

UT4 - Ejercicios

Cookies

y

Sesiones

1. Codificar un script PHP que genere una **cookie** llamada **idioma** con valor **'es'**. La página mostrará por defecto el mensaje en inglés *'There was upon a time ...'* La siguiente(s) veces que se ejecute el script, se recuperará dicha cookie y aparecerá el mensaje *'Érase una vez ...'*.
2. Codificar un script PHP tal que al conectarse un usuario por primera vez, le envíe una **cookie** llamada **identificador**, de contenido una cadena alfanumérica aleatoria de longitud fija y duración 1 hora. Así mismo, se informará al usuario que era *"nuevo en el lugar"*.

Al conectarse en sucesivas ocasiones, se recuperará la cookie y se informará al usuario de que *"ya estaba registrado"*.

Nota: Utilizar esta función para obtener la cadena aleatoria

```
function generateRandomString($length = 10) {  
    return  
    substr(str_shuffle("0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"), 0, $length);  
}
```

Para investigar:

Utilizar las *herramientas del desarrollador* de Google Chrome (**Inspeccionar Elemento**):

- **Network:** para comprobar qué se envía en las cabeceras HTTP (*Set-Cookie*, de la respuesta (*Response*) y *Cookie* de la solicitud (*Request*)).
- **Application/Storage/Cookies:** para consultar, eliminar, modificar el valor de la cookie.

Para gestionar las cookies desde Google Chrome también se puede instalar el complemento *EditThisCookie*.

3. Crear un pequeño script PHP que compruebe si hay cookies establecidas. **Sug:** crear alguna cookie y comprobar después con `count($_COOKIE)` y/o `var_dump($_COOKIE)`.
4. Cerrar el navegador web y volver a ejecutar el script. ¿Qué ocurre ?. Ejecutar el mismo script desde otro navegador web. ¿Qué ocurre ?. Dar una explicación razonada de todo lo ocurrido.
5. Codificar un script PHP que presente el número de veces que un usuario accede a una página. La primera vez, además de mostrar el valor 1, mostrará el mensaje *"Nuevo en el lugar"*. La segunda y sucesivas veces, mostrará el valor "n" del número de visitas y el momento que el usuario entró por última vez. Realizarlo primero mediante cookies y después mediante sesiones.

6. Crear un script PHP que tenga un formulario que llame a una segunda página PHP y que solicite el nombre y apellidos de un usuario y los almacene en sendas variables de **sesión** cuando se pulse el botón de enviar. En esta segunda página se verificará si existen ambas variables de sesión. En caso afirmativo, dará la bienvenida al usuario mostrando su nombre y apellidos. En caso contrario, mostrará un mensaje indicando que no puede visitar la página. Tendrá un hipervínculo al formulario inicial.
7. Modificar la segunda página del ejercicio anterior añadiendo un segundo hipervínculo a una tercera página PHP que se comportará exactamente igual que la segunda página. A ambas páginas (2ª y 3ª) añadirles otro hipervínculo a una cuarta página que lo único que hará será **cerrar la sesión** y redirigir automáticamente a la primera página.
Probar las diferentes situaciones cuando la sesión está abierta y cuando no lo está.

Ejemplo de Listado de páginas: Ejer-7.php, Ejer-7-a.php, Ejer-7-b.php, Ejer-7-cerrarsesion.php

8. Crear un formulario de tipo **login** (como punto de parte partida, vale alguno de los utilizados en temas anteriores) donde los usuarios corresponden al siguiente array asociativo:

```
$credenciales = array(
    array('nombre' => 'Ana', 'apell' => 'García', 'clave' => 'clave1', 'rol' => 'admin'),
    array('nombre' => 'Jose', 'apell' => 'González', 'clave' => 'clave2', 'rol' => 'usuario'),
    array('nombre' => 'Margarita', 'apell' => 'Ayuso', 'clave' => 'clave3', 'rol' => 'admin'),
    array('nombre' => 'Antonio', 'apell' => 'Merino', 'clave' => 'clave4', 'rol' => 'usuario')
);
```

Cuando el usuario se autentique correctamente, se creará una variable de sesión y se le remitirá a una página de administración o a una página de consultas según el valor del *rol* del usuario, respectivamente.

En cualquiera de estas dos páginas se mostrará en la cabecera el nombre del usuario y un mensaje indicando si se está en una página de administración o en una página de consultas. También tendrán un enlace [Cerrar Sesión](#) para cerrar la sesión y enviar automáticamente al usuario a la página de Login.

Además, si no hay sesión establecida, se tendrá que impedir acceder a cualquier página que no sea el *Login*. Por ejemplo, si un usuario introduce en el navegador web una URL como <http://localhost/aplic/Ejer-8-admin.php>, sin estar establecida la sesión, no se le dejará entrar *Ejer-8-admin.php* y se le redirigirá automáticamente a la página *Ejer-8-login.php*.

Ejemplo de Listado de páginas: Ejer-8.php, Ejer-8-admin.php, Ejer-8-usuario, Ejer-8-cerrarsesion.php.