

## MÉTODOS DE ARRAYS

Método	Qué hace	Ejemplo
<code>push()</code>	Añade un elemento al <b>final</b> del array.	<code>nums.push(5);</code>
<code>pop()</code>	Elimina el <b>último</b> elemento.	<code>nums.pop();</code>
<code>shift()</code>	Elimina el <b>primer</b> elemento.	<code>nums.shift();</code>
<code>unshift()</code>	Añade uno o más elementos al <b>inicio</b> .	<code>nums.unshift(1, 2);</code>
<code>concat()</code>	Une dos o más arrays y devuelve uno nuevo.	<code>arr3 = arr1.concat(arr2);</code>
<code>slice(inicio, fin)</code>	Extrae una parte sin modificar el original.	<code>arr.slice(1, 3);</code>
<code>splice(pos, num, elem)</code>	Quita o añade elementos (modifica el original).	<code>arr.splice(2, 1, "nuevo");</code>
<code>indexOf()</code>	Devuelve el índice del primer elemento igual.	<code>arr.indexOf("manzana");</code>
<code>includes()</code>	Devuelve <code>true</code> si contiene el valor.	<code>arr.includes("uva");</code>
<code>find()</code>	Devuelve el <b>primer elemento</b> que cumpla una condición.	<code>arr.find(x =&gt; x &gt; 10);</code>
<code>findIndex()</code>	Devuelve el <b>índice</b> del primer elemento que cumpla la condición.	<code>arr.findIndex(x =&gt; x &gt; 10);</code>
<code>filter()</code>	Devuelve un nuevo array con los elementos que cumplan una condición.	<code>arr.filter(x =&gt; x &gt; 10);</code>
<code>map()</code>	Crea un nuevo array aplicando una función a cada elemento.	<code>arr.map(x =&gt; x * 2);</code>
<code>reduce()</code>	Reduce el array a un único valor acumulando resultados.	<code>arr.reduce((a,b)=&gt;a+b,0);</code>

<b>sort()</b>	Ordena el array. <b>OJO:</b> lo modifica.	<code>arr.sort((a,b)=&gt;a-b);</code>
<b>reverse()</b>	Invierte el orden del array.	<code>arr.reverse();</code>
<b>forEach()</b>	Ejecuta una función por cada elemento (no devuelve nada).	<code>arr.forEach(x=&gt;console.log(x));</code>
<b>some()</b>	Devuelve <b>true</b> si <b>al menos un elemento</b> cumple la condición.	<code>arr.some(x=&gt;x&gt;5);</code>
<b>every()</b>	Devuelve <b>true</b> si <b>todos</b> cumplen la condición.	<code>arr.every(x=&gt;x&gt;5);</code>

---

## A B C D MÉTODOS DE STRINGS

Método	Qué hace	Ejemplo
<b>length</b>	Devuelve la longitud de la cadena.	<code>"hola".length;</code>
<b>toUpperCase()</b>	Convierte todo a mayúsculas.	<code>"hola".toUpperCase();</code>
<b>toLowerCase()</b>	Convierte todo a minúsculas.	<code>"HOLA".toLowerCase();</code>
<b>includes()</b>	Comprueba si contiene un texto.	<code>"hola mundo".includes("mundo");</code>
<b>startsWith()</b>	Comprueba si empieza con un texto.	<code>"hola".startsWith("ho");</code>
<b>endsWith()</b>	Comprueba si termina con un texto.	<code>"hola".endsWith("la");</code>
<b>indexOf()</b>	Posición de la primera coincidencia.	<code>"hola".indexOf("a");</code>
<b>slice()</b>	Extrae una parte del texto.	<code>"hola".slice(1,3); // 'ol'</code>
<b>replace()</b>	Reemplaza parte del texto.	<code>"hola".replace("ho", "Ha");</code>
<b>split()</b>	Convierte el string en array.	<code>"a,b,c".split(",");</code>
<b>trim()</b>	Quita espacios al inicio y fin.	<code>" hola ".trim();</code>



## MÉTODOS DE OBJETOS

Método / Función	Qué hace	Ejemplo
<code>Object.keys(obj)</code>	Devuelve array con los nombres de las propiedades.	<code>Object.keys(arbol);</code>
<code>Object.values(obj)</code>	Devuelve array con los valores.	<code>Object.values(arbol);</code>
<code>Object.entries(obj)</code>	Devuelve array de pares <code>[clave, valor]</code> .	<code>Object.entries(arbol);</code>
<code>hasOwnProperty()</code>	Comprueba si el objeto tiene esa propiedad.	<code>arbol.hasOwnProperty("_codigo");</code>
<code>delete</code>	Elimina una propiedad.	<code>delete arbol._codigo;</code>
<code>instanceof</code>	Comprueba si un objeto es instancia de una clase.	<code>a1 instanceof Perenne;</code>



