**TIPS Y MÉTODOS ÚTILES DE JAVASCRIPT**

**Índice**

[**TIPS:**](#_Toc42713645)

[**1. Utilizar un editor de código profesional**](#_Toc42713646)

[**2. Aprendizaje libre**](#_Toc42713647)

[**MÉTODOS:**](#_Toc42713648)

|  |  |  |
| --- | --- | --- |
| 1. [**Entrada y salida de datos**](#_Toc42713649) | 1. [**typeof**](#_Toc42713655) | 1. [**Métodos para Strings**](#_Toc42713656) |
| [**- console.log()**](#_Toc42713650)  [**- alert()**](#_Toc42713651)  [**- document.Write()**](#_Toc42713652)  [**- confirm()**](#_Toc42713653)  [**- prompt()**](#_Toc42713654) |  | [**- indexOf()**](#_Toc42713657)  [**- slice()**](#_Toc42713658)  [**- replace()**](#_Toc42713659)  [**- toUpperCase()**](#_Toc42713660)  [**- toLowerCase()**](#_Toc42713661)  [**- split()**](#_Toc42713662) |
| [**4. Métodos para Arrays**](#_Toc42713663)  [**- pop()**](#_Toc42713664)  [**- push()**](#_Toc42713665)  [**- splice()**](#_Toc42713666)  [**- sort()**](#_Toc42713667)  [**- join()**](#_Toc42713668) | [**5. Métodos para objetos**](#_Toc42713669)  [**- JSON.parse()**](#_Toc42713670)  [**- JSON.stringify()**](#_Toc42713671) | [**6. Tiempo y fechas**](#_Toc42713672)  [**- New Date()**](#_Toc42713673)  [**- getFullYear()**](#_Toc42713674)  [**- getMonth()**](#_Toc42713675)  [**- setFullYear() - setMonth()**](#_Toc42713676) |
| [**7. Métodos matemáticos**](#_Toc42713677)  [**- Math.random()**](#_Toc42713678)  [**- Math.max()**](#_Toc42713679)  [**- Math.round()**](#_Toc42713680) | [**8. Condición Ternaria**](#_Toc42713681) | [**9. HTML DOM (Document Object Model)**](#_Toc42713682)  [**- Métodos:**](#_Toc42713683)  [**- Cambiar elementos HTML:**](#_Toc42713684)  [**- Agregar y eliminar elementos HTML:**](#_Toc42713685)  [**- Algunos ejemplos:**](#_Toc42713686) |
| [**10. Eventos**](#_Toc42713687)  [**- onchange()**](#_Toc42713688)  [**- onclick()**](#_Toc42713689)  [**- onmouseover()**](#_Toc42713690)  [**- onmouseout()**](#_Toc42713691)  [**- onkeydown()**](#_Toc42713692)  [**- onload()**](#_Toc42713693) |  |  |

TIPS:

1. Utilizar un editor de código profesional

Existen editores muy completos que facilitan el desarrollo de software debido a sus funcionalidades extras que incluyen, por ejemplo, una terminal, variedad de plugins para ordenar el código, poner iconos a las extensiones de los archivos, etc. Entre los editores más usados para desarrollo web y móvil se encuentran:

* Visual Studio Code
* Sublime Text
* RJ TextEd
* Light Table
* Netbeans
* Brackets
* Komodo Edit
* Atom
* WebStorm

1. Aprendizaje libre

La academia w3schools ofrece tutoriales gratis en su página web <https://www.w3schools.com/js/default.asp> que puede incluir una certificación pero ya pagada.

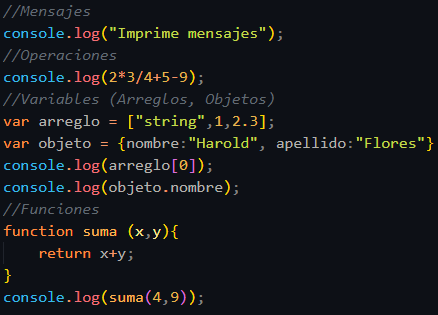
MÉTODOS:

1. Entrada y salida de datos

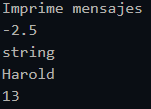
* console.log()

Imprime mensajes, operaciones, variables, etc. A través de la consola web o interprete de javascript.

*Script:*



*Consola:*



* alert()

Despliega una caja de texto en el navegador.

