser follows the redirect before it can display the error.

```
<?php
header("location: 1.html");
echo "send data";
header("location: 2.html"); //1.html already sent
?>
```

Wrapping the previous example in an output buffer actually changes the behavior of the script! This is because headers aren't sent until the output buffer is flushed.

```
<?php
ob_start();
header("location: 1.html");
echo "send data";
header("location: 2.html"); //replaces 1.html
ob_end_flush(); //now the headers are sent
?>
```

[up](#)
[down](#)

4

[jp at webgraphe dot com ¶](#)

19 years ago

A call to `session_write_close()` before the statement

```
<?php
    header("Location: URL");
    exit();
?>
```

is recommended if you want to be sure the session is updated before proceeding to the redirection.

We encountered a situation where the script accessed by the redirection wasn't loading the session correctly because the precedent script hadn't the time to update it (we used a database handler).

JP.

[up](#)
[down](#)

0

[David Spector ¶](#)

1 year ago

Please note that there is no error checking for the header command, either in PHP, browsers, or Web Developer Tools.

If you use something like `header('text/javascript');` to set the MIME type for PHP response text (such as for echoed or Included data), you will get an undiagnosed failure.

The proper MIME-setting function is `header('Content-type: text/javascript');`.

[up](#)
[down](#)

2

[mzheng/no-spam-thx/ at ariba dot com ¶](#)

14 years ago

For large files (100+ MBs), I found that it is essential to flush the file content ASAP, otherwise the download dialog doesn't show until a long time or never.

```
<?php
header("Content-Disposition: attachment; filename=" . urlencode($file));
header("Content-Type: application/force-download");
header("Content-Type: application/octet-stream");
header("Content-Type: application/download");
header("Content-Description: File Transfer");
header("Content-Length: " . filesize($file));
flush(); // this doesn't really matter.
```

```
$fp = fopen($file, "r");
while (!feof($fp))
{
    echo fread($fp, 65536);
    flush(); // this is essential for large downloads
}
fclose($fp);
?>
```

[up](#)
[down](#)

-1

[razvan_bc at yahoo dot com ¶](#)

4 years ago

```
<?php
/* This will give an error. Note the output
 * above, which is before the header() call */
header('Location: http://www.example.com/');
exit;
?>
```

this example is pretty good BUT in time you use "exit" the parser will still work to decide what's happening next the "exit" 's action should do ('cause if you check the manual exit works in others situations too).

SO MY POINT IS : you should use :

```
<?php
```

```
header('Location: http://www.example.com/');  
die();
```

```
?>
```

'CAUSE all die function does is to stop the script ,there is no other place for interpretation and the scope you choose to break the action of your script is quickly DONE!!!

there are many situations with others examples and the right choose for small parts of your scrips that make differences when you write your php framework at well!

Thanks Rasmus Lerdorf and his team to wrap off parts of unusual php functionality ,php 7 roolez!!!!

[up](#)

[down](#)

-1

[Angelica Perduta ¶](#)

2 years ago

I made a script that generates an optimized image for use on web pages using a 404 script to resize and reduce original images, but on some servers it was generating the image but then not using it due to some kind of cache somewhere of the 404 status. I managed to get it to work with the following and although I don't quite understand it, I hope my posting here does help others with similar issues:

```
header_remove();  
header("Cache-Control: no-store, no-cache, must-revalidate, max-age=0");  
header("Cache-Control: post-check=0, pre-check=0", false);  
header("Pragma: no-cache");  
// ... and then try redirecting  
// 201 = The request has been fulfilled, resulting in the creation of a new resource however it's still not loading  
// 302 "moved temporarily" does seems to load it!  
header("location:$dst", FALSE, 302); // redirect to the file now we have it
```

[up](#)

[down](#)

-1

[scott at lucentminds dot com ¶](#)

13 years ago

If you want to remove a header and keep it from being sent as part of the header response, just provide nothing as the header value after the header name. For example...

PHP, by default, always returns the following header:

```
"Content-Type: text/html"
```

Which your entire header response will look like

```
HTTP/1.1 200 OK
Server: Apache/2.2.11 (Unix)
X-Powered-By: PHP/5.2.8
Date: Fri, 16 Oct 2009 23:05:07 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
```

If you call the header name with no value like so...

```
<?php

    header( 'Content-Type:' );

?>
```

Your headers now look like this:

```
HTTP/1.1 200 OK
Server: Apache/2.2.11 (Unix)
X-Powered-By: PHP/5.2.8
Date: Fri, 16 Oct 2009 23:05:07 GMT
Connection: close
```

[up](#)

[down](#)

-3

[Vinay Kotekar ¶](#)

8 years ago

Saving php file in ANSI no issues but when saving the file in UTF-8 format for various reasons remember to save the file without any BOM (byte-order mark) support.

