SESIONES

Las sesiones sirven para almacenar datos que necesitan ser recordados en otras páginas de un sitio web sin necesidad de volver a solicitarlos.



Una definición simple de sesión sería el tiempo que un usuario se mantiene conectado a una aplicación web. Otra definición: una sesión es la secuencia de peticiones a un servidor web realizadas por un mismo cliente desde su navegador dentro del ámbito de una aplicación web.

Denominamos aplicación web a aquella que es algo más que una simple consulta de una página. Aquella en la que es importante autenticar al usuario, la que precisa conocer los pasos de la navegación y las preferencias del usuario, la que necesita acumular una cesta de la compra, la que debe actualizar una base de datos, etc.

La diferencia con respecto a [las cookies](http://www.mclibre.org/consultar/php/lecciones/php_cookies.html) es que mediante cookies la información se guarda en el ordenador del usuario y se envía en cada petición al servidor, mientras que mediante sesiones la información se guarda en el servidor.

Las sesiones permiten que páginas distintas puedan acceder a una variable común, la matriz $\_SESSION.

El trabajo con sesiones tiene tres partes:

* Creación o apertura de la sesión

Cuando alguna página crea una sesión utilizando la función correspondiente, el servidor asocia al navegador del usuario un identificador de usuario único. El identificador se guarda en el usuario en forma de cookie o, si no se permite la creación de cookie, añadiendo el identificador en la dirección de la página. Todo esto PHP lo hace automáticamente.

* Utilización de la sesión

Una vez creada la sesión, las páginas solicitadas por el mismo navegador pueden guardar y recuperar información en el servidor, información que se asocia al identificador de usuario, por lo que no es accesible a otros. La información se conserva hasta que el usuario o el servidor destruyan la sesión. Todas las páginas que necesitan utilizar la información almacenada en la sesión deberán ejecutar la función: session\_start().

* Destrucción o cierre de la sesión

Tanto el usuario como el servidor pueden cerrar la sesión. El usuario puede destruir la sesión cerrando el navegador. El servidor puede destruir la sesión cuando alguna página utilice la función correspondiente.

**Crear la sesión**

En PHP, las sesiones se crean mediante la función [session\_start()](http://www.php.net/manual/es/function.session-start.php). Si la sesión no existía, esta función crea la sesión y le asocia un identificador de sesión único. Si la sesión ya existía, esta función permite que la página tenga acceso a la información vinculada a la sesión.

Por ejemplo:

**<?php**

**session\_start();**

**?>**

Hay que tener la precaución de utilizar la función session\_start() antes de empezar a escribir el contenido de la página, porque si no PHP producirá un aviso y no se creará la sesión. El motivo es que el identificador de sesión se utiliza en las cabeceras de respuesta HTTP y las cabeceras se envían antes del texto de la página. Es decir, cuando PHP encuentra una instrucción que escribe texto, cierra automáticamente la cabecera; si a continuación PHP encuentra en el programa la función session\_start(), da un aviso porque ya se han enviado las cabeceras y no se crea la sesión.

**Utilizar la sesión**

Cuando una página ha creado una sesión o ha accedido a una sesión ya existente mediante session\_start(), la página tiene acceso a la matriz [$\_SESSION](http://www.php.net/manual/es/reserved.variables.session.php) que contiene las variables de esa sesión.

La matriz $\_SESSION es una matriz asociativa en la que se pueden definir valores como en cualquier otra matriz. La diferencia es que $\_SESSION es accesible desde páginas diferentes (siempre que esas páginas tengan asociada la misma sesión), manteniéndose los valores de una página a otra.

Un detalle importante que distingue a las sesiones de las cookies es que:

* en el caso de las cookies, cuando una página pide al navegador que cree una cookie, el valor de la cookie no está disponible en $\_COOKIE en esa página. Lo estará en páginas posteriores, cuando el navegador pida otra página y envíe el valor en la petición.
* en el caso de las sesiones, cuando una página crea un valor en $\_SESSION, ese valor está disponible en esa misma página.

La información se conserva hasta que el usuario o el servidor destruyan la sesión.

**Destruir la sesión**

El usuario puede destruir la sesión cerrando el navegador o también se destruye la sesión debido al tiempo transcurrido desde su creación y también se pueden destruir por programa mediante la función [session\_destroy()](http://www.php.net/manual/es/function.session-destroy.php)

Cuando se destruye una sesión, el programa que ha destruido la sesión sigue teniendo acceso a los valores de $\_SESSION creados antes de la destrucción de la sesión, pero las páginas siguientes no.

Comprobar en php.ini la directiva: session.auto\_start

Ésta directiva determina si el módulo que administra las sesiones se inicia automáticamente al comenzar la petición. Si está inicializada a 0 (predeterminado), en nuestras páginas .php activaremos el uso de sesiones con la función: session\_start().

