

Guía de Arrays en JavaScript

¿Qué es un Array?

Un **array** en JavaScript es una estructura de datos que nos permite almacenar múltiples valores en una sola variable. Los valores dentro de un array pueden ser de cualquier tipo: números, cadenas de texto, booleanos, objetos, etc.

Ejemplo básico de un array:

```
let numeros = [1, 2, 3, 4]; // Un array de números

let frutas = ["manzana", "plátano", "cereza"]; // Un array de
cadenas de texto
```

En este ejemplo, `frutas` es un array que contiene tres elementos: "manzana", "plátano", y "cereza". Los arrays en JavaScript comienzan en el índice 0, por lo que "manzana" está en la posición 0, "plátano" en la 1, y "cereza" en la 2.

¿Qué es un Índice?

El índice es la posición de un elemento dentro de un array. En JavaScript, los índices empiezan desde 0, es decir, el primer elemento tiene el índice 0, el segundo el índice 1, el tercero el índice 2, y así sucesivamente.

Ejemplo de índices:

```
let frutas = ["manzana", "plátano", "cereza"];

// Índices:

// manzana -> índice 0

// plátano -> índice 1

// cereza -> índice 2
```

Acceder a los elementos de un array usando el índice:

```
console.log(frutas[0]); // "manzana" (índice 0)

console.log(frutas[1]); // "plátano" (índice 1)

console.log(frutas[2]); // "cereza" (índice 2)
```

Métodos para Modificar Arrays

1. Agregar un elemento al final del array

Para agregar un nuevo elemento al final de un array, usamos el método `.push()`.

Sintaxis:

```
array.push(elemento);
```

Ejemplo:

```
let frutas = ["manzana", "plátano", "cereza"];  
frutas.push("naranja"); // Agrega "naranja" al final  
console.log(frutas); // ["manzana", "plátano", "cereza",  
"naranja"]
```

2. Agregar un elemento al principio del array

Para agregar un nuevo elemento al principio de un array, usamos el método `.unshift()`.

Sintaxis:

```
array.unshift(elemento);
```

Ejemplo:

```
let frutas = ["manzana", "plátano", "cereza"];  
frutas.unshift("kiwi"); // Agrega "kiwi" al principio  
console.log(frutas); // ["kiwi", "manzana", "plátano", "cereza"]
```

3. Agregar un elemento en cualquier posición

Para agregar un elemento en una posición específica, usamos el método `.splice()`.

Sintaxis:

```
array.splice(indice, 0, elemento);
```

- índice: es la posición en la que quieres agregar el nuevo elemento.
- 0: indica que no queremos eliminar ningún elemento.
- elemento: el valor que queremos agregar.

Ejemplo:

```
let frutas = ["manzana", "plátano", "cereza"];  
  
frutas.splice(1, 0, "mango"); // Agrega "mango" en la posición 1  
  
console.log(frutas); // ["manzana", "mango", "plátano", "cereza"]
```

4. Eliminar el último elemento del array

Para eliminar el último elemento de un array, usamos el método `.pop()`.

Sintaxis:

```
array.pop();
```

Ejemplo:

```
let frutas = ["manzana", "plátano", "cereza"];  
  
frutas.pop(); // Elimina el último elemento  
  
console.log(frutas); // ["manzana", "plátano"]
```

5. Eliminar el primer elemento del array

Para eliminar el primer elemento de un array, usamos el método `.shift()`.

Sintaxis:

```
array.shift();
```

Ejemplo:

```
let frutas = ["manzana", "plátano", "cereza"];  
  
frutas.shift(); // Elimina el primer elemento  
  
console.log(frutas); // ["plátano", "cereza"]
```

6. Eliminar un elemento en cualquier posición

Para eliminar un elemento en una posición específica, usamos el método `.splice()`.

Sintaxis:

```
array.splice(indice, 1);
```

- `indice`: es la posición del elemento que quieres eliminar.
- `1`: indica que eliminaremos un solo elemento en esa posición.

Ejemplo:

```
let frutas = ["manzana", "plátano", "cereza"];
```

```
frutas.splice(1, 1); // Elimina el elemento en la posición 1  
console.log(frutas); // ["manzana", "cereza"]
```

Resumen

Aquí están los métodos que hemos visto:

1. `.push()`: Agregar un elemento al final del array.
2. `.unshift()`: Agregar un elemento al principio del array.
3. `.splice()`: Agregar un elemento en cualquier posición o eliminar uno en una posición específica.
4. `.pop()`: Eliminar el último elemento del array.
5. `.shift()`: Eliminar el primer elemento del array.