Otherwise you will face problem of headers not being properly sent

eg.
 <?php header("Set-Cookie: name=user");?>

Would give something like this :-

Warning: Cannot modify header information - headers already sent by (output started at C:\www\info.php:1) in C:\www\info.php on line 1

[up](#)
[down](#)
 -2

[Cody G.](#)

12 years ago

After lots of research and testing, I'd like to share my findings about my problems with Internet Explorer and file downloads.

Take a look at this code, which replicates the normal download of a Javascript:

```
<?php
if(strpos($_SERVER["HTTP_USER_AGENT"],"MSIE")==false) {
    header("Content-type: text/javascript");
    header("Content-Disposition: inline; filename=\"download.js\"");
    header("Content-Length: ".filesize("my-file.js"));
} else {
    header("Content-type: application/force-download");
    header("Content-Disposition: attachment; filename=\"download.js\"");
    header("Content-Length: ".filesize("my-file.js"));
}
header("Expires: Fri, 01 Jan 2010 05:00:00 GMT");
if(strpos($_SERVER["HTTP_USER_AGENT"],"MSIE")==false) {
    header("Cache-Control: no-cache");
    header("Pragma: no-cache");
}
include("my-file.js");
?>
```

Now let me explain:

I start out by checking for IE, then if not IE, I set Content-type (case-sensitive) to JS and set Content-Disposition (every header is case-sensitive from now on) to inline, because most browsers outside of IE like to display JS inline. (User may change settings). The Content-Length header is required by some browsers to activate download box. Then, if it is IE, the "application/force-download" Content-type is sometimes required to show the download box. Use this if you don't want your PDF to display in the browser (in IE). I use it here to make sure the box

opens. Anyway, I set the Content-Disposition to attachment because I already know that the box will appear. Then I have the Content-Length again.

Now, here's my big point. I have the Cache-Control and Pragma headers sent only if not IE. THESE HEADERS WILL PREVENT DOWNLOAD ON IE!!! Only use the Expires header, after all, it will require the file to be downloaded again the next time. This is not a bug! IE stores downloads in the Temporary Internet Files folder until the download is complete. I know this because once I downloaded a huge file to My Documents, but the Download Dialog box put it in the Temp folder and moved it at the end. Just think about it. If IE requires the file to be downloaded to the Temp folder, setting the Cache-Control and Pragma headers will cause an error!

I hope this saves someone some time!

~Cody G.

[up](#)
[down](#)

-6

[Anonymous](#)
13 years ago

I just want to add, because I see here lots of wrong formatted headers.

1. All used headers have first letters uppercase, so you MUST follow this. For example:

Location, not location

Content-Type, not content-type, nor CONTENT-TYPE

2. Then there MUST be colon and space, like

good: `header("Content-Type: text/plain");`

wrong: `header("Content-Type:text/plain");`

3. Location header MUST be absolute uri with scheme, domain, port, path, etc.

good: `header("Location: http://www.example.com/something.php?a=1");`

4. Relative URIs are NOT allowed

wrong: `Location: /something.php?a=1`

wrong: `Location: ?a=1`

It will make proxy server and http clients happier.

[up](#)

[down](#)

-4

[Refugnic ¶](#)**12 years ago**

My files are in a compressed state (bz2). When the user clicks the link, I want them to get the uncompressed version of the file.

After decompressing the file, I ran into the problem, that the download dialog would always pop up, even when I told the dialog to 'Always perform this operation with this file type'.

As I found out, the problem was in the header directive 'Content-Disposition', namely the 'attachment' directive.

If you want your browser to simulate a plain link to a file, either change 'attachment' to 'inline' or omit it altogether and you'll be fine.

This took me a while to figure out and I hope it will help someone else out there, who runs into the same problem.

