

Chuleta 1: Operadores de Asignación

Operator	Example	Same As
=	x = y	x = y
+=	x += y	x = x + y
-=	x -= y	x = x - y
*=	x *= y	x = x * y
/=	x /= y	x = x / y
%=	x %= y	x = x % y

El operador de **asignación** (=) asigna un valor a una variable.

% (Módulo de la división)

** (Exponente) 10**2

X==Y ¿Es X = Y?

X===Y ¿Es X estrictamente igual a Y?

X!=Y ¿Es X desigual a Y?

X!==Y ¿Es estrictamente desigual?

&& → Y

|| → o

X+=Y → x=x+y

X-=Y → x=x-y

X*=Y → x=x*y

X/=Y → x=x/y

+ Operador de cadena

```
txt1 = "John";
txt2 = "Doe";
txt3 = txt1 + " " + txt2;    // resultado "John Doe"
```

```
txt1 = "What a very ";
txt1 += "nice day";        //What a very nice day
```

Si se agrega un número y una cadena, el resultado será una cadena!

X++ → x=x+1

X-- → x=x-1

Incrementando:

```
var x = 5;
x++;
var z = x;
```

Decrementando:

```
var x = 5;
x--;
var z = x;
```

Imprimir en el documento

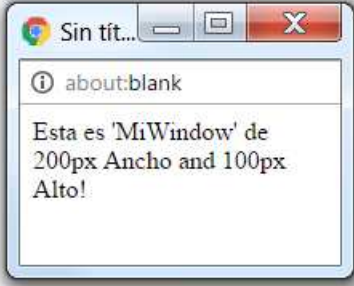
```
<!DOCTYPE html>
<html>
<body>

<p>Haz clic en el boton para abrir y escribir un texto.</p>

<button onclick="myFunction()">Aquí</button>

<script>
function myFunction() {
    document.open();
    document.write("<h1>Qué tal?</h1>");
    document.close();
}
</script>

</body>
</html>
```

Imprimir en una ventana nueva	<pre> <!DOCTYPE html> <html> <body> <p>Haz clic en el botón y se abrirá una nueva ventana "MiWindow" with some text.</p> <button onclick="myFunction()">Try it</button> <script> function myFunction() { var myWindow = window.open("", "MiWindow", "width=200,height=100"); myWindow.document.write("<p>Esta es 'MiWindow' de 200px Ancho and 100px Alto!</p>"); } </script> </body> </html> </pre> 
Salto de línea	<pre> <!DOCTYPE html> <html> <body> <p>write() no añade línea después del párrafo</p> <pre> <script> document.write("Hello World!"); document.write("Have a nice day!"); </script> </pre> <p>writeln() añade línea después del párrafo:</p> <pre> <script> document.writeln("Hello World!"); document.writeln("Have a nice day!"); </script> </pre> </body> </html> </pre>

Chuleta: Arrays	
Declarando un array. "Método Literal"	<pre>var cars = ["Saab", "Volvo", "BMW"]; var cars = ["Saab", "Volvo", "BMW"];</pre>
Operador NEW "Método el objeto Array"	<pre>var cars = new Array("Saab", "Volvo", "BMW");</pre>
- Modifico el primer valor	<pre>cars[0] = "Opel";</pre>
- Acceder a un Elemento	<pre>var cars = ["Saab", "Volvo", "BMW"]; document.getElementById("demo").innerHTML = cars[0];</pre>
- Acceder a todos los elementos.	<pre>var cars = ["Saab", "Volvo", "BMW"]; document.getElementById("demo").innerHTML = cars;</pre>
Matrices como propiedades de objetos	<pre>var person = {firstName:"John", lastName:"Doe", age:46}; person.firstName → John</pre> <pre>var person = []; person["firstName"] = "John"; person["lastName"] = "Doe"; person["age"] = 46; var x = person.length; // person.length → 0 var y = person[0]; // person[0] → undefined</pre>
Los elementos de una matriz pueden ser otras cosas	<pre>myArray[0] = Date.now; → Propiedad de un objeto myArray[1] = myFunction; → Una función myArray[2] = myCars; → Otro Array</pre>
Propiedades del Array	<pre>var x = cars.length; // length → Propiedad retorna "Nº de Elementos" var y = cars.sort(); // sort → Propiedad "ordena el array"</pre> <pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.length; // length fruits → 4</pre>
Bucle "for"	<pre>var fruits, text, fLen, i; fruits = ["Banana", "Orange", "Apple", "Mango"]; fLen = fruits.length; text = ""; for (i = 0; i < fLen; i++) { text += "" + fruits[i] + ""; }</pre>
Añadir un elemento	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.push("Lemon"); // añadimos (Lemon) a fruits</pre> <pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits[fruits.length] = "Lemon";</pre> <pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits[6] = "Lemon"; // ojo, con los agujeros</pre>
-Uso	<p>- Cuando los índices son números, usar: ¡Mejor opción!</p> <pre>var points = []; var points = [40, 100, 1, 5, 25, 10];</pre> <p>- Cuando los índices son cadenas de texto, usar: → Da problemas de elementos "undefined"</p> <pre>var points = new Array(); var points = new Array(40, 100, 1, 5, 25, 10);</pre>
Operador "typeof", para saber que es un array	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; typeof fruits; // retorna → Object</pre>
	<pre>Array.isArray(fruits); // retorna → true</pre>
Operador "instanceof" para saber si es Array, y es creado por Constructor.	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits instanceof Array // retorna → true</pre>

