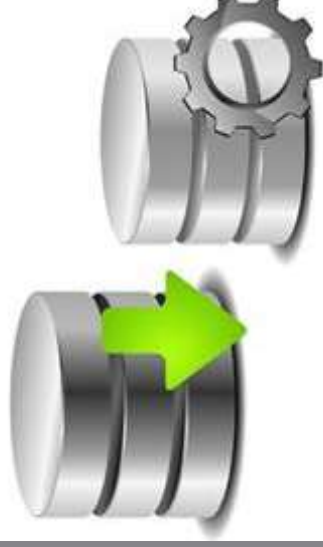




HTML



Parte 4: Almacenamiento local

UD3: Interacción con el usuario y modelo de objetos
del documento

María Rodríguez Fernández mariafer@educastur.org

Al final de este documento...



- Sabrás lo que es una cookie
 - Habrás aprendido a crear, modificar, eliminar y consultar una cookie con JavaScript
 - Habrás aplicado lo aprendido a un caso “real”
- Conocerás la alternativa que ofrece HTML5 para las Cookies, y comprobarás sus ventajas
 - Probarás los principales métodos
 - Aplicarás lo aprendido a un caso “real”

Cookies

- Son **ficheros de texto** que se almacenan en determinada carpeta en el navegador
 - Chrome
 - [user] \ Local Settings \ Application Data \ Google \ Chrome \ User Data \ Default.
 - Mozilla. En el perfil del navegador.
 - [windows]\Application Data\Mozilla\Profiles\[profilename]\;
 - IE: Archivos temporales de Internet.
 - Etc.
- A grandes rasgos:

```
todasLasCookies = document.cookie; //Mostrar todas las cookies  
document.cookie = nuevaCookie; //Escribir una cookie
```

Cookies: crear una cookie

- Sintaxis ([] → Opcional)

**Obligatorio
aunque sea
vacío**

Si no existe una cookie con el nombre la creará automáticamente.
Si ya existe una cookie con el nombre se reemplaza.

```
document.cookie = "nombreCookie=datosCookie";  
[; expires=horaformatoGMT]  
[; max-age=duracion]  
[; path=ruta]  
[; domain=nombreDominio]  
[; secure] "
```

– **Ejemplo:**

```
document.cookie="username=manolo";  
document.cookie="genero=masculino";
```

Grabar una cookie: Otros parámetros

- **expires.** Fecha de caducidad
 - expires=Thu, 01-Jan-15 00:00:01 GMT;
 - Si no se establece la cookie dura el tiempo de la sesión (hasta cierre de navegador).
- **max-age.** Duración máxima en segundos
- **path.** Ruta actual de nuestra Web
- **domain.** Si no se pone es el dominio de la página que creó la cookie
- **secure.** Nuestra cookie será accesible por cualquier programa en el dominio
- **Ejemplo:**

```
document.cookie="username=manolo;  
expires=Thu, 16 Nov 2025 12:00:00 UTC";
```

Cookies: Recuperar una cookie

- Accedemos a `document.cookie`
 - Debemos obtener la cadena de texto de la misma y extraer los datos necesarios de su contenido.

```
function getCookie(nomCookie) {  
    var cook=document.cookie.split(";"); // pares de valores  
  
    for (var i=0; i<cook.length; i++) { // revisamos todos los pares  
        var n = cook[i].split("="); // separamos nombre/valor  
        var nombre=n[0];  
        var valor =n[1];  
        if (nombre.trim()===nomCookie.trim()) // si es el buscado  
            return valor;// devolvemos su valor  
        }  
    return null; // si no se encuentra = nulo  
}
```

Cookies: Modificar y borrar una cookie

- **Modificar:** Se le asigna un nuevo valor usando el mismo nombre que el crearla

– **IMPORTANTE:** Hay que repetir todos los parámetros que en su creación, sino se crea una cookie nueva

