**INTRODUCCIÓN AL MANEJO DE VARIABLES DE SESIÓN**

El soporte para sesiones en PHP consiste en una forma de preservar cierta información a través de accesos subsiguientes. Esto habilita la construcción de aplicaciones más personalizadas e incrementa el atractivo de un sitio web.

El término sesión hace referencia al conjunto de información relativa a un usuario concreto. Esta información puede ser el nombre del propio usuario y el password, o más compleja, como una lista de artículos que ha depositado en la cesta de compra de una tienda online.

Un visitante que accede a su sitio web se le asigna un id único, también llamado id de sesión. Cada usuario distinto de un sitio web tiene su propia información de sesión. Para distinguir una sesión de otra se usan los identificadores de sesión (SID), que se asignan a cada uno de los visitantes de un sitio web y lo identifica.

De esta forma, si el servidor web conoce el SID de un usuario, puede relacionarlo con toda la información que posee sobre él, que se mantiene en la sesión del usuario. Esa información se almacena en el servidor web, generalmente en ficheros aunque también se pueden utilizar otros mecanismos de almacenamiento como bases de datos.

A la información que se almacena en la sesión de un usuario también se le conoce como cookies del lado del servidor (server side cookies). Debemos tener en cuenta que aunque esta información no viaja entre el cliente y el servidor, sí lo hace el SID, bien como parte de la URL o en un encabezado HTTP si se guarda en una cookie. En ambos casos, esto plantea un posible problema de seguridad. El SID puede ser conseguido por otra persona, y a partir del mismo obtener la información de la sesión del usuario.

En PHP para indicar a  que inicie o reanude una sesión usaremos la función **session\_start**.

En PHP para eliminar la información almacenada en la sesión tenemos dos funciones:

* **session\_unset**. Elimina las variables almacenadas en la sesión actual, pero no elimina la información de la sesión del dispositivo de almacenamiento usado.
* **session\_destroy**. Elimina completamente la información de la sesión del dispositivo de almacenamiento.

<?

session\_start();

#pedimos que escriba el identificador único y el nombre de la sesión

echo session\_id(),"<br>";

echo session\_name(),"<br>";

$SID=session\_id();

?>

Mientras la sesión permanece abierta, utilizaremos la variable superglobal **$\_SESSION** para añadir información a la sesión del usuario, o para acceder a la información almacenada en la sesión. Por ejemplo, para contar el número de veces que un usuario visita la página, podemos escribir

<?php

// Iniciamos la sesión o recuperamos la anterior sesión existente

session\_start();

// Comprobamos si la variable ya existe

if (isset($\_SESSION['visitas']))

$\_SESSION['visitas']++;

else

$\_SESSION['visitas'] = 0;

echo "te has conectado ". $\_SESSION['visitas']. " veces";

?>

Queremos que cuando llegue a 10 se destruya la sesión, para ello

<?php

// Iniciamos la sesión o recuperamos la anterior sesión existente

session\_start();

// Comprobamos si la variable ya existe

if (isset($\_SESSION['visitas']))

$\_SESSION['visitas']++;

else

$\_SESSION['visitas'] = 0;

echo "te has conectado ". $\_SESSION['visitas']. " veces";

// eliminar la sesion cuando se ha visitado la página mas de 10 veces

if ($\_SESSION['visitas']>10)

unset($\_SESSION['visitas']);

?>

Si en lugar del número de visitas (cuenta), queremos almacenar el instante en que se produce cada una, la variable de sesión "horavisitas" deberá ser un array para almacenar la colección de valores, el código ahora será:

<?php

// Iniciamos la sesión o recuperamos la anterior sesión existente

session\_start();

// Cada vez que recargamos la página añadimos un valor al array "horavisitas"

$\_SESSION['horavisitas'][] = time();

print\_r($\_SESSION['horavisitas']);

foreach($\_SESSION['horavisitas'] as $v)

echo date("d/m/y \a \l\a\s H:i", $v) . "<br />"; // dar formato

?>