

# { JavaScript }



## OBJETOS JSON

JavaScript Object Notation (JSON), es un formato de datos basado en texto que sigue la sintaxis de objeto de JavaScript.

Un objeto JSON puede ser almacenado en su propio archivo, que es básicamente sólo un archivo de texto con una extensión .json, y una MIME type (en-US) de application/json.

- JSON es sólo un formato de datos — contiene sólo propiedades, no métodos.
- JSON puede tomar la forma de cualquier tipo de datos que sea válido para ser incluido en un JSON, no sólo arreglos u objetos. Así, por ejemplo, una cadena o un número único podrían ser objetos JSON válidos.
- En algunas ocasiones, se recibirá una cadena JSON sin procesar, y será necesario convertirla en un objeto. Y cuando sea necesario enviar un objeto Javascript a través de la red, será necesario convertirlo a un JSON (una cadena) antes de ser enviado.

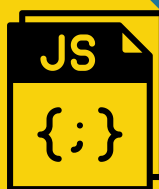
## ESTRUCTURAS DE CONTROL

Se utiliza para controlar el flujo de las operaciones durante el programa.

- If/Else
- SWITCH
- BUCLE FOR
- BUCLE WHILE
- BUCLE DO/WHILE
- Sentencias break y continue: Break: Salir o saltar fuera un bucle, finalizándolo. Continue: Saltar hasta la siguiente iteración del bucle. Permite saltar una o más iteraciones.



## LET- CONST vs. VAR



**Const:** Las variables declaradas con const mantienen valores constantes. Solo se puede acceder a las declaraciones const dentro del bloque en el que fueron declaradas.

**Let:** Un bloque es un fragmento de código delimitado por {}. Un bloque vive entre llaves. Cualquier cosa entre llaves es un bloque. Entonces, una variable declarada en un bloque con let solo está disponible para usar dentro de ese bloque.

**Var:** Las declaraciones de var tienen un alcance global o una función/un alcance local. El alcance es global cuando una variable var se declara fuera de una función. Tiene un alcance local cuando se declara dentro de una función. Esto significa que está disponible y solo se puede acceder a ella dentro de esa función.

## FUNCIONES

**Normales:** Se crean mediante la declaración "funcion", indicando después el nombre de la función y un paréntesis de apertura y otro de cierre.

**Funciones flecha:** Para declarar estas funciones se declara primero una variable y luego, se escribe el conjunto de símbolos =>, que son la representación de una flecha. El nombre de las funciones flecha viene precisamente de este símbolo. Tras la flecha, se coloca el cuerpo de la función entre llaves.

**Anónimas:** no es accesible después de su creación inicial, solo se puede acceder a ella mediante una variable en la que está almacenada como una función como un valor . Una función anónima también puede tener varios argumentos, pero solo una expresión.



# { JavaScript }

## Truthy y Falsy

- En JavaScript, un valor Truthy es un valor que se considera verdadero cuando se encuentra en un contexto booleano. Todos los valores son verdaderos a menos que se definan como falsos. Es decir, todos los valores son verdaderos excepto falso, 0, -0, 0n, "", null, undefined y NaN.
- Un valor Falsy(a veces escrito falsey) es un valor que se considera falso cuando se encuentra en un contexto booleano. JavaScript utiliza la conversión de tipos para forzar cualquier valor a un valor booleano en contextos que lo requieran, como condicionales y bucles.

## Short Circuit

En JavaScript, el cortocircuito es la evaluación de una expresión de izquierda a derecha con `||` y operadores `&&`.

Si la condición se cumple y el resto de las condiciones no afectarán el resultado ya evaluado, la expresión hará un short-circuit y devolverá ese resultado (valor).

## Null vs. Undefined (Diferencias)

Cambos sirven para algo muy parecido, que es indicar la ausencia de valor. Lo que ocurre es que undefined es un valor que denota que no hay valor porque no se ha definido todavía, mientras que null se usa para indicar que no hay valor porque así lo hemos querido indicar expresamente.

la diferencia principal, es que Undefined, es asignado por JS y solo por JS de forma automática como valor inicial en caso de que no le asignes un valor. En cambio el valor Null, debe ser asignado mediante el código que escribamos.