

# Tarea 2.4 Estándares, variables y operadores

Desarrollo Web en Entorno Cliente

Rafael López Cruz

### Contenido

Ejercicio 1 .....	3
Ejercicio 2 .....	3
Ejercicio 3 .....	3
Ejercicio 4 .....	3
Ejercicio 5 .....	3
Ejercicio 6 .....	3
Ejercicio 7 .....	4
Ejercicio 8 .....	4
Ejercicio 9 .....	4
Ejercicio 10 .....	4
Ejercicio 11 .....	5
Ejercicio 12 .....	5

### Ejercicio 1. Define qué es ECMA

Es una especificación de lenguaje de programación publicada por ECMA International. Está basado en el lenguaje JavaScript. ECMAScript define un lenguaje de tipos dinámicos ligeramente inspirado en Java y en otros lenguajes de estilo de C y soporta algunas características de la programación orientada a objetos.

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

ECMA-262 es una estandarización que de Ecma International y Netscape les envió un borrador de JavaScript para su estandarización en noviembre de 1996. En junio de 1997 fue adoptada por asamblea general de Ecma.

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

- ECMAScript 2019. Décima edición publicada en junio de 2019.
- ECMAScript 2018. Novena edición publicada en junio de 2018.

### Ejercicio 4. Indica qué es ES5 y ES6

ES5 es ECMAScript 2009 y fue lanzada en diciembre de 2009. Esta versión agrego nuevas características para Arrays y JSON principalmente.

ES6 es ECMAScript 2015 y fue lanzada en junio de 2015. Esta versión agrego el let y const. Agrego también nuevas características para Arrays y clases, funciones arrow de javascript y valores por defecto en parámetros.

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

Pues que es como si las declaramos al principio del código y es global.

### Ejercicio 6. Hoisting. Indica qué es.

El hoisting es ese comportamiento que hace que una variable sin declaración previa la mueva a la cima del código global o de la función.

### Ejercicio 7. Tipos de datos en JavaScript.

Primitivos:

- String
- Number
- Boolean
- Null
- Undefined
- Symbol

Objeto:

- Object

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

typeof devuelve el tipo de dato de lo que le hayamos pasado. Con paréntesis o sin paréntesis devuelve el mismo resultado. Por ejemplo:

```
typeof 0 // "number"
```

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

Los ámbitos de bloque quiere decir todo aquello que solo está dentro de un bloque. Por ejemplo una variable definida con let y const solo va a estar dentro del bloque que fue definida.

Los ámbitos de función funcionan de forma de que el compilador de JavaScript comprueba si la variable que se llama en una función comprueba si una variable está definida dentro de la función o está fuera en el ámbito global. Si no fue declarada en ningún momento, el compilador la declara como variable global.

### Ejercicio 10.Cuál es el operador exponencial.

Evalúa el resultado de elevar primero el operador a la potencia del segundo operando. Por ejemplo:

Operador:  $x ** y$

Significado:  $x = x ** y$

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

El operador ternario es el único operador de JavaScript que tiene tres operandos. Este operador se usa con frecuencia como atajo para la instrucción if. Esta es su sintaxis:

Condición ? expresión1 : expresión2;

Por ejemplo:

(miEdad > 18) ? "Eres mayor de edad" : "No eres mayor de edad";

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

El operador igualdad(==) quiere indicar si dos valores son iguales. Por ejemplo un 0 number y un '0' string, los dos son iguales.

El operador igualdad estricta(===) quiere indicar si dos valores son iguales y del mismo tipo. Por un ejemplo 0 number y un '0' strings, los dos tienen el mismo valor pero al ser de distinto tipo son diferentes.