



Funciones



Las funciones son uno de los bloques de construcción fundamentales en JavaScript.

Una función en JavaScript es similar a un procedimiento (**un conjunto de instrucciones que realiza una tarea**).

Podemos decir que una función es un bloque de instrucciones donde se realizará cierto **algoritmo/tarea/instrucción** que **devolverá un resultado** o modificara ciertos valores para un futuro.

Y también podemos recalcar que la funciones son principalmente usadas para ejecutar **código repetitivo**.

Tenemos varias formas de declararlas, así que vamos a verlas.

Function Declaration

Funciones - Function declaration

La manera más común de definir una función es usando la palabra clave **function** seguida del nombre, los **parámetros** de entrada -los cuales son opcionales- entre paréntesis y separados por comas, así como las instrucciones encerradas en llaves {}.

Un punto importante de las Function Declaration es que se cargan antes de que se ejecute cualquier código, por lo que se pueden llamar incluso antes de su declaración 😱. Este comportamiento es conocido como **Hoisting**.

¿Cuándo podemos usarlas?

Al ser la forma más fácil y común, es mi primer opción cuando quiero crear una función normal que necesita ser llamada una o más veces 😊.

Funciones - Function declaration



```
function saludar(nombre){  
  return `Hola! Mi nombre es ${nombre}`;  
}  
  
saludar( 'Gustavo' );
```

Function Expression

La diferencia entre la forma anterior y ésta es que las **function expression pueden ser asignadas a una variable** y el nombre es opcional. Funciones que no tienen nombre son llamadas anonymous functions. Existe más de una forma de declararlas y esta es una de ellas.

¿Cuándo podemos usarlas?

Las function expressions son comunes cuando necesitamos pasar dicha función como un **parámetro a otra función**. Esto comúnmente se llama **callback** en JavaScript.

Funciones - Function Expression



```
const saludar = function(nombre, apellido){  
  return `Hola! Mi nombre es ${nombre} ${apellido}`;  
}
```

```
saludar('Gustavo', 'Rodríguez');
```

Funciones de Flecha (Arrow Function)



Funciones - Arrow Function

Una de las nuevas características que nos trajo **ES6** fueron las **Arrow function**. Al igual que las function expressions, éstas son funciones anónimas que **pueden ser asignadas a una variable**, la diferencia es que en las arrow functions, omitimos la palabra clave "function", por lo que lo primero que agregamos es la lista de parámetros, si no hay ninguno se indica con (). En caso de que solo haya un parámetro, los paréntesis no son necesarios. Luego ponemos el símbolo => y las llaves {}.

¿Cuándo podemos usarlas?

Puedes sacar el mejor provecho de las Arrow function cuando necesites crear **funciones simples** que devuelvan algo concreto, ya que ahorrarías líneas de código 💪.

Funciones - Arrow Function



```
const saludar = (nombre, apellido) => {  
  return `Hola! Mi nombre es ${nombre} ${apellido}`;  
}
```

```
const saludar2 = (nombre) => `Hola ${nombre}!`;
```

```
saludar('Gustavo', 'Rodríguez');  
saludar2('Senpai');
```

Buenas prácticas



Algunas buenas prácticas para las funciones son:

- Elegir nombres descriptivos.
- Que tengan una única responsabilidad.
- Que no posean más de 4 líneas de código.
 - Esta última no descarta anidaciones de funciones que en total sumen más líneas.

Estas buenas prácticas ayudan a poder leer el código de una forma más simple y una buena nomenclatura y definición de funciones permite evitarnos repetición de código

Ejercicios 🙌

Funciones - Ejercicios

Dado el **peso** y **altura** de un adulto, desarrollar una función para determinar el **índice de masa corporal**.

¿Cómo se calcula el IMC?

El IMC se calcula de la misma manera tanto para adultos como para niños. El cálculo se basa en las siguientes fórmulas:

Unidades de medida	Fórmula y cálculo
Kilogramos y metros (o centímetros)	<p>Fórmula: $\text{peso (kg)} / [\text{estatura (m)}]^2$</p> <p>Con el sistema métrico, la fórmula para el IMC es el peso en kilogramos dividido por la estatura en metros cuadrados. Debido a que la estatura por lo general se mide en centímetros, divida la estatura en centímetros por 100 para obtener la estatura en metros.</p> <p>Ejemplo: Peso = 68 kg, Estatura = 165 cm (1.65 m) Cálculo: $68 \div (1.65)^2 = 24.98$</p>

IMC	Nivel de peso
Por debajo de 18.5	Bajo peso
18.5 – 24.9	Normal
25.0 – 29.9	Sobrepeso
30.0 o más	Obesidad

LINKS

LINKS

- **Funciones**

- <https://dev.to/soyleninjs/crear-y-entender-funciones-en-javascript-3lkh>
- <https://devcode.la/tutoriales/funciones-en-javascript/>
- <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Functions>
- <https://dev.to/alecodesdancer/el-mundo-de-las-funciones-en-javascript-16n>

- **Hoisting**

- https://www.w3schools.com/js/js_hoisting.asp
- <https://developer.mozilla.org/es/docs/Glossary/Hoisting>



[gustavgueez](#)



[gustavgueez](#)



[gustavgueez](#)

GUSTAVO RODRIGUEZ

FULL STACK DEVELOPER
SOLCRE