

Gestión de sesiones con PHP

I. SESIONES EN PHP

El soporte de sesiones en PHP es un mecanismo de programación, habilitado por defecto, que permite conservar cierta información sobre un usuario cuando éste pasa de una página web a otra dentro del sitio web.

Cuando se crea una sesión, con independencia de que se guarden o no datos de la misma, la información asociada se almacena en el servidor, en un archivo que se crea para la sesión y que se ubica en el directorio especificado por la opción `session.save_path` dentro del archivo `php.ini`.

Para almacenar datos asociados con una sesión se puede utilizar la variable global `$_SESSION`, que consiste en un array asociativo donde se guardan las variables de sesión disponibles para el script actual.

La creación de una variable de sesión se hace estableciendo un miembro apropiado en esta matriz. Por ejemplo:

```
$_SESSION['nombre_variable'];
```

Cuando se usa `$_SESSION`, se puede utilizar `unset()` para dejar de registrar una variable de sesión. Por ejemplo:

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

Con la función `unset()` hay que tener cuidado y no ejecutarla sobre `$_SESSION`, ya que si se hace `unset($_SESSION)` se deshabilitará el registro de variables de sesión a través del array superglobal `$_SESSION`.

Algunas de las funciones existentes para el manejo de sesiones son:

- `session_start()` : crea una sesión o reanuda la actual basada en un identificador de sesión pasado mediante una petición GET o POST, o pasado mediante una cookie.
- `session_id()` : se utiliza para obtener o establecer el id de sesión para la sesión actual; si se especifica un argumento como identificador, reemplazará el identificador de sesión actual si se llama antes de `session_start()`; el identificador debe estar formado por caracteres en el rango a-z A-Z 0-9 , (coma) y - (menos), y debe contener al menos una letra; cuando se usan cookies de sesión, al especificar un identificador para `session_id()` se enviará siempre una nueva cookie cuando se llame a `session_start()`, sin importar si el identificador de sesión actual es idéntico al que se ha especificado. En PHP el identificador de una sesión se guarda en una cookie denominada PHPSESSID.
- `session_name()` : obtiene y/o establece el nombre de la sesión actual usado cookies y URLs; si se le da un argumento, que debería contener sólo caracteres alfanuméricos, actualizará con él el nombre de la sesión y devolverá el nombre antiguo de la sesión; el nombre de la sesión se reinicia al valor predeterminado almacenado en `session.name` en el momento de iniciar una petición, por tanto, se necesita llamar a `session_name()` para cada petición (y antes de llamar a `session_start()`).
- `session_destroy()` : destruye toda la información asociada con la sesión actual, pero no destruye ninguna de las variables globales asociadas con la sesión, ni destruye la cookie de sesión si la hubiera; para volver a utilizar las variables de sesión se debe llamar a `session_start()`; para destruir una sesión completamente, como por ejemplo al desconectar un usuario, debe destruirse el id de sesión; si se usa una cookie para propagar el id de sesión (comportamiento por defecto), entonces también debe borrarse la cookie de sesión (se puede utilizar `setcookie()` para eso).

- `session_status()` : se usa para devolver el estado de la sesión actual; los valores devueltos son: `PHP_SESSION_DISABLED` si las sesiones están deshabilitadas, `PHP_SESSION_NONE` si las sesiones están habilitadas, pero no existe ninguna y `PHP_SESSION_ACTIVE` si las sesiones están habilitadas, y existe una.
- `session_set_cookie_params()` : devuelve una matriz con la información de la cookie de sesión actual; la matriz contiene los siguientes elementos: `lifetime` (tiempo de vida de la cookie en segundos), `path` (ruta donde se almacena la información), `domain` (dominio de la cookie), `secure` (la cookie debería ser enviada solamente sobre conexiones seguras) y `httponly` (la cookie sólo puede ser accedida a través del protocolo HTTP).
- `session_save_path()` : devuelve la ruta del directorio actual usado para guardar la información de sesión; si se le da como argumento una ruta y la llamada se hace antes de `session_start()`, cambiará el directorio donde se guarda la información.
- `session_write_close()` : finaliza la sesión actual y almacena la información de la sesión; normalmente la información de la sesión se almacena después de que el script termine sin necesidad de llamar a `session_write_close()`, pero como la información de la sesión está bloqueada para prevenir escrituras concurrentes, el script sólo puede operar sobre una sesión cada vez (esto es particularmente importante cuando se usan sub-ventanas (frames) junto con sesiones, pues se observará que las sub-ventanas se cargan de una en una debido a ese bloqueo).