Métodos:	
toString() Convierte el Array a una cadena separada por comas.	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; document.getElementById("demo").innerHTML = fruits.toString();</pre> <p>→ Banana,Orange,Apple,Mango</p>
join() Convierte el Array a una cadena pero puedo especificar el separador	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; document.getElementById("demo").innerHTML = fruits.join(" * ");</pre> <p>→ Banana*Orange*Apple*Mango</p>
pop() Elimina el último elemento de una matriz	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; var x = fruits.pop(); // El valor de x es "Mango" fruits.pop(); // Elimina el ultimo element de la matriz.</pre>
push() Añade un elemento al final	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.push("Kiwi"); // Añade al final ("Kiwi") a fruits var x = fruits.push("Kiwi"); // Devuelve longitud del array 5</pre>
shift() Elimina el primer elemento y desplaza el resto a la izquierda, a un índice menor	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.shift(); // Elimina el primer elemento "Banana" de fruits</pre>
unshift() Añade un elemento al principio	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.unshift("Lemon"); // Añadir el elemento "Lemon" a fruits</pre>
Cambio de elementos	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits[0] = "Kiwi"; // Cambia el primer elemento de fruits a "Kiwi"</pre> <pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits[fruits.length] = "Kiwi"; // Añado "Kiwi" a fruit</pre>
Borrado de elementos delete	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; delete fruits[0]; // Cambio el primer element de fruits to undefined //!Ojo! como deja huecos es conveniente usar el método pop() ó shift() para no dejar huecos</pre>
splice() Para añadir varios elementos al mismo tiempo	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.splice(2, 0, "Lemon", "Kiwi");</pre> <p>El primer parámetro (2) posición en la que se añaden nuevos elementos. El segundo parámetro (0) el número de elementos debe ser eliminado. El resto de los parámetros ("Lemon", "Kiwi") los nuevos elementos que se añaden.</p>

splice() para eliminar elementos	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.splice(0, 1); // Borra el primer element de fruits</pre> <p>El primer parámetro (0) define la posición en la que se van a añadir elementos</p> <p>El segundo parámetro (1) define el número de elementos debe ser eliminado.</p> <p>El resto de los parámetros se omiten. No se agregarán nuevos elementos.</p>
concat() para unir dos arrays	<pre>var myGirls = ["Cecilie", "Lone"]; var myBoys = ["Emil", "Tobias", "Linus"]; var myChildren = myGirls.concat(myBoys); //Concatena myGirls y myBoys</pre> <pre>var arr1 = ["Cecilie", "Lone"]; var arr2 = ["Emil", "Tobias", "Linus"]; var arr3 = ["Robin", "Morgan"]; var myChildren = arr1.concat(arr2, arr3); //Concatena arr1, arr2 y arr3</pre>
slice() rebanar un trozo de matriz de uno o varios elementos	<pre>var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"]; var citrus = fruits.slice(1); //Crea una nueva matriz a partir del elemento con índice 1</pre> <pre>var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"]; var citrus = fruits.slice(3); //Crea una nueva matriz a partir del elemento con índice 3</pre> <pre>var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"]; var citrus = fruits.slice(1, 3); //Crea una nueva matriz a partir del elemento con índice 1 hasta el 3</pre>

valueOf() convierte un array en una cadena	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; document.getElementById("demo").innerHTML = fruits;</pre>
toString() convierte un array en una cadena	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; document.getElementById("demo").innerHTML = fruits.valueOf(); var fruits = ["Banana", "Orange", "Apple", "Mango"]; document.getElementById("demo").innerHTML = fruits.toString();</pre>

