
Actividades UD3



04/12/2025
CURSO:2ºDAW

Desarrollo entorno cliente
Celia Caravaca Vega

índice

ACTIVIDADES UD3:	1
Preguntas teóricas	1
1. Cookies y almacenamiento web:	1
2. localStorage y sessionStorage:	1
3. Manipulación de strings	2
4. Objetos en JavaScript	2
5. Funciones en JavaScript	2
6. Temporizadores	3
7. Arrays y colecciones	3

ACTIVIDADES UD3:

Cookies, localStorage, Strings, Objetos y Funciones.

Preguntas teóricas

1. Cookies y almacenamiento web:

a. **Explica qué es una cookie y para qué sirve en el desarrollo web.**

- Una cookie es un pequeño archivo de texto para recopilar información del usuario temporalmente.
- Sirve para que esa información que recopilamos del usuario sea utilizada de manera que tenga una experiencia más personalizada.

b. **Diferencia entre expires y max-age.**

- Expires se usa para definir una fecha límite de uso a la cookie una vez terminado se elimina.
- max-age la podemos usar para borrar la cookie(aunque su función principal es decir cuánto durará la cookie).

c. **¿Qué tamaño máximo suele tener una cookie?**

- Generalmente es de 4 KB.

d. **Explica por qué es necesario usar `encodeURIComponent()` al guardar una cookie.**

- Porque codifican y descodifican los caracteres especiales en las cookies para que no tengamos errores.

e. **¿Cómo se borra una cookie?**

- Las cookie se borran normalmente si no tienen fecha de vencimiento en cuanto la sesión del usuario se expire, para borrarla nosotros tenemos que volverla a crear con una fecha pasada o utilizar el parámetro max-age=0

2. **localStorage y sessionStorage:**

- Son propiedades que acceden al objeto Storage y almacenan datos de manera local.

a. **Diferencias entre localStorage y sessionStorage.**

- LocalStorage almacena la información de forma indefinida o se decide limpiar los datos del navegador mientras que sessionStorage almacena información mientras la pestaña esté abierta.

b. **¿Qué métodos tiene localStorage para guardar y recuperar información?**

- Existen dos formas de guardar información dentro de localStorage:

- 1 `localStorage.setItem(name,content);`
- 2 `localStorage.name = content;`

- Para recuperar información también tenemos dos maneras de hacerlo:

- 1 `localStorage.getItem(name)`
- 2 `localStorage.name`

c. **Explica un caso real donde sería adecuado usar sessionStorage.**

- Lo normal sería utilizarlo con datos irrelevantes que no nos importara perder como por ejemplo un formulario que al recargar la página pierda la información de las casillas.

3. Manipulación de strings

a. ¿Qué diferencia hay entre `.substring()` y `.substr()`?

- `substring(par1,par2)` devuelve el substring(subcadena) desde la posición del primer parámetro hasta el segundo parámetro.
- `substr(par1,tamaño)` devuelve el substring desde el primer parámetro hasta la longitud del segundo parámetro OJO contando la longitud del primero.

b. ¿Qué hace `.startsWith()` y qué devuelve?

- Comprueba si la cadena empieza por el primer parámetro introducido, devuelve un boolean.

c. ¿Qué diferencia existe entre `.replace()` y `.replaceAll()`?

- `replace` devuelve la cadena con solo la primera iteración que se encuentra del patrón introducido o texto.
- `replaceAll` lo hace con todas las iteraciones que se encuentra.

4. Objetos en JavaScript

a. Explica dos formas de crear objetos en JavaScript.

- 1º Forma:

```
let objeto={nombre:"Claudia"};
```

- 2º Forma:

```
let objeto=new Object({nombre:"Claudia"});
```

b. ¿Qué hace el operador `in` en un bucle `for (campo in objeto)`?

- Recorrer todos los campos declarados de ese objeto en este caso las claves.

c. ¿Qué diferencia hay entre acceder a una propiedad con `usuario.nombre` y `usuario['nombre']`?

- Por punto solo permite acceder a propiedades cuyo nombre es un identificador válido, es decir:
 - No pueden contener espacios
 - No pueden empezar por números
 - No pueden incluir caracteres especiales (salvo _ y \$)
 - Deben conocerse literalmente en el código
- Por corchetes permite acceder a propiedades usando cadenas, por lo que admite:
 - Nombres dinámicos (variables)
 - Propiedades con espacios
 - Propiedades con caracteres especiales
 - Propiedades con nombres numéricos

5. Funciones en JavaScript

a. ¿Qué es una función anónima?

- Es una función a la que no se le define el nombre y se almacena una variable y nomas llamar la variable como si fuera una función se ejecuta.

b. ¿Qué es una IIFE? Pon un ejemplo.

- Es una función que solo se ejecuta una vez y no es accesible desde afuera y sus variables desaparecen.
- Ejemplo:

```
(function (){  
    let mensajeSecreto="Hola carolina";  
    console.log(mensajeSecreto);  
})()
```

c. ¿Qué ventajas ofrecen las funciones flecha?

- Sintaxis más corta y clara, ideal para fusiones cortas.

6. Temporizadores

a. ¿Qué diferencia hay entre `setTimeout()` y `setInterval()`?

- `setTimeout()` Se ejecuta una vez después pasado el tiempo indicado en milisegundos
- `setInterval()` Se ejecuta repetidamente cada cierto intervalo de tiempo indicado en milisegundos.

b. ¿Qué hace `clearTimeout()`? ¿Y `clearInterval()`?

- `clearTimeout()` cancela un `setTimeout()` que aún no se ha ejecutado.
- `clearInterval()` cancela un `setInterval()` deteniendo la repetición.

7. Arrays y colecciones

a. ¿Para qué sirve `filter()` en un array?

- Filtra cada uno de los elementos del array con una condición y los elementos que cumplan la condición se pasan a un array nuevo.

b. ¿Qué diferencia hay entre Map y Array?

- Map es una estructura de datos pares clave-valor mientras que array es otra estructura de datos.

c. ¿Qué devuelve `map.get(key)` si la clave no existe?

- Devuelve undefined .