Otras funciones útiles para trabajar con sesiones son las que permiten manejar variables. Entre ellas se encuentra la función `isset()`, que permite comprobar si una variable ha sido inicializada.

Ejemplo:

En el caso siguiente se utilizan tres páginas web cuyos nombres de archivos se indican delante del código correspondiente. La primera página web sirve para que un usuario se identifique en una sesión y pueda tener acceso a partes reservadas en la segunda página y en la primera; además, enlaza con la tercera página donde se cierra la sesión del usuario. Para ejecutar este caso los archivos deben estar ubicados en el directorio `htdocs`, debe estar funcionando el servidor Apache y en el navegador debe escribirse:

`http://localhost/sesionPagina_1.php`

Archivo: `sesionPagina_1.php`

```
<?php
// Se abre una sesión
session_start();
?>
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8"/>
  <title>Sesiones: página 1</title>
</head>
<body>
  <?php
  if(isset($_SESSION['usuario'])) {
    // Si estando el usuario identificado accede
    // a la página por segunda vez o más,
    // entonces se muestra un mensaje avisando de que la sesión está activa
    echo "<p>Ha vuelto a la página 1. Su sesión está activa y la inició como: " .
$_SESSION['usuario'] . " ";
    echo "<p><a href='sesionPagina_3.php'>Cerrar sesión</a></p>";
  } else {
```

```
?>
    <! Si el usuario accede a la página sin estar identificado, se muestra un
formulario>
    <form action="sesionPagina_2.php" method="POST">
        <p>Nombre de usuario: <input type="text" placeholder="Escriba su
nombre" name="usuario" required/></p>
        <input type="submit" value="Pulse aquí para iniciar la sesión" />
    </form>
<?php
}
?>
<p><a href="sesionPagina_2.php">Ir a la página 2</a></p>
</body>
</html>
```

Archivo: sesionPagina_2.php

```
<?php
// Recupera la sesión abierta
session_start();
?>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8"/>
    <title>Sesiones: página 2</title>
</head>
<body>
    <?php
    if(isset($_POST['usuario'])) {
        // Si el usuario accede a la página inmediatamente
        // después de identificarse se muestra un mensaje inicial
        $_SESSION['usuario'] = $_POST['usuario'];
        echo "Está en la página 2 y ha iniciado
            la sesión como: " . $_POST['usuario'];

    } else {
        if(isset($_SESSION['usuario'])) {
            // Si estando el usuario identificado accede a la página por segunda vez o más,
            // entonces se muestra un mensaje diferente y la información de la sesión
            echo "Ha vuelto a la página 2. Su sesión sigue activa y la inició
como: " . $_SESSION['usuario'];
            echo "<p>Otra información de su sesión es la siguiente:";
            echo "<p>session_id: " . session_id();
            echo "<p>session_name: " . session_name();
            echo "<p>";
            print_r(session_get_cookie_params());

        } else {
            // Si el usuario accede a la página sin identificarse se
muestra un mensaje de aviso
            echo "Aviso: Acceso restringido. Debe iniciar sesión dando su
nombre.";
        }
    }
?>
    <p><a href="sesionPagina_1.php">Ir a la página 1</a></p>
</body>
</html>
```

Archivo: sesionPagina_3.php

```
<?php
// Se recupera la sesión abierta y se cierra
session_start();
session_destroy();
?>
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8"/>
  <title>Sesiones: página 3</title>
</head>
<body>
  <p>Está en la página 3 y ha cerrado la sesión.</p>
  <p><a href="sesionPagina_1.php">Ir a la página 1</a></p>
</body>
</html>
```