

Manejo de la sintaxis del lenguaje

--

*Sentencias,
decisiones y bucles*

Sentencias y declaraciones por categorías	3
Control de flujo	3
Declaraciones	3
Funciones	4
Iteraciones	4
Otros	5
Especificaciones	5

1. Sentencias y declaraciones por categorías

Las aplicaciones JavaScript se componen de sentencias con una sintaxis propia. Una sentencia puede estar formada por múltiples líneas. Puede haber varias sentencias en una sola línea si separamos cada una de las sentencias por un punto y coma. No es una palabra clave, sino un grupo de palabras clave.

a. Control de flujo

Block

Un bloque de sentencias se utiliza para agrupar cero o más sentencias. El bloque se delimita por un par de llaves.

break

Finaliza la sentencia actual loop, switch, o label y transfiere el control del programa a la siguiente sentencia de la sentencia finalizada.

continue

Finaliza la ejecución de las sentencias dentro de la iteración actual del actual bucle, y continúa la ejecución del bucle con la siguiente iteración.

Empty

Una sentencia vacía se utiliza para proveer una "no sentencia", aunque la sintaxis de JavaScript esperaba una.

if...else

Ejecuta una sentencia si una condición especificada es true. Si la condición es falsa, otra sentencia puede ser ejecutada.

switch

Evalúa una expresión, igualando el valor de la expresión a una cláusula case y ejecuta las sentencias asociadas con dicho case.

throw

Lanza una excepción definida por el usuario.

try...catch

Marca un bloque de sentencias para ser probadas (try) y especifica una respuesta, en caso de que se lance una excepción.

b. Declaraciones

var

Declara una variable, opcionalmente inicializándola a un valor.

let

Declara una variable local de ámbito de bloque, opcionalmente inicializándola a un valor.

const

Declara una constante de solo lectura.

c. Funciones

function

Declara una función con los parámetros especificados.

function*

Declara una función generadora. Las funciones que todos conocemos tratan de un fragmento de código que se ejecuta de principio a fin cuando las invocamos. Estas funciones no pueden ser pausadas para ejecutar otro código entre medias, ni tampoco podemos cancelarlas.

Sin embargo, las funciones generadoras rompen con ese comportamiento: Estas pueden ser pausadas y se pueden retomar en otro momento con el fin de que puedas hacer otras tareas entre medias, o incluso cancelarlas si fuera necesario. (Extraído de <https://www.returngis.net/2019/01/para-que-sirve-function/>)

async function

Declara una función asíncrona con los parámetros especificados.

return

Especifica el valor a ser retornado por una función.

class

Declara una clase.

d. Iteraciones

do...while

Crea un bucle que ejecuta una instrucción especificada mientras la condición sea verdadera.

for

Un ciclo for es una estructura iterativa para ejecutar un mismo segmento de código una cantidad de veces deseada; conociendo previamente un valor de inicio, un tamaño de paso y un valor final para el ciclo.

for each...in

Itera una variable especificada sobre todos los valores de las propiedades del objeto. Para cada propiedad distinta, se ejecuta la instrucción especificada.

for...in

Itera sobre las propiedades de un objeto (puede ser un array), ejecutando las instrucciones especificadas.

for...of

La sentencia `for...of` ejecuta un bloque de código para cada elemento de un objeto iterable, como lo son: `String`, `Array` y otros objetos similares a arrays.

while

Crea un bucle que ejecuta la instrucción especificada siempre que la condición de prueba se evalúe como verdadera. La condición se evalúa antes de ejecutar la instrucción.

e. Otros

debugger

Invoca cualquier funcionalidad de depuración disponible. Si no hay funcionalidad de depuración disponible, esta instrucción no tiene efecto.

export

Usada para permitir a un script firmado proveer propiedades, funciones y objetos a otros scripts firmados o sin firmar. Esta antigua funcionalidad de Netscape ha sido eliminada y será redefinida por los módulos de ECMAScript 6.

import

Usada para permitir a un script importar propiedades, funciones y objetos desde otro script firmado que ha exportado su información. Esta antigua funcionalidad de Netscape ha sido eliminada y será redefinida por los módulos de ECMAScript 6.

label

Etiqueta para referirse a una instrucción en las sentencias `break` o `continue`.

with

Extiende la cadena de alcance para una instrucción.

Especificaciones

Especificación	Status	Comentario
ECMAScript 1st Edition (ECMA-262) La definición de 'Statements' en esta especificación.	Standard	Definición inicial
ECMAScript 3rd Edition (ECMA-262) La definición de 'Statements' en esta especificación.	Standard	
ECMAScript 5.1 (ECMA-262) La definición de 'Statements' en esta especificación.	Standard	

[ECMAScript 2015 \(6th Edition, ECMA-262\)](#)
[La definición de 'ECMAScript Language: Statements and Declarations' en esta especificación.](#)

Standard

Nuevo: function*, let, for...of, yield, class

[ECMAScript \(ECMA-262\)](#)
[La definición de 'ECMAScript Language: Statements and Declarations' en esta especificación.](#)

Living
Standard