[up](#)[down](#)

-4

[bMindful at fleetingiamge dot org ¶](#)**19 years ago**

If you haven't used, HTTP Response 204 can be very convenient. 204 tells the server to immediately terminate this request. This is helpful if you want a javascript (or similar) client-side function to execute a server-side function without refreshing or changing the current webpage. Great for updating database, setting global variables, etc.

```
header("status: 204"); (or the other call)
header("HTTP/1.0 204 No Response");
```

[up](#)[down](#)

-12

[nobileelpirata at hotmail dot com ¶](#)**15 years ago**

This is the Headers to force a browser to use fresh content (no caching) in HTTP/1.0 and HTTP/1.1:

```
<?PHP
header( 'Expires: Sat, 26 Jul 1997 05:00:00 GMT' );
header( 'Last-Modified: ' . gmdate( 'D, d M Y H:i:s' ) . ' GMT' );
header( 'Cache-Control: no-store, no-cache, must-revalidate' );
header( 'Cache-Control: post-check=0, pre-check=0', false );
header( 'Pragma: no-cache' );
```

?>

[up](#)

[down](#)

-13

[*jamie ¶*](#)

14 years ago

The encoding of a file is discovered by the Content-Type, either in the HTML meta tag or as part of the HTTP header. Thus, the server and browser does not need - nor expect - a Unicode file to begin with a BOM mark. BOMs can confuse *nix systems too. More info at

http://unicode.org/faq/utf_bom.html#bom1

On another note: Safari can display CMYK images (at least the OS X version, because it uses the services of QuickTime)

[up](#)

[down](#)

-11

[*er dot ellison dot nyc at gmail dot com ¶*](#)

7 years ago

DO NOT PUT space between location and the colon that comes after that ,

// DO NOT USE THIS :

header("Location : #whatever"); // -> will not work !

// INSTEAD USE THIS ->

header("Location: #wahtever"); // -> will work forever !

[up](#)

[down](#)

-7

[*hamza dot eljaouhari dot etudes at gmail dot com ¶*](#)

4 years ago

// Beware that adding a space between the keyword "Location" and the colon causes an Internal Sever Error

//This line causes the error

7

header('Location : index.php&controller=produit&action=index');

// While It must be written without the space

header('Location: index.php&controller=produit&action=index');

[up](#)

[down](#)

-12

[ASchmidt at Anamera dot net ¶](#)

4 years ago

Setting the "Location: " header has another undocumented side-effect!

It will also disregard any expressly set "Content-Type: " and forces:

"Content-Type: text/html; charset=UTF-8"

The HTTP RFCs don't call for such a drastic action. They simply state that a redirect content SHOULD include a link to the destination page (in which case ANY HTML compatible content type would do). But PHP even overrides a perfectly standards-compliant

"Content-Type: application/xhtml+xml"!

[up](#)

[down](#)

-36

[cedric at gn dot apc dot org ¶](#)

11 years ago

Setting a Location header "returns a REDIRECT (302) status code to the browser unless the 201 or a 3xx status code has already been set". If you are sending a response to a POST request, you might want to look at RFC 2616 sections 10.3.3 and 10.3.4. It is suggested that if you want the browser to immediately GET the resource in the Location header in this circumstance, you should use a 303 status code not the 302 (with the same link as hypertext in the body for very old browsers). This may have (rare) consequences as mentioned in bug 42969.

[+ add a note](#)

- [Funciones de red](#)
 - [checkdnsrr](#)
 - [closelog](#)
 - [dns_check_record](#)
 - [dns_get_mx](#)
 - [dns_get_record](#)
 - [fsockopen](#)
 - [gethostbyaddr](#)
 - [gethostbyname](#)
 - [gethostbyname1](#)
 - [gethostname](#)
 - [getmxrr](#)
 - [getprotobyname](#)
 - [getprotobynumber](#)
 - [getservbyname](#)
 - [getservbyport](#)
 - [header_register_callback](#)

- [header_remove](#)
- [header](#)
- [headers_list](#)
- [headers_sent](#)
- [http_response_code](#)
- [inet_ntop](#)
- [inet_pton](#)
- [ip2long](#)
- [long2ip](#)
- [net_get_interfaces](#)
- [openlog](#)
- [pfsockopen](#)
- [setcookie](#)
- [setrawcookie](#)
- [socket_get_status](#)
- [socket_set_blocking](#)
- [socket_set_timeout](#)
- [syslog](#)

- [Copyright © 2001-2023 The PHP Group](#)
- [My PHP.net](#)
- [Contact](#)
- [Other PHP.net sites](#)
- [Privacy policy](#)

