



DESARROLLO WEB EN ENTORNO CLIENTE

· INTERACCIÓN CON EL USUARIO: EVENTOS Y
FORMULARIOS

5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



Cookie:

- Fichero de texto propio de cada navegador.
- Se almacena en diferentes lugares según el navegador y el sistema operativo.
- Se comenzaron a utilizar para guardar la información de los carritos de compra.
- Son específicas del dominio: un dominio no puede acceder a la cookie de otro dominio.
- Tienen una fecha de caducidad: como el número de cookies es limitado, el propio navegador se encarga de borrarlas.



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



Formato de registro de una cookie:

- Dominio del servidor que creó la cookie.
- Información de si es necesaria una conexión http segura para acceder a la cookie.
- Trayectoria de las URL que podrán acceder a la cookie.
- Fecha de caducidad de la cookie.
- Nombre de una entrada de datos.
- Cadena de texto asociada a ese nombre de entrada de datos.



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



- En Windows 7 y Vista:
“\Usuarios\<Tu_usuario>\AppData\Roaming\Mozilla\Firefox\Profiles\serie_de_números_y_letras.default”.
- En Unix/Linux en la carpeta:
“~/.mozilla/”.
- En Mac OS X: “~/Library/Mozilla/”.

Almacenamiento: “cookies.sqlite” (a partir de la versión 3 de Firefox). También os encontraréis con los archivos “cookies.sqlite-shm”, “cookies.sqlite-wal” y “cookies-nontor.xml” para poder explorar (darle formato visual) la base de datos de cookies.



- En Windows vista o 7:
“\Usuarios\<Tu_usuario>\AppData\Local\Google\Chrome\User Data\Default”.
- En Mac OS X:
“~/Library/Preferences/Opera Preferences/cookies4.dat”.
- En UNIX y distros Linux:
“~/.config/google-chrome/Default”.

Almacenamiento: “cookies” sin extensión en formato base de datos.

5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



- Windows Vista y 7:
“\Usuarios\<Tu_usuario>\AppData\Roaming\Opera\Opera”.
- Mac OS X: “~/Library/Cookies”.
- Unix y otras distros Linux: “~/.opera/”.

Almacenamiento: “cookies4.dat” en formato binario.



- En Windows 7 y Vista:
“\Usuarios\<Tu_usuario>\AppData\Roaming\Microsoft\Windows\Cookies”.
- Almacenamiento: en formato texto, en archivos codificados con extensión .txt.



- Windows Vista y 7:
“\Usuarios\<Tu_usuario>\AppData\Roaming\Apple Computer\Safari\Cookies”.
- Mac OS X: “~/Library/Cookies”.
- Unix y otras distros Linux: no hay una versión oficial del navegador de Apple para Linux.

Almacenamiento: “Cookies.binarycookies” en formato binario.

5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES

Opciones de utilización:

Mantener
opciones de
visualización.

Almacenamiento
de variables.

Seguimiento de
la actividad del
usuario.

Autenticación de
usuarios.



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



Grabar una cookie:

`document.cookie = "nombreCookie = datosCookie`

`[; expires=horaFormatoGMT]` Ej. `expires=01 Jan 70 00:00:00 GMT`;

`[; path=ruta]` Ruta actual de nuestra página web.

`[; domain=nombreDominio]` Por defecto, el que creó la cookie.

`[; secure]` Si no se pone, puede ser accesible por cualquier programa que se
ajuste al dominio y path

Borrar una cookie:

Se le asigna una fecha de caducidad anterior a la actual.

Ejemplos:

`document.cookie = "usuario=Laura; expires=2 Dec 2014, 23:59:59 GMT";`

`document.cookie = "contador=0";`

Si la cookie no existe, la crea; si existe, la sobrescribe.



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES

Función para inicializar una cookie:

```
function setCookie(cname, cvalue, exdays) {  
    var d = new Date(); //Se extrae la fecha actual  
    d.setTime(d.getTime() + (exdays*24*60*60*1000)); //Se le suman los días  
    indicados por parámetro  
    var expires = "expires="+d.toUTCString(); //Se pasa a formato GMT  
    document.cookie = cname + "=" + cvalue + "; " + expires; //Se crea la cookie  
}
```



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



Leer una cookie:

`var x = document.cookie; //Devuelve toda la cadena con todas las variables;`

Leer un valor concreto de la cookie:

```
function getCookie(cname) {  
    var name = cname + "="; //Crea una variable con el texto del campo y un igual  
    var ca = document.cookie.split(';'); //divide la cookie en trozos separados por ;  
    for(var i=0; i<ca.length; i++) { //Recorre cada trozo de la cookie  
        var c = ca[i];  
        while (c.charAt(0)==' ') c = c.substring(1);  
        if (c.indexOf(name) == 0) //Si encuentra el campo buscado  
            return c.substring(name.length,c.length); //lo devuelve  
    }  
    return ""; //Sino devuelve vacío  
}
```



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.6. LAS COOKIES



Comprobar si una cookie está inicializada:

```
function checkCookie() {  
    var username=getCookie("username");//Se extrae el valor (ver función  
    anterior)  
    if (username!="") { //Si no está vacío  
        alert("Bienvenido de nuevo, " + username); //Se le saluda  
    }else{//Sino  
        username = prompt("Introduzca sunombre:", ""); //Se le solicita con prompt  
        if (username != "" && username != null) {  
            setCookie("username", username, 365); //Se inicializa la variable  
        }  
    }  
}
```



5. INTERACCIÓN CON EL USUARIO: EVENTOS Y FORMULARIOS

5.5. EXPRESIONES REGULARES



EJERCICIO: u5e5_cookies:

- *Crea un fichero que se llame “cookies.js” y que permita trabajar con las cookies de manera que puedas crear, borrar y consultar una cookie. Consulta los métodos tratados en clase y recuerda que el método borrar no ha sido creado aún así que deberás crearlo tú mismo.*
- *Crea un archivo HTML sencillo con el que pruebes probar “cookies.js”:*
 - *Pon un elemento HTML que al hacer click incremente un contador.*
 - *Pon un elemento HTML que al hacer click devuelva el valor del contador.*
 - *Pon un elemento HTML que resetee el contador.*
 - *¿El contador se mantiene al cerrar el navegador?*

