

# For y variantes en JavaScript

## For clásico

Iterar un número de veces definido/determinado.

```
// Con arrays
let frutas = ['manzana', 'fresa', 'naranja'];

for (let i = 0; i < frutas.length; i++) {
  console.log(frutas[i]);
}
```

```
// Con strings
let frase = "Programador: Persona que tiene relaciones más duraderas y profundas con una computadora que con otra persona.";

for (let i = 0; i < frase.length; i++) {
  console.log(frase[i]);
}
```

## for...in

Iterar sobre las **propiedades enumerables** de un objeto (las claves).

```
let persona = {
  nombre: 'Ana',
  edad: 30,
  ciudad: 'Madrid'
};

for (let clave in persona) {
  console.log(clave + ': ' + persona[clave]);
}
```

**NOTA:** También se puede usar con arrays, pero NO es recomendable, porque los índices se tratan como strings y puede incluir propiedades añadidas manualmente.

**for...in** recorre todas las claves enumerables, incluidas esas que no son índices del array.

```
let frutas = ['manzana', 'banana'];
frutas.colorFavorito = 'rojo';

for (let clave in frutas) {
  console.log(clave); // ¡Sorpresa!
}
```

## for...of

Iterar sobre valores de objetos iterables como arrays, strings, mapas, sets, ...

```
let frutas = ['manzana', 'banana', 'naranja'];

for (let fruta of frutas) {
  console.log(fruta);
}
```

## .forEach()

Método de los arrays que ejecuta una función para cada elemento

### Sintaxis

```
array.forEach((elemento, índice, arrayCompleto) => {
  // Aquí las sentencias de la función
});
```

```
let nombres = ['Ana', 'Luis', 'Carlos'];

nombres.forEach((nombre, i, arrayCompleto) => {
  if (i === arrayCompleto.length - 1) {
    console.log(`${nombre} es el último`);
  } else {
    console.log(`${nombre} no es el último`);
  }
});
```

```
let frutas = ['manzana', 'banana', 'naranja'];

frutas.forEach((fruta, i) => {
  console.log(`${i}: ${fruta}`);
});
```

```
let numeros = [3, 5, 7, 9, 11];

numeros.forEach((numero, i) => {
  console.log(`${i}: ${numero} x 2 = ${numero * 2}`);
});
```