Ordenación:	
sort() ordena la matriz por orden alfabético	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.sort(); // Ordena los elementos de fruits</pre>
reverse() invierte los elementos de una matriz	<pre>var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.sort(); // Ordena los elementos de fruits fruits.reverse(); // Ordena en orden inverso los elementos</pre>
Si necesito ordenar números, ojo 22 es mas grande que 100	<pre>var points = [40, 100, 1, 5, 25, 10]; points.sort(function(a, b){return a - b}); // Orden ascendente var points = [40, 100, 1, 5, 25, 10]; points.sort(function(a, b){return b - a}); // Orden deccendente</pre> <p><u>La función de comparación:</u></p> <pre>function(a, b){return a-b}</pre> <p>devuelve un valor, positivo, negativo o cero.</p> <p><u>Ejemplo:</u></p> <ul style="list-style-type: none"> - Al comparar 40 y 100, el método sort () llama a la función de comparación (40.100). - La función calcula 40-100, y devuelve -60 (un valor negativo). - La función de clasificación clasificará 40 como un valor inferior a 100.
Ordenación en orden aleatorio	<pre>var points = [40, 100, 1, 5, 25, 10]; points.sort(function(a, b){return 0.5 - Math.random()});</pre>
Encontrar el valor más alto de una matriz	<pre>var points = [40, 100, 1, 5, 25, 10]; points.sort(function(a, b){return b - a}); // Ahora points[0] contiene el más alto valor</pre>
Encontrar el valor más pequeño	<pre>var points = [40, 100, 1, 5, 25, 10]; points.sort(function(a, b){return a - b}); // Ahora points[0] contiene el valor más bajo.</pre>

Ejemplo 1: Mostrar el Array en la página web

```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>

<script>
var cars = ["Saab", "Volvo", "BMW"];
document.getElementById("demo").innerHTML = cars;
</script>

</body>
</html>
```

Ejemplo 2: Propiedades de Objetos

```
<!DOCTYPE html>
<html>
<body>

<h1>JavaScript Objects</h1>
<p>JavaScript uses names to access object properties.</p>
<p id="demo"></p>

<script>
var person = {firstName:"John", lastName:"Doe", age:46};
document.getElementById("demo").innerHTML = person["firstName"];
</script>

</body>
</html>
```

Ejemplo 3:

Recorrer el Array.

```
<!DOCTYPE html>
<html>
<body>

<h1>JavaScript Arrays</h1>

<p>Recorremos el Array con un for:</p>
<p id="demo"></p>

<script>
var fruits, text, flen, i;

fruits = ["Banana", "Orange", "Apple", "Mango"];
flen = fruits.length;
text = "<ul>";
for (i = 0; i < flen; i++) {
    text += "<li>" + fruits[i] + "</li>";
}
text += "</ul>";
document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

Ejemplo 4: Creación de un array y mostrar su primer elemento.

```
<!DOCTYPE html>
<html>
<body>

<p> Usar [] en vez de new Array y dejar este para objetos.</p>

<p id="demo"></p>

<script>
//var points = new Array(40, 100, 1, 5, 25, 10);
var points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points[0];
</script>

</body>
</html>
```

Ejemplo 5: Ordenar Alfabéticamente y Numéricamente

Ordenar alfabéticamente y numéricamente.

```
<button onclick="myFunction1()">Ordenar Alfabéticamente</button>
<button onclick="myFunction2()">Ordenar Numéricamente</button>

<p id="demo"></p>

<script>
var points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;

function myFunction1() {
    points.sort();
    document.getElementById("demo").innerHTML = points;
}
function myFunction2() {
    points.sort(function(a, b){return a - b});
    document.getElementById("demo").innerHTML = points;
}
</script>
```