Si le asignamos el valor 1, no utilizaremos la función session\_start().

Función: session\_id([id]);

Retorna el identificador de la sesión vigente si no recibe nada, si recibe un identificador estaremos estableciendo dicho identificador como identificador de la sesión vigente.

Función: session\_name([nombre]);

Retorna el nombre de la sesión vigente si no recibe nada, si recibe un nombre estaremos estableciendo dicho nombre como el nombre de la sesión vigente. Por defecto el nombre es: PHPSESSID

Los datos de la sesión se guardan en la carpeta indicada en la opción session.save\_path del archivo php.ini, se crea un archivo por sesión.

Ejemplo

<?php

**session\_start**();

$fechahora = "Este es el primer acceso";

$contador = 0;

**if** (**isset**($\_SESSION['fechahora'])){

$fechahora = $\_SESSION['fechahora'];

$contador = $\_SESSION['contador'];

}

date\_default\_timezone\_set('Europe/Paris');

$\_SESSION["fechahora"] = **date**('d/m/Y h:i:s'); // actualiza la variable

$contador++;

$\_SESSION["contador"] = $contador; // actualiza la variable

?>

<HTML>

<HEAD>

<TITLE>**Uso de sesiones en PHP** </TITLE>

</HEAD>

<BODY>

<H3>**Fecha y hora actual:** <?php **echo** **date**('d/m/Y h:i:s') ?></H3><BR><B>

**Contenido de la superglobal $\_SESSION**</B><BR>

<?php

**echo** "Identificador de la sesi󮺠" . session\_id() . "<BR>";

**echo** "Nombre de la sesi󮺠" . session\_name() . "<BR>";

**echo** "Elemento fechahora: " . $\_SESSION['fechahora'] . "<BR>" ;

**echo** "Elemento contador: " . $\_SESSION['contador'] . "<BR><BR>" ;

**echo** "<BR>La ltima vez que accedi󠡠esta p᧩na fue : **$fechahora** <BR>";

**echo** "<BR>Cantidad de accesos a esta p᧩na : **$contador** <BR>";

?>

</BODY>

</HTML>

El borrado de una variable de sesión se efectúa mediante la función unset(). El borrado solamente afecta a un elemento de la matriz asociativa $\_SESSION, la sesión sigue manteniendo su identificador y su nombre.

Ejercicio

En el ejemplo anterior, eliminar el elemento contador de la matriz $\_SESSION y comprobar que queda vacío en la visualización

Ejemplo:

 Página 1 – En esta página ira el formulario de inicio de sesión

 Página 2 – Esta página será la página con contenido para solo usuarios logeados

 Página 3 – Esta página será para cerrar la sesión

<?php

**session\_start**();

?>

<!DOCTYPE html>

<html>

<head>

<meta charset=**"UTF-8"**/>

<title>**Pagina 1**</title>

</head>

<body>

<?php

**if**(**isset**($\_SESSION['nombre'])){

**echo** "<p>Has iniciado sesion: " . $\_SESSION['nombre'] . "";

**echo** "<p><a href='sesion3.php'>Cerrar Sesion</a></p>";

}**else** {

?>

<form action=**"sesion2.php"** method=**"POST"**>

<p>**Nombre:**<input type=**"text"** placeholder=**"Tu**

**Nombre!"** name=**"nombre"** required/></p><br />

<input type=**"submit"** value=**"Enviar"** />

</form>

<?php

}

?>

<a href=**"sesion2.php"**>**Ir a pagina 2**</a>

</body>

</html>

<?php

**session\_start**();

?>

<!DOCTYPE html>

<html>

<head>

<meta charset=**"UTF-8"**/>

<title>**Pagina 2**</title>

</head>

<body>

<?php

**if**(**isset**($\_POST['nombre'])){

$\_SESSION['nombre'] = $\_POST['nombre'];

**echo** "Bienvenido! Has iniciado sesion: ".$\_POST['nombre'];

}**else**{

**if**(**isset**($\_SESSION['nombre'])){

**echo** "Has iniciado Sesion: ".$\_SESSION['nombre'];

}**else**{

**echo** "Acceso Restringido";

}

}

?>

<br /><a href=**"sesion1.php"**>**Ir a pagina 1**</a>

</body>

</html>

Ejercicio:

Confeccionar un formulario que solicite teclear el mail de un alumno. Si el mail existe en la tabla alumnos, rescatar su nombre y almacenarlo en una variable de sesión. Además disponer un hipervínculo a una tercera página que verifique si existe la variable de sesión y dé la bienvenida al alumno, en caso contrario mostrar un mensaje indicando que no puede visitar esta página.