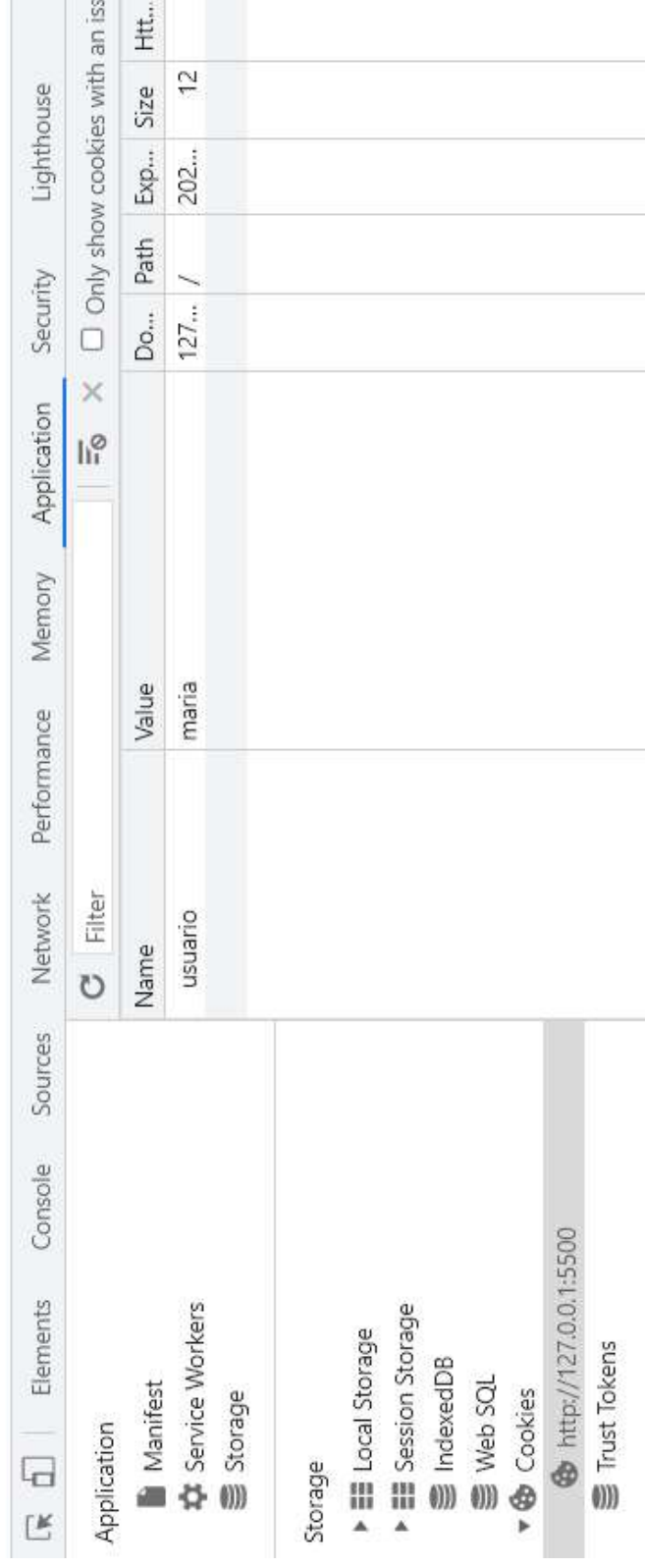
```
document.cookie="genero=femenino";
```

- **Borrar:** Se fija su fecha de expiración a un momento pasado:

```
document.cookie="username=; expires=Thu, 01 Jan 1970 00:00:01 GMT;";  
document.cookie="genero=; expires=Thu, 01 Jan 1970 00:00:01 GMT; ";
```

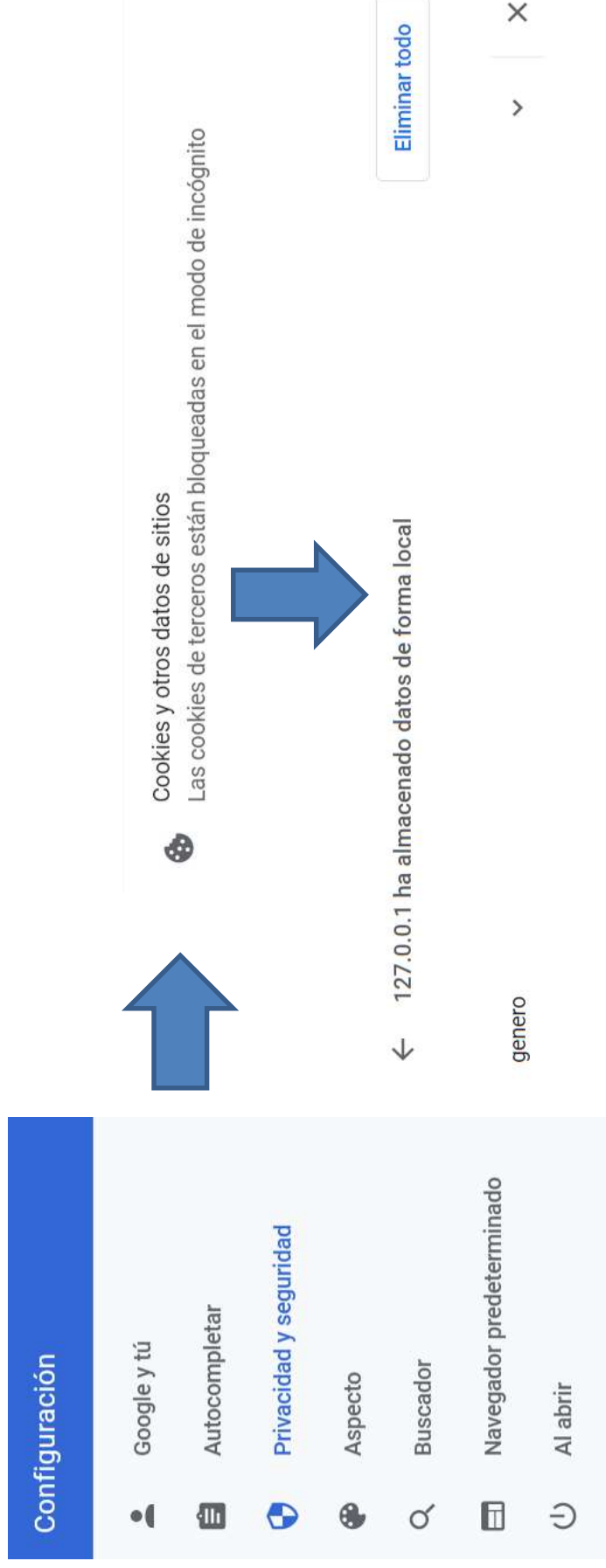
En la consola...

