

Alumno/a: Manuel Ariza Serrano

Estándares, variables y operadores

1. Define qué es ECMA

Ecma International es una organización internacional basada en membresías de estándares para la comunicación y la información. La organización fue fundada en 1961 para estandarizar los sistemas computarizados en Europa. La membresía está abierta a las empresas que producen, comercializan o desarrollan sistemas computacionales o de comunicación en Europa.

2. Define qué es ECMA-262 y su relación con JavaScript.

ECMAScript es una especificación de lenguaje de programación publicada por ECMA International. El desarrollo empezó en 1996 y estuvo basado en el popular lenguaje JavaScript propuesto como estándar por Netscape Communications Corporation. Actualmente está aceptado como el estándar ISO 16262.

3. Indica las dos últimas versiones del ECMA-262 y sus fechas de actualización.

ECMAScript 6, cuyo nombre oficial es en realidad ECMAScript 2015. En su momento nos hicimos eco del lanzamiento y de sus novedades. Añade muchas cosas nuevas al lenguaje, algunas de las cuales se planeaban para la fallida versión 4, y otras son nuevas. Lleva el lenguaje a un nivel superior, con muchas mejoras en cuanto a sintaxis, pero también conceptos nuevos como los símbolos o las lambdas, o tipos de datos que no existían, estructuras mejoradas para iteración, etc, etc...

ECMAScript 7, cuyo nombre oficial es ECMAScript 2016. En realidad lo que añade es mínimo: básicamente el operador de exponenciación y un método nuevo para las matrices que permite comprobar si existen ciertos elementos dentro de éstas.

4. Indica qué es ES5 y ES6

ES5 o ES2015 es el nuevo estándar de Javascript (versión 6) y tiene algunos cambios significativos en su sintaxis que permite escribir código de forma más rápida y limpia. Actualmente los navegadores y NodeJS usan la versión 5 o también conocido como ES5.

ES6 es el estándar que sigue JavaScript desde Junio de 2015. Hasta ese momento la versión de JS que estábamos usando en nuestros navegadores y Node.js, era la v5.

5. ¿Qué sucede si se usa una variable sin declaración previa (let, var ó const)? Responde dentro y fuera de una función.

Si se usa una variable sin declaración previa sube al principio de la función(si está dentro de una función) y al principio del programa(si no está en una función). Es lo que se conoce como hoisting.

6. Hoisting. Indica qué es.

Es una característica que tiene JavaScript por la cual las variables que se usan sin declaración previa se subirían al principio de la función (si está en una función) o al principio del programa.

7. Tipos de datos en JavaScript.

JavaScript tiene seis tipos primitivos: Sin definir (`undefined`), Nulo (`null`), Lógicos (`boolean`), Numérico (`number`), Cadena (`string`), Símbolo (`symbol`).

8. Diferencia entre `typeof` y `typeof()`. Qué son y qué valores devuelven

Con paréntesis es para agrupar varios datos y se los comprueba todos a la vez, mientras que sin paréntesis es para solo un dato.

9. Ámbito de bloque y ámbito de función. Indica qué son con respecto a la definición de las variables.

De bloque se refiere a las estructuras de control como `if`, `for`, etc. Mientras que funciones son las funciones que te permite crear JavaScript.

10. Cuál es el operador exponencial.

Es `**`.

11. ¿Qué es el operador ternario? Indícalo con un ejemplo

Es un operador utilizado en programación para estructuras condicionales. Un ejemplo: `var max = (num1 > num2) ? num1 : num2`

12. Indica diferencia entre el operador igualdad (`==`) y el operador identidad o igualdad estricta (`===`). Indícalo mediante un ejemplo.

La diferencia es que el operador de `==` se usa para comparar dos datos sin tener en cuenta que sean del mismo tipo mientras que el operador `===` sí que lo tiene en cuenta.