

## ¿Qué son las variables?

¡Hola! Seguimos profundizando en los aspectos más importantes de JavaScript; en este sentido vamos a destacar la definición y uso de las variables. En programación, las variables son contenedores que nos permiten almacenar valores numéricos, cadenas de caracteres o estructuras de datos como listas, arreglos de valores u objetos. Por el momento, vamos a estudiar las variables que almacenan valores simples como números o cadenas de caracteres. La principal propiedad de las variables, como su nombre lo indica, es almacenar valores que pueden cambiar a lo largo de la ejecución del programa; es por esta razón que reciben el nombre de variables.

Estos valores almacenados se pueden utilizar o acceder en cualquier momento de la ejecución de nuestro programa. Así como en álgebra utilizamos variables para trabajar con ecuaciones y fórmulas complejas, en programación trabajamos con variables para poder ejecutar los pasos que componen los algoritmos de la aplicación que estamos codificando.

Las variables en los lenguajes de programación siguen una lógica similar a las variables utilizadas en otros ámbitos como las matemáticas. Una variable es un elemento que se emplea para almacenar y hacer referencia a otro valor. Gracias a las variables es posible crear “programas genéricos”, es decir, programas que funcionan siempre igual independientemente de los valores concretos utilizados.

De la misma forma que si en Matemáticas no existieran las variables no se podrían definir las ecuaciones y fórmulas, en programación no se podrían hacer programas realmente útiles sin las variables.

Si no existieran variables, un programa que suma dos números podría escribirse como:

$$\text{resultado} = 3 + 1$$

El programa anterior es tan poco útil que sólo sirve para el caso en el que el primer número de la suma sea el 3 y el segundo número sea el 1. En cualquier otro caso, el programa obtiene un resultado incorrecto.

Sin embargo, el programa se puede rehacer de la siguiente manera utilizando variables para almacenar y referirse a cada número:

*numero\_1 = 3*

*numero\_2 = 1*

*resultado = numero\_1 + numero\_2*

Los elementos `numero_1` y `numero_2` son variables que almacenan los valores que utiliza el programa. El resultado se calcula siempre en función del valor almacenado por las variables, por lo que este programa funciona correctamente para cualquier par de números indicado. Si se modifica el valor de las variables `numero_1` y `numero_2`, el programa sigue funcionando correctamente.

Ya que vimos algunas propiedades de las variables, pasemos a aprender cómo utilizarlas en JavaScript.

## **Declaración de variables: estructura básica**

Para declarar una variable es necesario tener presente la siguiente estructura:

*var x = 5;*

A continuación, se describen los componentes de la estructura básica para la declaración de variables en JavaScript:

- `var` es la palabra reservada que le indica a JavaScript que estamos declarando una variable. También son utilizadas para este objetivo las palabras reservadas `let` y `const`, pero vamos a ver más adelante las características de cada declaración.
- `x`, en el caso del ejemplo, es el nombre que le asignamos a la variable.
- `=` es el operador de asignación que le indica a JavaScript qué valor asignar.
- `5` corresponde al valor inicial que vamos a almacenar en la variable.

## **Buenas prácticas**

Los nombres de las variables son muy flexibles, aunque te recomendamos estas buenas prácticas para nombrar tus variables:

- Los nombres deben expresar el contenido de la variable, así hacemos nuestro código más entendible a otros programadores; por ejemplo, si en la variable almacenamos la edad de un estudiante entonces la variable puede llamarse: edad, edadEstudiante o edadAlumno.
- El nombre de las variables puede contener los caracteres, underscore ( \_ ), o signo de dólar ( \$ ) al inicio, en medio o al final del nombre.
- El nombre de las variables debe iniciar con una letra ASCII sea mayúscula o minúscula, también puede empezar con un guión bajo ( \_ ) o un signo de dólar ( \$ ).
- Para separar las palabras que componen el nombre de la variable, JavaScript utiliza la convención camelCase, la cual indica que las variables empiezan en minúscula y cuando cambias de palabra inicias el primer carácter con letra mayúscula. Por ejemplo: para nombrar la variable color del carro puedes utilizar los siguientes nombres: colorCarro o colorDelCarro.
- No debes usar palabras reservadas de JavaScript para nombrar variables.
- El nombre de las variables no debe empezar con números.
- El nombre de las variables no puede contener caracteres especiales como: &, ~, %, #.
- Las variables en JavaScript nunca pueden iniciar con números, pero sí pueden contener números en la mitad o al final del nombre.
- 

A continuación tenés unos ejemplos de nombres de variables válidos en JavaScript:

- `var pizza2go = "pizza";`
- `var aprendo_JAVASCRIPT = true;`
- `var $_$ = "moneda";`
- `var _saludo = "Hola";`
- `var nombreVariable = "camel case";`

Aquí puedes ver algunos ejemplos de nombres inválidos para las variables:

- *var function = true;*
- *var total% = 125;*
- *var class = "variables";*
- *var 2nombre = "clase de variables";*

Las variables son sensibles a mayúsculas de manera que *miVar*, *MiVar*, *mivar* son variables completamente diferentes. Es una buena práctica de programación, evitar nombres similares entre las variables, o nombres que no nos den una referencia clara de cuál puede ser su contenido, por ejemplo:

*var x = true;*

Después de declarar una variable, es posible hacer referencia a ésta (utilizar u obtener el valor de la variable) por su nombre en cualquier parte de nuestro programa, teniendo en cuenta el scope o ámbito de las variables que estudiaremos más adelante. Así como también podemos hacer diferentes operaciones con las mismas.

*var valor1 = 5;*

*var valor2 = 7;*

*valor1 + valor2;*

Resultado

12

En el ejemplo anterior se declararon dos variables (*valor1*) y (*valor2*) que luego son utilizadas en una suma; al sumar las dos variables, el resultado es 12.

También es posible hacer uso de variables cuando estamos declarando otras variables:

```
var suma = valor1 + 95;
```

Después de declarar variables ( `var` ) podemos re-asignar nuevos valores o datos para que sean almacenados en dichas variables:

```
var nombre = "carlos";
```

```
nombre = "pedro";
```

```
nombre;
```

Resultado

Pedro

En JavaScript, otra manera de declarar variables se puede lograr usando los comandos `let` y `const`, palabras reservadas para el lenguaje, definidas en el estándar ECMAScript 6. Con el comando `let` declaramos variables limitando su uso a un bloque de código determinado, mientras que con el comando `const` definimos constantes o variables que tendrán un valor que no podrá modificarse posteriormente.