- En Chrome, podemos ver las Cookies en la sección **Application** de la consola:



En nuestro navegador...

- En la sección de configuración de nuestro navegador podemos gestionar las cookies. Por ejemplo, en Chrome:



Códigos útiles con Cookies

- Función crear Cookie (para no repetir el mismo código):

```
function setCookie(nombre, valor, caduca) {  
    var hoy = new Date();  
    hoy.setTime(hoy.getTime()+caduca);  
    var expiration = "expires="+d.toUTCString();  
    document.cookie = nombre+"="+valor+";"+expiration+";path=/";  
}
```

- Antes de trabajar con cookies podemos comprobar si están activas en el navegador
 - Ej: Podemos intentar guardar un valor y luego recuperarlo

```
function haycookies(){  
    document.cookie="micookie=hay";  
    return ( getCookie("micookie") == "hay" );  
}
```

EJERCICIO PROPUESTO V (1 de 2)

- A partir del siguiente código HTML, implementa la funcionalidad de cada botón creando una función para cada manejador

Ver todas las cookies

Crear cookie

Modificar cookie

Leer cookie

Borrar cookie

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <title>Cookies - Ejemplo completo</title>
</head>
<body>
  <button type="button" id="verTodas">Ver todas las cookies</button>
  <button type="button" id="crearCookie">Crear cookie </button>
  <button type="button" id="modificarCookie">Modificar cookie </button>
  <button type="button" id="leerCookie">Leer cookie </button>
  <button type="button" id="borrarCookie">Borrar cookie </button>
</body>
</html>
```

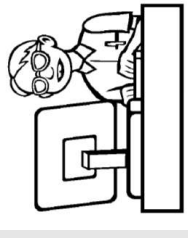
EJERCICIO PROPUESTO V (2 de 2)

- **verCookies:** accede a **document.cookie** y lo muestra. Se le llamará también después de cada operación realizada.
- **crearModifCookie:** pide nombre, valor y días para expirar con prompt y llama a **setCookie** para crearla:
 - **setCookie (nombre, valor, fecha):** crea una nueva cookie con los valores que se le pasan, calculando la fecha de expiración a partir de la actual, añadiéndole los segundos correspondientes (*ver transparencia 11*)
- **borrarCookie:** pide el nombre y llama a **deleteCookie:**
 - **deleteCookie (nombre):** borra la cookie poniendo su fecha a 0
- **leerCookie:** pide el nombre de la cookie y llama a **getCookie:**
 - **getCookie (nombre):** recorre todas las cookies de `document.cookie` y compara el nombre con el buscado. Si lo encuentra devuelve el valor.

¡OJO! Es necesario que ejecutes el código en un servidor (ej. Go Live de VS Code)

EJERCICIO PROPUESTO IV

(Continuación)



- Modifica el formulario para que las comprobaciones se hagan al pulsar el botón "Enviar Datos"
 - `checkValidity` al hacer click en el botón
 - Evento `invalid` en cada campo para mostrar el error
 - Evento `input` para limpiar el error
- Mejora el formulario del ejercicio propuesto V para que:
 - Guarde el número de intentos en una cookie llamada "contador"
 - Haz que no se pueda mandar el formulario cuando se supere un número máximo de intentos

Nombre	<input type="text"/>
Apellidos	<input type="text"/>
Fecha de nacimiento	<input type="text" value="dd/mm/aaaa"/>
DNI	<input type="text"/>
Email	<input type="text"/>
Contraseña	<input type="password"/>
Contraseña (repetición)	<input type="password"/>
Género	<input checked="" type="radio"/> Hombre <input type="radio"/> Mujer
<input type="checkbox"/> Suscribirse al boletín de novedades	
<input type="checkbox"/> Informarme sobre ofertas	
Producto favorito	<input type="text" value="Elige uno..."/>
Comentario	<input type="text" value="Introduce tu comentario..."/>
<input type="button" value="Enviar datos"/>	
<div>Este es tu primer intento</div>	

¿Cansado/a de introducir tus datos?

- Usa el panel Grabador
 - Pestaña "Recorder" de DevTools
 - Crea una nueva grabación en el botón "+"
 - Comienza la grabación
 - Introduce los datos
 - Finaliza la grabación
 - Pulsa Replay para reproducir esas acciones.

