**Tutorial JavaScript: Funciones Flecha (Arrow Functions)**

**1) ¿Qué es una función?**

Una **función** es un bloque de código que **realiza una tarea** cuando la llamas.

function saludar(nombre) {

return "Hola " + nombre;

}

console.log(saludar("Ana")); // "Hola Ana"

**2) ¿Qué es una función flecha?**

La **función flecha** es una forma **más corta** de escribir funciones.

const saludar = nombre => "Hola " + nombre;

console.log(saludar("Ana")); // "Hola Ana"

**Explicación:**

* saludar → nombre de la función.
* nombre → **parámetro** (variable dentro de la función que recibe el valor).
* => → flecha que indica “esto es una función”.
* "Hola " + nombre → se devuelve automáticamente (solo una línea).

**3) Variantes importantes**

* Sin parámetros:

const hola = () => "Hola mundo";

console.log(hola()); // "Hola mundo"

* Varios parámetros:

const sumar = (a, b) => a + b;

console.log(sumar(2, 3)); // 5

* Con varias líneas (usar {} y return si quieres devolver algo):

const doblarYMostrar = n => {

const r = n \* 2;

console.log("Doble:", r);

return r;

};

doblarYMostrar(4); // imprime "Doble: 8"

**4) Idea clave**

**La función flecha no recorre el arreglo por sí sola.**  
Quien recorre el arreglo es forEach.  
La flecha **recibe** cada valor como parámetro. Esa variable **va tomando los valores uno por uno**.

**5) Recorrer un arreglo con forEach y función flecha (paso a paso)**

**Ejemplo completo:**

const frutas = ["manzana", "pera", "uva"];

// Creamos una función flecha que recibe un elemento

const mostrar = fruta => {

console.log("Dentro de la arrow, fruta =", fruta);

};

// Usamos forEach para recorrer el arreglo

frutas.forEach(mostrar);

**Paso a paso:**

1. forEach toma el primer elemento "manzana" → llama a mostrar("manzana").
   * Dentro de la arrow, fruta = "manzana".
   * Se imprime: Dentro de la arrow, fruta = manzana.
2. Segundo elemento "pera" → mostrar("pera") → imprime: Dentro de la arrow, fruta = pera.
3. Tercer elemento "uva" → mostrar("uva") → imprime: Dentro de la arrow, fruta = uva.

**Ejemplo directo sin función aparte:**

const frutas = ["manzana", "pera", "uva"];

frutas.forEach(fruta => {

console.log("Elemento actual:", fruta);

});

**Paso a paso:**

* forEach toma cada elemento del arreglo y lo pasa a la flecha como parámetro.
* La variable del parámetro (fruta) **toma cada valor uno por uno**.
* Dentro de la arrow puedes **hacer cualquier acción**: imprimir, calcular, etc.

**6) Ejemplo práctico: duplicar valores de un arreglo**

const numeros = [2, 4, 6];

const duplicar = n => n \* 2;

const resultado = [];

numeros.forEach(n => {

const r = duplicar(n); // llamamos la arrow con el valor actual

resultado.push(r); // guardamos el resultado en un nuevo arreglo

});

console.log(resultado); // [4, 8, 12]

**Paso a paso para el primer elemento:**

1. n = 2 → duplicar(2) devuelve 4.
2. resultado.push(4) → el nuevo arreglo ahora tiene [4].
3. Se repite para 4 y 6.

**7) Resumen clave**

* La **arrow** es la función: const f = x => x \* 2;
* La **variable del parámetro** (x, fruta, n) **toma los valores del arreglo uno por uno** cuando se llama la arrow.
* forEach **recorre el arreglo automáticamente** y llama a la arrow por cada elemento.
* Dentro de la arrow puedes **imprimir, calcular o modificar** valores.

**8) Ejercicios básicos**

**Ejercicio 1 — Saludo simple**

Crea una arrow que reciba nombre e imprima Hola, NOMBR

**Ejercicio 2 — Recorrer un arreglo**

Dado el arreglo:

const numeros = [1, 2, 3];

Crea una arrow que reciba un número e imprima: "Número actual: N".  
Recorre el arreglo con forEach y llama a la arrow.

**Ejercicio 3 — Crear un nuevo arreglo**

Dado:

const nums = [2, 4, 6];

* Crea una arrow duplicar = n => n\*2.
* Recorre nums con forEach y guarda los resultados en un nuevo arreglo resultado.
* Imprime resultado al final.

**Ejercicio 4 — Sumar elementos de un arreglo**

Dado:

const nums = [3, 5, 7];

* Crea let suma = 0.
* Crea una arrow sumar = n => suma += n.
* Recorre nums con forEach y llama a sumar.
* Al final imprime suma (debe dar 15).

**Ejercicio 5 — Arrow sin parámetros**

Crea una arrow que no reciba nada y devuelva "Hola mundo!".  
Llama a la función y muestra el resultado.