

## Las cookies

De igual modo que ocurría con la función **mail**, no todos los *hosting* tienen habilitada la opción de envío de **cookies**.

Como sabes, las *cookies* son pequeños ficheros que se escriben **en el ordenador del cliente**.

Si utilizas *Internet Explorer* podrás ver que se almacenan como ficheros de texto en un directorio llamado *Archivos temporales de Internet* y que su nombre es de este tipo:

**xxx@nombre[z].txt**

donde **xxx** suele ser el nombre que figura en el registro de Windows como nombre del equipo (el que se pone al instalar Windows); **nombre** suele ser *el nombre del directorio de servidor* desde el que se *envió* la cookie y el número **z** suele ser el ordinal del número de accesos a la página que envía la **cookie**.

*Netscape* trata las **cookies** de distinta forma ya que las almacena en un **único fichero** llamado **cookies.txt** que está en el *subdirectorio del usuario* que se crea en la instalación del programa y que está dentro de otro llamado **users**.

## ¿Cómo enviar cookies?

La instrucción para el envío de *cookies* debe insertarse *al principio de la página* y antes de cualquier etiqueta HTML o línea en blanco. Esto es muy importante, no lo olvides

La sintaxis es la siguiente:

**setcookie(nom, val, exp)**

donde:

**nom** es una cadena que contiene el *nombre de la variable* que recoge el valor de la *cookie*.

**val** es el valor que *se asignará a la variable anterior*. Puede ser tanto numérico como de tipo cadena.

**exp** indica cuál ha de ser la *fecha de caducidad* de la *cookie*.

Suele escribirse: **time()** (hora actual) **más** un número que representa los *segundos* que han de transcurrir hasta que la *cookie* **expire**.

## Una cookie muy simple

Aquí tienes un ejemplo, el más simple, del uso de esta función.

Si ejecutas el ejemplo por primera vez observarás que solo aparecerá el texto *Esto es la galletita*: sin **ningún** valor. Sin embargo, si actualizas el navegador o ejecutas más tarde el ejemplo siguiente aparecerá *Mi regalito* como valor de la variable.

La explicación es la siguiente: las instrucciones PHP se ejecutan en el servidor *antes* de enviar la página al cliente. Eso significa que, al ejecutar por primera vez, se inserta la orden de escritura y se comprueba el valor de la variable, que aun no ha sido creada y por ello aparece en blanco. Será en la actualización –ya se habría producido un envío al navegador y ya se habría escrito la cookie– cuando si se leerá el valor *anterior*.

Siempre que tratemos de visualizar el valor de una cookie estaremos viendo el valor asignado en la petición anterior

```
<?
# setcookie escribe el contenido de la cookie
# en el ordenador del cliente
setcookie("cookie1","Mi regalito",time()+3600);
# escribe el valor leído en la cookie
echo "Esta es la galletita:",$_COOKIE['cookie1'];
?>
```

[ejemplo108.php](#)

Si hemos ejecutado el ejemplo anterior este script ya podrá leer el contenido de la cookie escrita a través de aquel.

```
<?
echo "Esta es la galletita:",$_COOKIE['cookie1'];
?>
```

[ejemplo109.php](#)

## Una cookie con valores asignados mediante una variable

```
<?
$z="Mi regalito";
setcookie("cookie2",$z,time()+3600);
echo "Esta es la galletita:",$_COOKIE['cookie2'];
?>
```

[ejemplo110.php](#)

Este segundo ejemplo **ya** visualizará el valor de esta cookie siempre que hubiéramos ejecutado previamente el script anterior.

```
<?
echo "Esta es la galletita:",$_COOKIE['cookie2'];
?>
```

[ejemplo111.php](#)

Los valores contenidos en las *cookies* pueden ser leídos por el servidor a partir de variables predefinidas o recogidas a través de un formulario.

Tal como puedes ver en el ejemplo **ejemplo112.php**, es posible crear *cookies* en las que la variable contiene un *array* de tipo *escalar* o de tipo *asociativo*.

Como puedes ver en el ejemplo, es necesario incluir un **setcookie** por cada uno de los valores del *array* aunque a la hora de devolver esa variable el navegador del cliente envía el *array* completo.

### ¿Cómo leer cookies?

Al invocar la variable mediante la que fue escrita la *cookie* —desde cualquier página que esté alojada en el subdirectorío del servidor desde el que fué creada— tomará —de forma automática— el valor contenido en la *cookie*.

Esto sólo sería posible en el caso de que la *cookie* hubiera sido creada con anterioridad y que no hubiera expirado.

Tal como comentábamos a estudiar los formularios, las *cookies* pueden leerse directamente (imprimir la variable PHP con nombre idéntico a la *cookie*, sólo en el caso de que la opción **register\_globals=on**.

Si no fuera así (de forma similar al caso de los formularios) para poder leer su contenido tendríamos que recurrir al uso de la variable superglobal **\$\_COOKIE** (caso de que la versión de PHP las soporte) o, alternativamente, de **\$HTTP\_COOKIE\_VARS** teniendo en cuenta que esta última no tiene carácter superglobal.

Puedes comprobar que en los ejemplos se visualizan los contenidos de las *cookies* usando la instrucción **echo** (o **print**) y utilizando el array superglobal **\$\_COOKIE** para garantizar el funcionamiento con cualquier configuración de **register\_globals**.

## Una cookie definida como array

```
<?
$valores=Array("Verde","Verano","Rolls-Royce","Millonario");
# a diferencia de lo que ocurre al definir elementos de array asociativos
# en este caso los índices asociativos (color, estación, etc.) no van
# entre comillas
setcookie("cookie3[color],$valores[0],time()+3600);
setcookie("cookie3[estacion],$valores[1],time()+3600);
setcookie("cookie3[coche],$valores[2],time()+3600);
setcookie("cookie3[finanzas],$valores[3],time()+3600);
# la variable superglobal $_COOKIE['cookie3'] contiene un array, por ello
# la lectura de sus valores debe hacerse considerando que se trata de un
# array bidimensional
if (isset($_COOKIE['cookie3'])) {
    while( list( $indice, $valor) = each($_COOKIE['cookie3']) ) {
        echo "$indice == $valor\n";
    }
}
?>
```

ejemplo112.php

Como en los casos anteriores también este script permite leer los contenidos de la *cookie* después de haber accedido a la página anterior.

```
<?
if (isset( $_COOKIE['cookie3'] ) ) {
    while( list( $indice, $valor) = each($_COOKIE['cookie3']) ) {
        echo "$indice == $valor\n";
    }
}
?>
```

ejemplo113.php

## Un contador como aplicación práctica

```
<?
$numero=$_COOKIE['visitante'];
$numero+=1;
setcookie("visitante",$numero,time()+86400);

if($numero==1){print "Es la $numero vez que visitas esta página";}
if($numero>1){print "Es la $numero ª vez que visitas esta página";}
?>
```

ejemplo114.php

[Anterior](#)

[Índice](#)

[Siguiente](#)