Cookies y Sessions en el lenguaje PHP

Cookies

- Es un **fichero de texto** que un sitio web guarda en el entorno del **usuario del navegador**.
- Su uso más típico es el almacenamiento de las **preferencias del usuario** (por ejemplo, el idioma en que se deben mostrar las páginas), para que no tenga que volver a indicarlas la próxima vez que visite el sitio.
- Con las herramientas de desarrollador podemos ver las cookies que hay guardadas en nuestro equipo para una determinada página.
- Por ejemplo, en Chrome accedemos a las cookies a través de la pestaña *Application* de las herramientas de desarrollador.

Cookies en PHP

- En PHP, para almacenar una cookie en el navegador del usuario, usamos la función **setcookie**.
- El único **parámetro obligatorio** que tienes que usar es el **nombre** de la cookie, pero admite varios parámetros más opcionales.
  - http://es.php.net/manual/es/function.setcookie.php
- Por ejemplo, si queremos almacenar en una cookie el color seleccionado por el usuario en un formulario de preferencias, podemos hacer algo así:
  - setcookie("color", $_POST["color"], time()+3600);
- Los dos primeros parámetros son el nombre de la cookie y su valor.
- El tercero es la fecha de caducidad de la misma (una hora desde el momento en que se ejecute). Si no se indica, la cookie se eliminará cuando se cierre el navegador.

Dónde cargar las cookies

- Las cookies se transmiten entre el navegador y el servidor web utilizando los encabezados del protocolo HTTP.
- Las sentencias **setcookie** deben enviarse antes de que el navegador muestre información alguna en pantalla.

Acceder a las cookies en PHP

- Cuando accedes a un sitio web, **el navegador le envía** de forma automática todo el contenido de **las cookies** que almacene relativas a ese sitio en concreto.
- Desde PHP puedes acceder a esta información por medio de la variable super global **$_COOKIE**.

Disponibilidad de las cookies

- En última instancia la disponibilidad de las cookies está **controlada por el cliente**.
- Algunos usuarios deshabilitan las cookies en el navegador porque piensan que la información que almacenan puede suponer un potencial **problema de seguridad**.
- También pueden **eliminar las cookies** por temas de mantenimiento (formatear el equipo, limpiar cookies, etc.).

Eliminar cookies del cliente

- Podemos eliminar una cookie almacenada en el navegador antes de que expire utilizando la misma **función setcookie** pero indicando una **fecha de caducidad anterior a la actual**.

Sesiones

- Existen diversos **problemas** asociados a las cookies:
  - Número de cookies que admite el navegador.
  - Tamaño máximo, etc.
- Para solventar estos inconvenientes, existen las sesiones.
- El término sesión hace referencia al **conjunto de información relativa a un usuario concreto**.
- Esta información puede ser tan simple como el **nombre del propio usuario**, o más compleja, como los artículos que ha depositado en la **cesta de compra** de una tienda online.

Identificadores de sesión SID

- 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).
- Un SID es un **atributo** que se asigna a cada uno de los **visitantes de un sitio web** y lo identifica.
- El servidor web utiliza el SID de un usuario para relacionarlo con la información que posee sobre él, que se mantiene en la sesión del usuario.
- Esa información **se almacena en el servidor web**.

Propagar el SID

- Existen dos maneras de mantener el SID entre las páginas de un sitio web que visita el usuario:
  - Utilizando cookies
  - Propagando el SID en un parámetro de la URL:
    - http://www.misitioweb.com/tienda/listado.php?PHPSESSID=34534fg4ffg34ty
- El **mejor método y más utilizado es utilizar una cookie** para almacenar el SID.
- Propagar el SID como parte de la URL conlleva mayores desventajas, como la imposibilidad de mantener el SID entre distintas sesiones, o el hecho de que compartir la URL con otra persona implica compartir también el identificador de sesión.

Server side cookies

- El proceso de manejo de sesiones en **PHP está automatizado**.
- No es necesario guardar una cookie con el SID.
- Esto lo hace PHP automáticamente.
- 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).

