

Cabeceras HTTP

Cada petición que envía un cliente solicitando una página web va precedida de una cabecera.

Ejemplo: cabecera enviada por navegador de cliente al acceder a www.google.com

```
GET / HTTP/1.1
Host: www.google.com
User-Agent: Mozilla/5.0 (Windows NT 6.0; rv:5.0) Gecko/20100101 Firefox/5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-es;es;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
Cookie: PREF=ID=0c4479bc48cc54b9:U=2153a747ac5d1320:FF=0:TM=1234112738:LM=1296123415:GM=1:
S=wSoNqwpmFJGDC_wn;NID=42=qaDR0eGhuO5H-sGvS67GmQvMsKCxQ8N9U9_L a6rq2m3nWwktAFUTrHF3 P9d-jzAi-
X5j5SrbFFD_dlgLSW60CnMCxiupnBhapGUpLvIFdq3XsJzpOT4xHLtVF4t7zdy_
```

Cada respuesta que devuelve un servidor también va precedida de una cabecera.

Ejemplo: cabecera que envía el servidor de google en respuesta a la petición anterior

```
HTTP/1.1 302 Found
Location: http://www.google.es/
Cache-Control: private
Content-Type: text/html; charset=UTF-8
Date: Sun, 10 Jul 2011 16:35:32 GMT
Server: gws
Content-Length: 218
X-XSS-Protection: 1; mode=block
```

Como ves, nos indica que el URL no existe y redirige nuestro navegador a www.google.es.

Existen muchos tipos de cabeceras HTTP; unas simplemente aportan información, como la fecha, pero otras modifican el comportamiento de nuestro navegador (por ejemplo, Content-Type puede modificar la codificación de caracteres de nuestro navegador).

Desde el punto de vista de la programación en PHP, nos interesa este asunto porque disponemos de la función **header**, con la que podemos enviar cabeceras a nuestros clientes. Esta función recibe un único argumento, que es el texto de la cabecera que deseamos enviar. Por ejemplo, la siguiente instrucción activa la codificación UTF-8 en el navegador del cliente:

```
header ( 'Content-Type: text/html; charset=UTF-8' );
```

Importante la función header debe aparecer delante de cualquier otra información que se envíe al cliente (esto es obvio, pues las cabeceras se envían antes que el contenido de las peticiones). Esto significa que no puede haber ningún echo por delante de un header.

Extraer las cabeceras HTTP del cliente mediante PHP

Si en lugar de generar las cabeceras HTTP de una respuesta mediante PHP quisiéramos averiguar qué cabeceras ha enviado el cliente deberíamos recurrir a la función **apache_request_headers()**, que devuelve un array con todas las cabeceras.

```
<?php
    $x=apache_request_headers();
    foreach($x as $k=>$v)
        echo "en $k viene $v <br>";
?>
```

USOS MAS COMUNES DE LAS CABECERAS HTTP EN PHP

▪ Redirigir al cliente a otra dirección

Para ello solo tenemos que utilizar la cabecera Location.

```
header ( 'Location: acceso_no_autorizado.php' );
```

▪ Mostrar un mensaje y redirigir al cliente a otra dirección

La cabecera Refresh permite establecer un plazo en segundos antes de redirigir al navegador a otra dirección.

```
header ('Refresh: 5; url=http://www.google.es');  
echo 'Lo que busca no existe, le redirigiremos a Google en 5 segundos'
```

▪ Ocultar la versión del intérprete PHP

Muchos robots surcan la red en busca de servidores que tengan versiones desfasadas de PHP con problemas de seguridad conocidos. El servidor envía la versión de PHP en la cabecera **X-Powered-By**, que podemos sobrescribir

```
header( 'X-Powered-By: adivina-adivinanza' );
```

También se puede ocultar versión de PHP asignando valor 0 a la directiva expose_PHP en el php.ini

▪ Ofrecer la descarga de un archivo desde PHP

Hasta ahora, los scripts PHP han generado páginas HTML, pero también se le puede indicar al cliente que va a recibir otro tipo de contenido y utilizar la función readfile para enviársela.

Los navegadores leen el tipo MIME de las cabeceras HTTP para saber cómo deben mostrar los contenidos.