## FUNCIONES Y CONCEPTOS GENERALES

Concepto	Qué hace	Ejemplo
<code>function</code> / <code>arrow function</code>	Define funciones normales o flecha.	<code>function suma(a,b){return a+b} / (a,b)=&gt;a+b</code>
<code>return</code>	Devuelve un valor desde una función.	<code>return resultado;</code>
<code>this</code>	Hace referencia al <b>objeto actual</b> .	<code>this._codigo = codigo;</code>
<code>class</code> / <code>extends</code> / <code>super</code>	Define clases y herencia.	<code>class Perenne extends Arbol { super(...); }</code>

<b>constructor</b>	Inicializa objetos al crearlos.	<code>constructor(codigo){this._codigo=codigo;}</code>
<b>get / set</b>	Definen métodos para acceder y modificar propiedades.	<code>get codigo(){return this._codigo;}</code>
<b>try...catch</b>	Controla errores en tiempo de ejecución.	<code>try{...}catch(e){console.log(e);}</code>
<b>alert() / prompt() / confirm()</b>	Muestran mensajes, piden datos o confirman.	<code>alert("Hola");</code>

## MÉTODOS Y FUNCIONES MATH

Método	Qué hace	Ejemplo
Math.PI	Devuelve $\pi$	<code>Math.PI; // 3.1415...</code>
Math.round(x)	Redondea al entero más cercano	<code>Math.round(3.6); // 4</code>
Math.floor(x)	Redondea hacia abajo	<code>Math.floor(3.9); // 3</code>
Math.ceil(x)	Redondea hacia arriba	<code>Math.ceil(3.1); // 4</code>
Math.abs(x)	Valor absoluto	<code>Math.abs(-5); // 5</code>
Math.sqrt(x)	Raíz cuadrada	<code>Math.sqrt(16); // 4</code>
Math.pow(x,y)	Potencia $x^y$	<code>Math.pow(2,3); // 8</code>
Math.random()	Número aleatorio entre 0 y 1	<code>Math.random(); // 0.0-0.999...</code>
Math.max(a,b,...)	Devuelve el mayor	<code>Math.max(5,10,2); // 10</code>
Math.min(a,b,...)	Devuelve el menor	<code>Math.min(5,10,2); // 2</code>



## MÉTODOS Y FUNCIONES DATE

Método	Qué hace	Ejemplo
<code>new Date()</code>	Crea una fecha actual	<code>let hoy = new Date();</code>

getFullYear()	Año de la fecha	hoy.getFullYear();
getMonth()	Mes (0-11)	hoy.getMonth(); // 0 = enero
getDate()	Día del mes	hoy.getDate();
getDay()	Día de la semana (0-6)	hoy.getDay(); // 0 = domingo
getHours()	Hora	hoy.getHours();
getMinutes()	Minutos	hoy.getMinutes();
getSeconds()	Segundos	hoy.getSeconds();
setFullYear(año)	Cambiar año	hoy.setFullYear(2025);
setMonth(mes)	Cambiar mes	hoy.setMonth(9); // octubre
setDate(día)	Cambiar día	hoy.setDate(28);

Método	Qué hace	Ejemplo
<code>toString()</code>	Devuelve la fecha en formato legible (sin hora)	<pre>let hoy = new Date(); hoy.toString(); // "Tue Oct 28 2025"</pre>
<code>toTimeString()</code>	Devuelve solo la hora	<pre>hoy.toTimeString(); // "14:35:07 GMT+0100 (CET)"</pre>
<code>toISOString()</code>	Devuelve la fecha en formato ISO	<pre>hoy.toISOString(); // "2025-10-28T13:35:07.123Z"</pre>
<code>toLocaleDateString()</code>	Devuelve fecha en formato local (país/idioma)	<pre>hoy.toLocaleDateString('es-ES') ; // "28/10/2025"</pre>
<code>toLocaleTimeString()</code>	Devuelve hora en formato local	<pre>hoy.toLocaleTimeString('es-ES') ; // "14:35:07"</pre>
<code>toLocaleString()</code>	Fecha y hora en formato local	<pre>hoy.toLocaleString('es-ES'); // "28/10/2025 14:35:07"</pre>

<code>getTime()</code>	Devuelve la fecha en <b>milisegundos desde 1970</b>	<code>hoy.getTime(); // 1748499307000</code>
<code>setHours(h,m,s,ms)</code>	Cambia hora, minutos, segundos y milisegundos	<code>hoy.setHours(15,0,0,0);</code>
<code>setMinutes(m)</code>	Cambia minutos	<code>hoy.setMinutes(30);</code>
<code>setSeconds(s)</code>	Cambia segundos	<code>hoy.setSeconds(0);</code>
<code>setMilliseconds(ms)</code>	Cambia milisegundos	<code>hoy.setMilliseconds(500);</code>