

# JavaScript desde cero

Módulo 2

# Consola

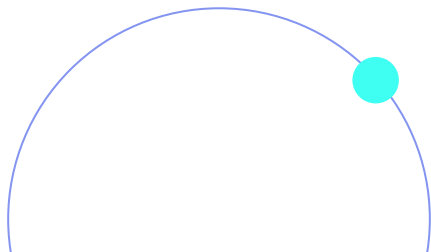
# Consola de JavaScript

La consola de JavaScript es una herramienta **disponible en todos los navegadores**. Sirve para realizar la **depuración del código**.

Cuando hablamos de depuración (*debug* en inglés) nos referimos al **mantenimiento y corrección de un código fuente**.

La consola ofrece dos herramientas principales:

- **La vista del navegador** (mediante las teclas *F12, Ctrl + Shift + I* ó *Click Derecho > Inspeccionar*). Esta vista nos permite **leer** las salidas por consola.
- **El objeto console** (mediante la palabra 'console' en JavaScript). Este objeto nos permite **escribir** en la consola.



Los métodos que explicaremos en los siguientes *slides* son:

- `console.assert.`
- `console.clear.`
- `console.count.`
- `console.countReset.`
- `console.debug.`



## *console.assert*

Registra un mensaje y envía una traza de error a la consola si el primer argumento es falso.

Este método sirve para **evaluar y probar un código mediante los *Assertion Test***, que son pruebas en las que se evalúa el estado de una variable. A diferencia de un if/else, `console.assert` es **útil solo para el desarrollador**.

```
let edad = 18  
  
console.assert( edad > 18, "Eres mayor de edad")
```



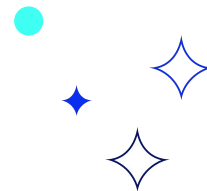
`console.assert`

## *console.clear*

Limpia la consola, **borra todos los mensajes previos**.

Es muy útil para cuando tenemos mucho por depurar, incluir este método en el código para hacer más legible el área de depuración de la aplicación web.

```
// Limpia la consola JS de múltiples mensajes  
  
console.clear()
```



### *console.count*

Registra el **número de veces** que esta línea ha sido llamada con la etiqueta dada:

```
console.count("Mi contador")
console.count("Mi contador")
console.count("Mi contador")
console.count("Mi contador")
```

### *console.countReset*

Reinicia el **valor del contador** para la etiqueta dada:

```
console.count("Mi contador")
console.count("Mi contador")

console.countReset("Mi contador")

console.count("Mi contador")
console.count("Mi contador")
```

## *console.debug*

Registra un mensaje con el nivel de *debug*. El nivel *debug* es un nivel **menos grave** que el error.



```
console.debug("Este es un mensaje de error/depuración")
```





# Cronómetros: temporizador

Una herramienta útil para **medir el rendimiento** de un programa es el *temporizador*.

Con el temporizador es posible evaluar **cuánto tarda el navegador en ejecutar un código determinado**.

Con esta información se puede verificar si es necesario cambiar el código para que funcione más rápido.

```
console.time("MediaPerformance")

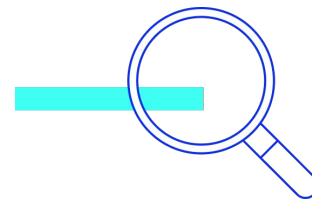
let valor = console.time("MediaPerformance")
console.log(valor)

console.timeEnd("MediaPerformance")
```



# Revisión

- Examinar la **interfaz de la consola**.
- Comprender que la consola te sirve como desarrollador para el **mantenimiento de tu código**.
- Hacer ***Assertion Test*** con la consola.
- **Mostrar diferentes tipos de mensajes** por la consola.
- **Medir el tiempo de ejecución** de un programa.



**¡Sigamos  
trabajando!**