

JavaScript desde cero

Módulo 5

Funciones flecha

Funciones flecha

Las funciones flecha son una forma alternativa de escribir una función del ES6. Proveen una serie de ventajas que no las hacen una simple sintaxis alternativa para funciones.

- Permiten ahorrar código.
- Permiten expresar funciones como expresiones.
- Mejoran la gestión del ámbito de las variables (no cambian el **this***).

*: La palabra **this** es una herramienta avanzada para gestionar las variables en JavaScript. Si bien es una herramienta avanzada, la mencionamos aquí para que tengas esta referencia cuando vayas a estudiarla.



Ahorrar código

En este caso, la función flecha permite ahorrar código omitiendo la palabra reservada **function**.

Las funciones flecha son expresiones que representan funciones, por lo que, para ser utilizadas como una función, deben asignarse a una variable.

Su sintaxis es muy simple

(parámetros) => proceso

```
1  function dividir(n1,n2) {  
2      if(n2 > 0)  
3          return n1 + n2;  
4      else  
5          return false;  
6  }  
7  
8  
9  // Con funciones flecha  
10 const dividir = (n1, n2) => {  
11     if(n2 > 0)  
12         return n1 + n2;  
13     else  
14         return false;  
15 }  
16
```

Escribir funciones como expresiones

```
1 function itemLista(nombre) {  
2   return '<li>'+nombre+'</li>'  
3 }  
4  
5  
6 // Con funciones flecha  
7 const itemLista = (nombre) => '<li>'+nombre+'</li>'  
8
```

—
Estos dos códigos
son equivalentes.

Las funciones flecha permiten ahorrar código en las funciones que sólo retornan un valor, que no tienen un proceso complejo dentro. Esto es especialmente útil cuando tengo una expresión (como un texto HTML) la cual necesito parametrizar.

En este caso, podemos omitir las llaves y la palabra reservada return en las funciones flechas.

¡Muchas gracias!

¡Sigamos trabajando!