Si mediante PHP indicamos que el tipo MIME es, por ejemplo, Content-type: application/pdf, podremos enviar un documento PDF.

Además, si combinamos esto con la cabecera 'Content-Disposition: attachment; filename = "f1.pdf" ' podremos forzar la descarga del archivo en lugar de que se muestre dentro del navegador.

```
header('Content-type: application/pdf' );  
header('Content-Disposition: attachment; filename="descarga.pdf" ');  
readfile('original.pdf')
```

Con este método podemos mantener en secreto el nombre del fichero original

▪ Generar contenidos diferentes a páginas HTML con PHP

Se puede generar con PHP imágenes, documentos PDF, etc. en tiempo real, es decir, sin tener que leerlos de ningún archivo. Para ello es necesario usar librerías especiales de PHP como GD, PDFlib,...

```
header( "Content-type: image/png");
```

IMAGENES DINAMICAS CON PHP

PHP dispone de una biblioteca de funciones para la generación dinámica de imágenes. Estas funciones son interesantes para generar gráficos estadísticos actualizados al instante

Formatos GIF, PNG, JPEG

Configuración:

- Requiere el uso de la librería php_gd2.dll(Windows) o gd.so (Unix, Linux)
- Debe activarse la extensión correspondiente en el fichero php.ini:
extension= php_gd2.dll
extension= gd.so

La librería gráfica GD está escrita en lenguaje C y permite crear y manipular gráficos fácilmente. Permite importar y exportar gráficos de distinto tipo (GIF, JPG y PNG).

Los archivos GIF y PNG guardan en memoria el conjunto completo de píxeles con su color concreto y comprimen el resultado para que el archivo final no sea muy pesado.

Los archivos JPG también están comprimidos, pero se basan en algoritmos complejos que permiten un mayor rendimiento y la hacen muy propicia para usarse con fotografías.

Tipos MIME

El estándar MIME se utiliza para describir el tipo de archivo que enviamos a través de Internet. Es importante conocer los tipos MIME para trabajar con las librerías GD.

En general, antes de recibir la respuesta del servidor, éste debe enviarnos una cabecera anunciando el tipo de contenido que vamos a recibir usando una cabecera especial: **Content-Type**.

PHP, por defecto, envía el tipo MIME text/html (documento HTML); cuando usemos la librería GD se necesita enviar una cabecera con el tipo de la imagen:

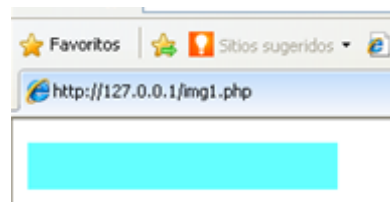
<u>Formato Imagen</u>	<u>Tipo MIME</u>
JPG	image/jpeg
PNG	image/png
GIF	image/gif
BMP	image/bmp

```
<?php
.....
header("Content-Type: image/png") ;
.....
?>
```

El procedimiento general para tratar con imágenes es:

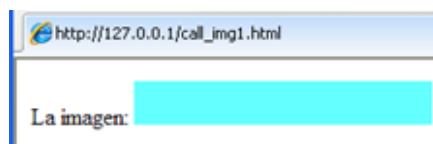
1) Generar la imagen con PHP

```
<?php // script img1.php
$imagen= imagecreate(300, 30); // ancho x alto
// establece color de fondo
$fondo= imagecolorallocate($imagen,100,255,255);
// envía imagen tras la cabecera
header("Content-type: image/png");
imagepng($imagen);
imagedestroy($imagen);
?>
```



2) Invocar imagen en documentos web

```
<html>
....
La imagen: <IMG SRC='img1.php'>
....
```



</html>

Principales funciones para el tratamiento de imagenes

Crear imagenes

```
$imagen= imagecreate(ancho , alto);
$imagen2 = imagecreatefrompng("fichero.png")
$imagen3 = imagecreatefromjpeg("fichero.jpg")

Ej. $cuadro = imagecreatefromjpeg ("nenufares.jpg");
```

