

UD4. Pt1 – Cookies y Sesiones (1)

En esta práctica comenzareis a trabajar con cookies. Lo más importante que hay que saber es que las cookies son ficheros que guardan información mientras navegamos entre diferentes páginas. A diferencia de las sessions, las cookies se almacenan en el navegador, no en el servidor.

Podéis consultar estos enlaces para ver cómo se trabaja con cookies en PHP:

- https://www.w3schools.com/php/php_cookies.asp
- <http://php.net/manual/en/features.cookies.php>

Como veis en los tutoriales, hay 4 acciones básicas que hay que saber hacer:

- Crear la cookie (tiene un tiempo de expiración).
- Leer el valor de la cookie que deseemos.
- Modificar (actualizar) el valor de la cookie.
- Eliminar la cookie.

También es conveniente tener en cuenta los siguientes aspectos:

- Para crear y eliminar las cookies, se recomienda crear nuestras propias funciones a partir de las funciones primitivas de PHP, testearlas, y hacer los includes necesarios para poderlas utilizar en nuestro código:
 - ✓ function **crearCookie**(\$nombreCookie, \$valorCookie) {...}

Esta función la utilizaremos para:

- Crear una nueva cookie.
- Sobrecribir una cookie que ya existe.
- ✓ function **eliminarCookie**(\$nombreCookie) {...}
- Cuando se trabaja con cookies, es muy útil visualizarlas en el navegador.
 - ✓ En Firefox, disponemos del atajo de teclado **Ctrl+Shift+K** con el que se abre la consola del navegador y tenemos acceso a las herramientas de desarrollador.
 - ✓ En la sección «Almacenamiento», podemos visualizar las cookies del sitio web donde nos encontramos.

Podemos comprobar qué cookies tenemos creadas, y visualizar su valor. Podemos incluso eliminarlas manualmente.

En una cookie es posible guardar arrays o objetos de la siguiente manera:

- ✓ la función `serialize()` nos permite obtener una representación almacenable como string
- ✓ después, para recuperar el valor de la cookie habrá que hacer `unserialize()`

Se pide implementar una aplicación de servidor con una estructura tipo Modelo-Vista-Controlador (MVC):

- Raíz del proyecto con el fichero:

index.php

- Carpeta controller con los ficheros:

controller.php

crearCookie.php

eliminarCookie.php

reiniciar.php

- Carpeta view con los ficheros:

form.php

introducida.php

list_frases.php

header.php

footer.php

En el header abriremos los tags de HTML, mientras que en el footer mostraremos un mensaje -pie de página- y cerraremos los tags abiertos en el header.

La idea es incluir el header y el footer en todas las vistas de la aplicación, mediante la sentencia **include** de PHP (o con `require_once`).

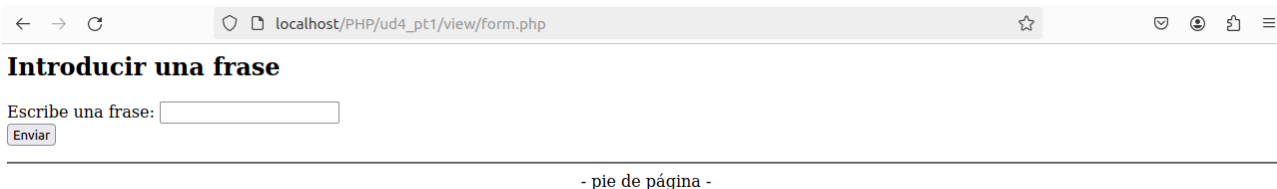
El funcionamiento de la aplicación será el siguiente:

- 1) Página de inicio. Tendrá links a las funcionalidades siguientes:

- ⑩ «introducir una frase».
- ⑩ «ver todas las frases».