Seguridad en server side cookies

- Las server side cookies no viajan entre el cliente y el servidor, pero sí lo hace el SID.
- 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.
- La manera más segura de utilizar sesiones es **almacenando los SID en cookies y utilizar HTTPS para encriptar la información** que se transmite entre el servidor web y el cliente.

Configuración

- Por defecto, PHP tiene soporte de sesiones incorporado.
- En el fichero **php.ini** existen una serie de directivas de configuración de las sesiones.
- La función **phpinfo** ofrece información sobre la configuración actual de las directivas de sesión.
- En la documentación de PHP podemos encontrar información sobre las directivas que permiten configurar el manejo de sesiones.
  - http://es.php.net/manual/es/session.configuration.php

Directivas de configuración

Directiva	Significado
session.use_cookies	Indica si se deben usar cookies (1) o propagación en la URL (0) para almacenar el SID.
session.use_only_cookies	Se debe activar (1) cuando utilizas cookies para almacenar los SID, y además no quieres que se reconozcan los SID que se puedan pasar como parte de la URL (este método se puede usar para usurpar el identificador de otro usuario).
session.save_handler	Se utiliza para indicar a PHP cómo debe almacenar los datos de la sesión del usuario. Existen cuatro opciones: en ficheros (`files`), en memoria (`mm`), en una base de datos SQLite (`sqlite`) o utilizando para ello funciones que debe definir el programador (`user`). El valor por defecto (`files`) funcionará sin problemas en la mayoría de los casos.
session.name	Determina el nombre de la cookie que se utilizará para guardar el SID. Su valor por defecto es `PHPSESSID`.
session.auto_start	Su valor por defecto es 0, y en este caso deberás usar la función `session_start` para gestionar el inicio de las sesiones. Si usas sesiones en el sitio web, puede ser buena idea cambiar su valor a 1 para que PHP active de forma automática el manejo de sesiones.
session.cookie_lifetime	Si utilizas la URL para propagar el SID, éste se perderá cuando cierres tu navegador. Sin embargo, si utilizas cookies, el SID se mantendrá mientras no se destruya la cookie. En su valor por defecto (0), las cookies se destruyen cuando se cierra el navegador. Si quieres que se mantenga el SID durante más tiempo, debes indicar en esta directiva ese tiempo en segundos.
session.gc_maxlifetime	Indica el tiempo en segundos que se debe mantener activa la sesión, aunque no haya ninguna actividad por parte del usuario. Su valor por defecto es 1440. Es decir, pasados 24 minutos desde la última actividad por parte del usuario, se cierra su sesión automáticamente.

Inicio de sesión automático

El inicio de una sesión puede tener lugar de dos formas:

- Si está activa la directiva **session.auto_start** en la configuración de PHP, **la sesión comenzará automáticamente** en cuanto un usuario se conecte a tu sitio web.
- Si la sesión ya se abrió anteriormente se reanudará la anterior utilizando el SID anterior, que estará almacenado en una cookie (si usamos propagación del SID, no podremos restaurar sesiones anteriores; el SID figura en la URL y se pierde cuando se cierra el navegador).

Inicio de sesión manual

- Utilizaremos la función **session_start** para indicar a PHP que inicie una nueva sesión o reanude la anterior.
- Devuelve false en caso de no poder iniciar o restaurar la sesión.
- El inicio de sesión requiere utilizar cookies, por lo tanto habrá que hacer **la llamada antes de que la página web muestre información** en el navegador.
- **Todas las páginas** que necesiten utilizar la información almacenada en la sesión, deberán ejecutar la función **session_start**.

Acceder a los datos de la sesión

- Una vez abierta la sesión, usamos 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 el usuario visita la página, podemos hacer:

```
// 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;
```

Eliminar una variable de la sesión

- Para eliminar una variable de la sesión usaremos la función **unset()**

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

Destruir la sesión

- Para cerrar la sesión de forma manual utilizaremos la función **session_destroy**
- Antes debemos eliminar las variables de la sesión de la siguiente forma:

```
$_SESSION = [];
```