Establecer colores

```
$color= imagecolorallocate( $imagen, R, G, B)

Ej. $azul = imagecolorallocate ( $imagen, 0,0,255) ;
```

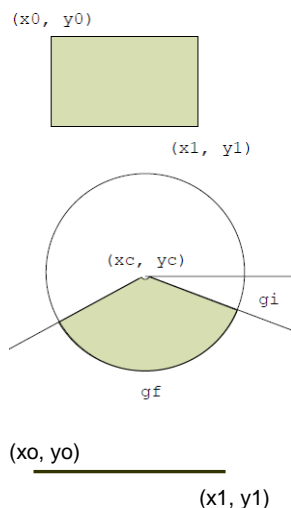
Textos

```
imagestring( $imagen, tamaño, x, y, $texto, $color )
imagefttext( $imagen, tamaño, angulo, x, y, $color, fuente, $texto )

Ej. imagestring ( $imagen2, 20, 100, 50 , 'Hola', $azul) ;

imagefttext ( $imagen3, 20, 30, 100, 50 , $azul, 'arial.ttf', 'Adios') ;
```

Formas Geométricas



```
imagerectangle( $imagen, x0, y0, x1, y1, $color )
imagefilledrectangle( $imagen, x0, y0, x1, y1, $color )

imagearc( $imagen, xc, yc, ancho, alto, gi, gf, $color)
imagefilledarc( $imagen, xc, yc, ancho, alto, gi, gf, $color1, IMG_ARC_PIE)

imageline( $imagen, x0, y0, x1, y1, $color )
```

Obtener tamaño de imagenes

```
getimagesize(fichero)

Devuelve array con datos del tamaño, bits para color, tipo MIME,...
```

Copiar y cambiar tamaño de imagenes

```
imagecopyresized( $d_img, $o_img, d_x, d_y, o_x,o_y, d_ancho, d_alto, o_ancho,o_alto)

d_xx : imagen destino                      o_xx : imagen origen
```

Voltear imagenes

```
imageflip( $imagen, modo )

modo: IMG_FLIP_HORIZONTAL ó IMG_FLIP_VERTICAL ó IMG_FLIP_BOTH
```

Enviar/guardar imagenes


```
imagepng( $imagen [, fichero.png] )
imagejpeg( $imagen [, fichero.jpg] )
imagegif( $imagen [, fichero.gif] )
```

Eliminar imagenes

```
imagedestroy( $imagen )
```


Ejemplos de usos de funciones

```
<?php
$datos = getimagesize('fondo.jpg');
foreach ($datos as $k=>$v)
    echo " indice $k contiene $v <br>";
?>
```

 http://localhost/img_size.php

```
indice 0 contiene 200
indice 1 contiene 100
indice 2 contiene 2
indice 3 contiene width="200" height="100"
indice bits contiene 8
indice channels contiene 3
indice mime contiene image/jpeg
```

```
<?php
$nombre_archivo = 'fondo.jpg';
$porcentaje = 0.5;
header('Content-Type: image/jpeg');
// Obtener nuevos tamaños
$datos = getimagesize($nombre_archivo);
$n_ancho = $datos[0] * $porcentaje;
$n_alto = $datos[1] * $porcentaje;
// Cargar
$thumb = imagecreate ($n_ancho, $n_alto);
$origen = imagecreatefromjpeg($nombre_archivo);
// Cambiar el tamaño
imagecopyresized($thumb, $origen, 0, 0, 0, 0, $n_ancho, $n_alto, $ancho, $alto);
// Enviar
imagejpeg($thumb);
?>
```