- 2) Introducir una frase:

- ⑩ Al clicar este link, se abrirá una página con un formulario como este:



← → ↻ localhost/PHP/ud4_pt1/view/form.php ☆ 📄 📄 📄

Introducir una frase

Escribe una frase:

Enviar

- pie de página -

- ⑩ **Ayuda:** podemos crear un array vacío, al cual iremos añadiendo elementos (frases) con la instrucción `array_push()`

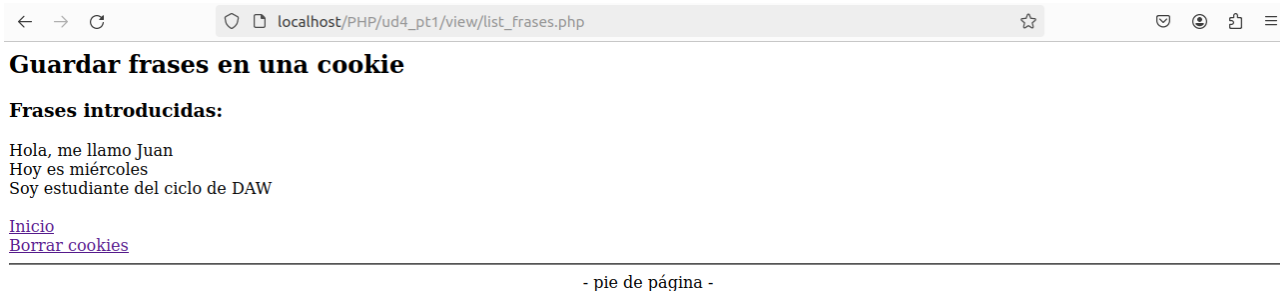
Atención: no podemos almacenar directamente el array en una cookie. Tenemos que hacer uso de estas instrucciones de PHP:

- ⑩ **serialize()** para convertir el array a su representación en formato cadena de texto antes de guardarlo en la cookie.
- ⑩ **unserialize()** para llevar a cabo el proceso inverso: una vez leída la cookie, tenemos que pasar del formato cadena de texto al formato array original.

Una vez cliquemos a enviar, veremos una pantalla con un mensaje de confirmación: «Frase introducida correctamente !»

- 3) Ver todas las frases:

- ⑩ Al clicar este link, se abrirá una página donde se visualizarán todas las frases introducidas hasta ahora, separadas por un salto de línea.



Si clicamos la opción «borrar cookies», se borrarán todas las frases introducidas y la aplicación nos redirigirá al inicio.

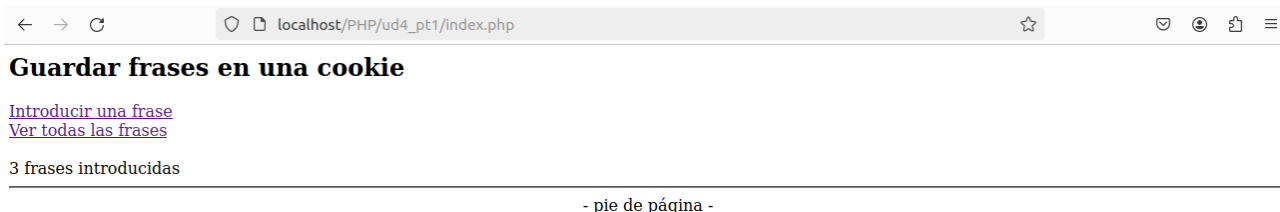
El link de borrar cookies sólo tiene que verse si existe alguna frase.

4) var_dump del array de frases.

- ⑩ Ejecuta en controller.php la instrucción **var_dump()** para comprobar en pantalla cómo efectivamente hemos añadido la frase a un array de frases.
- ⑩ Tendremos que añadir después del `var_dump()` una instrucción **exit** de PHP, que detiene la ejecución del programa. Si no lo hacemos, se efectuará la redirección a la pantalla de confirmación y no veremos el `var_dump()`
- ⑩ Lo mejor es que las dos instrucciones estén en la misma línea, así podemos comentar el código añadido fácilmente una vez hayamos terminado de hacer las pruebas.

```
// var_dump($array); exit;
```

5) Añade un mensaje en la página de inicio que informe de la cantidad de frases introducidas.



Posibles soluciones:

- ⑩ utilizar una cookie contador.
- ⑩ contar la cantidad de elementos del array (usando una función de la librería «Array Functions» de PHP) cada vez que carguemos la página de inicio.

Si hemos borrado las cookies, se tiene que mostrar el mensaje:

0 frases introducidas

(hay que evitar errores de PHP por intentar acceder a una cookie que no existe)

Entrega de la práctica: carpeta comprimida con los ficheros de código fuente.