

## 23. Sesiones y Cookies

### 23.1 Sesiones.

Las sesiones son una forma sencilla de almacenar datos para usuarios de manera individual usando un ID de sesión único. Esto se puede usar para hacer persistente la información de estado entre peticiones de páginas. Los ID de sesiones normalmente son enviados al navegador mediante cookies de sesión, y el ID se usa para recuperar los datos de sesión existente. La ausencia de un ID o una cookie de sesión permite saber a PHP para crear una nueva sesión y generar un nuevo ID de sesión.

Las sesiones siguen un flujo de trabajo sencillo. Cuando una sesión se inicia, PHP recuperará una sesión existente usando el ID pasado (normalmente desde una cookie de sesión) o, si no se pasa una sesión, se creará una sesión nueva. PHP rellenará la variable superglobal `$_SESSION` con cualesquiera datos de la sesión iniciada. Cuando PHP se cierra, automáticamente toma el contenido de la variable superglobal `$_SESSION`, la serializa, y la envía para almacenarla usando el gestor de almacenamiento de sesiones.

Las sesiones se puede iniciar manualmente usando la función **`session_start()`**.

Para dar por cerrada una sesión se utiliza **`session_destroy()`**.

**Header.php:**

```
<!DOCTYPE HTML>
<html lang="es">
<style>
  ul.lista li {
    display: inline;
  }
</style>
<head>
<meta charset="UTF-8" />
<title>Ejemplo sesi n</title>
</head>

<body>

  <div class="cabecera">
    <h1>Ejemplo de sesi n</h1>

    <ul class="lista">
      <li><a href="index.php">Login</a><li>
      <li><a href="infosesion.php">Informaci n</a><li>
      <li><a href="logout.php">Logout</a><li>

    </ul>
    <hr/>

  </div>
```

*Index.php:*

```
<?php
include ('header.php');
?>

<div>

    <h2>Login</h2>

    <?php
    session_start();
    if (! empty($_SESSION['login'])) {
        echo '<p>Se ha iniciado sesión con el usuario ' . $_SESSION['login'] . ' ';
        echo 'Introduzca otro usuario si desea cambiar<p>';
    }

    ?>
    <form action="procesalogin.php" method="POST">
        Nombre
        <input type="text" name="login"> <input type="submit" value="Login">

    </form>

</div>

</body>

</html>
```

*procesalogin.php:*

```
<?php

session_start();

echo $_POST['login'];
if (!empty($_POST['login'])) {
    $_SESSION['login'] = $_POST['login'];
    header('Location: index.php');
} else {
    echo "Error en el login";
}
?>
```

*logout.php:*

```
<?php
include('header.php');

    session_start();
    session_destroy();

?>

    <div>
        <h2>Se ha cerrado la sesión</h2>
    </div>

</body>
</html>
```

## 23.2 Cookies.

Las cookies son como archivos de texto que se guardan en el ordenador. A petición de un servidor web, el navegador crea un archivo de este tipo. Una vez que ocurre esto el servidor puede leer y escribir contenido en este archivo. Puede parecer peligroso, pero existen una serie de restricciones para hacerlo lo más seguro posible:

Los servidores web sólo pueden acceder a cookies establecidas a su propio dominio. Este dominio es establecido por el navegador cuando el servidor crea una nueva cookie, y sólo puede ser un dominio o subdominio del servidor.

- Según el protocolo HTTP, las cookies no pueden ser más grandes de 4096 Bytes (4KB).
- Hay un límite de cookies por dominio. Depende del navegador, pero suelen ser 20 cookies.
- También hay un límite en el número total de cookies en el disco duro del cliente. Suele ser de unas 300 cookies. Cuando se llega a este número, una cookie antigua se elimina antes de crear la nueva.
- Las cookies tienen una fecha de expiración. Esta fecha permite al navegador eliminar cookies antiguas cuando ya no son requeridas por el servidor web.
- Si la fecha de expiración está vacía, la cookie se eliminará cuando finalice la conexión con el servidor. Esto ocurrirá cuando el usuario cierre la ventana o pestaña del sitio web, o directamente cierre el navegador. Estas cookies, normalmente denominadas session cookies, son usadas principalmente para guardar ajustes temporales.

**Poncookie.php:**

```
<!DOCTYPE HTML>
<html lang="es">

<head>
<meta charset="UTF-8" />
<title>Ejemplo cookie</title>
</head>

<body>
<?php

setCookie('moneda','EUR');

?>

</body>
```

**Vecookie.php:**

```
<!DOCTYPE HTML>
<html lang="es">

<head>
<meta charset="UTF-8" />
<title>Ejemplo cookie</title>
</head>

<body>
<?php

if (isset($_COOKIE['moneda'])) {
    echo 'La moneda elegida es ' . $_COOKIE['moneda'];
} else {
    echo 'No existe moneda elegida';
}

?>

</body>
```

**Borracookie.php:**

```
<!DOCTYPE HTML>
<html lang="es">

<head>
<meta charset="UTF-8" />
<title>Ejemplo cookie</title>
</head>

<body>
<?php

setCookie('moneda','EUR', time()-1);

?>

</body>
```

### ***Ejercicio 23.1***

Mejora el ejemplo de sesiones para guardar en la sesión más información, añadiendo los elementos que consideres en el formulario.

### ***Ejercicio 23.2***

Transforma el ejercicio 23.1 para que funcione con Cookies en vez de con sesión.

### ***Ejercicio 23.3***

Transforma el ejercicio 5.6 para que utilice la sesión y sea más fácil pasar la matriz.