 http://localhost/img_resize.php

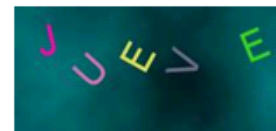


```
<?php
$nombre_fichero = 'phplogo.png';
header('Content-type: image/png');
$im = imagecreatefrompng($nombre_fichero);
// Voltea imagen verticalmente
imageflip($im, IMG_FLIP_VERTICAL);
// Enviar
imagejpeg($im);
imagedestroy($im);
?>
```



```
<?php
$captcha=imagecreatefromjpeg("fondo.jpg");
$texto='JUEVES';
for($i=0;$i<strlen($texto);$i++) {
//Elegir un color
    $color=imagecolorallocate($captcha,rand(0,255),rand(0,255),rand(0,255));
    $ordenada=rand(25,75); //Elegir una ordenada
    $abscisa=$i*40+rand(10,20); //Elegir una abscisa
    $angulo=rand(0,75); //Elegir un ángulo
//Escribir el carácter en la imagen
    $letra= substr($texto,$i,1);
    imagefttext($captcha, 24, $angulo,$abscisa,$ordenada,$color,'arial.ttf', $letra);
}
header("Content-type:image/jpeg");
header("Content-Disposition:inline ; filename=captcha.jpg");
imagejpeg($captcha);
?>
```

 http://localhost/img_captcha.php



ACTIVIDAD: Script que genera formulario que incluye una imagen captcha

Una imagen Captcha (*Completely Automated Public Turing test to tell Computers and Humans Apart* ó prueba de Turing pública y automática para diferenciar máquinas y humanos) es una serie de caracteres que se muestra en formato imagen dentro de un formulario para que el usuario tenga que copiar en un cuadro de texto esos caracteres. Con ello se pretende evitar que los robots puedan enviar datos automáticamente a través de formularios (participar en encuestas, registro en cuentas de correo, envío de correo basura,...).

La idea básica para crear un captcha es generar primero un texto aleatorio; luego, basado en este texto, generar una imagen que se muestra al usuario, y, finalmente, comparar el texto dado por el usuario con el texto generado.

form.php

```
<?php
    session_start();
    if (!isset($_POST['submit'])) {
?>
    <h1> Formulario de registro </h1>
    <form action='form.php' method='POST'>
        Nombre: <input type='text' name='nombre' ><br>
        Email : <input type='text' name='email' ><br>
        Introduzca código de seguridad: <img src='captcha.php' alt="" width='75' height='20' vspace='1' align='top' >
        <input type='text' name='captcha' size='8' maxlength='10'><br>
        <input type='submit' name='submit' value='Enviar'>
    </form>
<?php
    } else {
        If ($_SESSION['captcha'] == $_POST['captcha']) {
?>
        <h2> Registro correcto</h2>
        <?php
        } else {
?>
        <a href='form.php'>Intentar de nuevo </a>
        <?php
        }
    }
?>
```

captcha.php

```
<?php
function textoAzar( $longitud) {    $patron='1234567890abcdefghijklmnopqrstuvwxyz'; $key='';
                                    For ($i=0;$i<$longitud;$i++) { $key .= $patron{rand(0,35) };
                                    }
                                    return $key;
                                    }

    session_start();
    $_SESSION['captcha']= textoAzar(8);
    $img=imagecreate(75,20); // crea una imagen de 75 x 20
    $fondo= imagecolorallocate($img,100,255,255); // estable color de fondo
    $color_texto= imagecolorallocate($img,0,0,255); // establece color para el texto
    imagestring($img,5,0,0,$_SESSION['captcha'],$color_texto); // Coloca texto en la imagen
    // envía imagen tras la cabecera
    header("Content-type: image/png");
    imagepng($img);
?>
```

Ejemplo: aplicación PHP que permita realizar encuesta y mostrar sus resultados con un gráfico de tarta

```
<HTML> <HEAD> <TITLE>Encuesta</TITLE></HEAD>
<BODY>
<?PHP // encuesta.php
$enviar = $_REQUEST['enviar'];
if (isset($enviar)) {
    print("<H1>Encuesta. Voto registrado</H1>\n");
    // Conectar con la base de datos
    $connection = mysql_connect("localhost", "user1", "") or die("No se puede conectar al servidor");
    mysql_select_db("mercados") or die("No se puede seleccionar BD");

    $instruccion = "select votos1, votos2 from votos"; // Obtener votos actuales

    $consulta = mysql_query($instruccion, $connection) or die("Fallo en la selección");
    $resultado = mysql_fetch_array($consulta);

    // Actualizar votos
    $votos1 = $resultado["votos1"]; $votos2 = $resultado["votos2"]; $voto = $_REQUEST['voto'];
    if ($voto == "si") $votos1 = $votos1 + 1;
    else if ($voto == "no") $votos2 = $votos2 + 1;

    $instruccion = "update votos set votos1=$votos1, votos2=$votos2";
    $actualizacion = mysql_query($instruccion, $connection) or die("Fallo en la modificación");

    mysql_close($connection); // Desconectar

    // Mostrar mensaje de agradecimiento
    print("<P>Su voto ha sido registrado. Gracias por participar</P>\n");
    print("<A HREF='encuesta-resultados.php'>Ver resultados</A>\n");
}
else
{
?>
<H1>Encuesta</H1>
<P>¿Cree ud. que el precio de la vivienda seguirá bajando al ritmo actual?</P>

<FORM action="encuesta.php" method="POST">
    <input type="radio" name="voto" value="si" checked>sí<br>
    <input type="radio" name="voto" value="no">no<br><br>
    <input type="submit" name="enviar" value="votar">
</FORM>
<A HREF="encuesta-resultados.php">Ver resultados</A>
<?PHP
}
?>
</BODY>
</HTML>
```

```

<HTML><HEAD> <TITLE>Encuesta. Resultados de la votación</TITLE></HEAD>

<BODY>
<H1>Encuesta. Resultados de la votación</H1>
<?php
// Encuesta-resultados.php
// Conectar con la base de datos
$conn = mysql_connect ("localhost", "user1", "") or die ("No se puede conectar al servidor");
mysql_select_db ("mercados") or die ("No se puede seleccionar BD");

$instruccion = "select * from votos"; // Obtener datos actuales de la votación
$query = mysql_query ($instruccion, $conn) or die ("Fallo en la selección");
$resultado = mysql_fetch_array ($query);

$votos1 = $resultado["votos1"]; $votos2 = $resultado["votos2"]; $totalVotos = $votos1 + $votos2;

// Mostrar gráfico de tarta
print ("<IMG ALT='Tarta' SRC='encuesta-tarta.php?votos1=$votos1&votos2=$votos2'>\n");

print ("<P>Número total de votos emitidos: $totalVotos </P>\n");
print ("<P><A HREF='encuesta.php'>Página de votación</A></P>\n");

mysql_close ($conn); // Desconectar
?>
</BODY>
</HTML>

```

```

<?PHP
// Encuesta-tarta.php
header ("Content-type: image/png");

// Calcular ángulos
$votos1 = $_REQUEST['votos1']; $votos2 = $_REQUEST['votos2']; $totalVotos = $votos1 + $votos2;

$porcentaje1 = round (($votos1/$totalVotos)*100,2); $angulo1 = 3.6 * $porcentaje1;
$porcentaje2 = round (($votos2/$totalVotos)*100,2); $angulo2 = 3.6 * $porcentaje2;

// Crear imagen
$imagen = imagecreate (300, 300);
$colorfondo = imagecolorallocate ($imagen, 203, 203, 203); // CCCCCC
$color1 = imagecolorallocate ($imagen, 255, 0, 0); // FF0000
$color2 = imagecolorallocate ($imagen, 0, 255, 0); // 00FF00
$colortexto = imagecolorallocate ($imagen, 0, 0, 0); // 000000

// Mostrar tarta
imagefilledrectangle ($imagen, 0, 0, 300, 300, $colorfondo);
imagefilledarc ($imagen, 150, 120, 200, 200, 0, $angulo1, $color1, IMG_ARC_PIE);
imagefilledarc ($imagen, 150, 120, 200, 200, $angulo1, $angulo2, $color2, IMG_ARC_PIE);

// Mostrar leyenda
imagefilledrectangle ($imagen, 60, 250, 80, 260, $color1);
$texto1 = "Sí: " . $votos1 . " votos (" . $porcentaje1 . "%)";
imagestring ($imagen, 3, 90, 250, $texto1, $colortexto);
imagefilledrectangle ($imagen, 60, 270, 80, 280, $color2);
$texto2 = "No: " . $votos2 . " votos (" . $porcentaje2 . "%)";
imagestring ($imagen, 3, 90, 270, $texto2, $colortexto);

imagepng ($imagen);
imagedestroy ($imagen);
?>

```