*Script:*



*Navegador:*

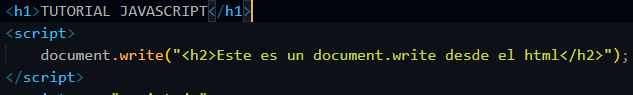


* document.Write()

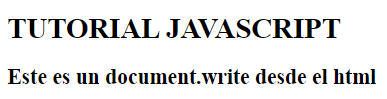
Esta funcionalidad modifica directamente el index.html del proyecto. Si index.html ya fue cargado completamente, antes de ejecutarse el script, el document.write borrará todo el index.html y lo sustituirá completamente con su contenido. Si el script se ejecuta antes que se complete la carga del index.html, entonces el contenido del document.write solo se añadirá al contenido del index.html.

* + Document.write antes de que se cargue completamente el index.html

*Html:*

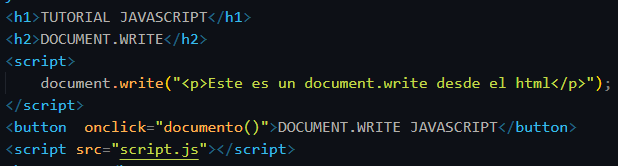


*Navegador:*

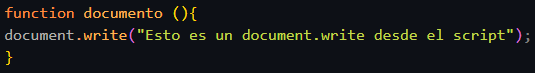


* + Document.write ejecutado una vez completada la carga del index.html.

*Html:*



*Script:*



*Navegador con carga completa del HTML:*



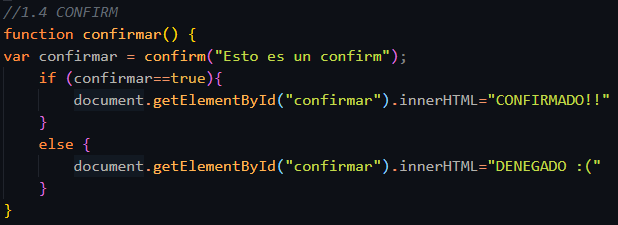
Resultado de ejecutar el document.write después de la carga completa del Html:



* confirm()

Muestra una ventana con las opciones de aceptar o cancelar. Estas opciones pueden almacenarse en una variable: si la opción escogida es aceptar, devuelve un valor de true; si la opción escogida es cancelar, devuelve un valor de false.

*Html:*



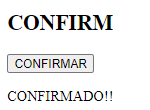
*Script:*



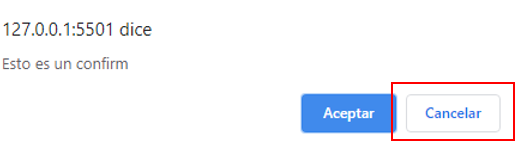
*Navegador – Confirmar (Aceptar):*

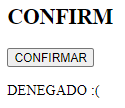






*Navegador – Confirmar (Cancelar):*

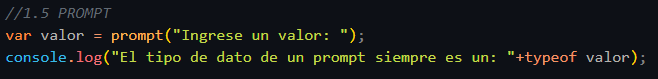




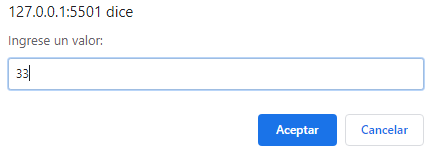
* prompt()

Esta funcionalidad permite capturar **texto** desde una ventana similar a la del alert y confirm. Siempre devuelve valores tipo string a menos que se cancele la opción solo ahí será tipo object.

*Script:*



*Navegador:*



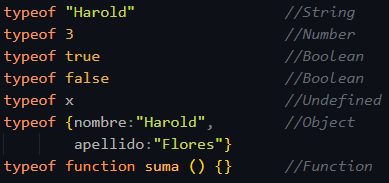
*Consola:*



1. typeof

Devuelve el tipo de dato de una variable o de una función. Puede devolver los valores: string, number, boolean, undefined, function y object.

*Script:*

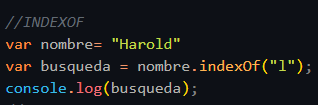


1. Métodos para Strings

* indexOf()

Encuentra la posición de un caracter en un string.

*Script:*



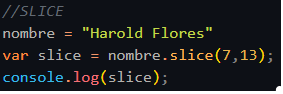
*Consola:*



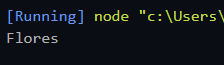
* slice()

Este método devuelve parte de un string.

*Script:*



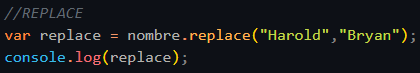
*Consola:*



* replace()

Reemplaza parte de un string con otro string.

*Script:*



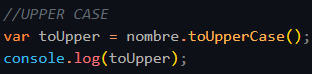
*Consola:*



* toUpperCase()

Transforma las letras de un string a mayúsculas.

*Script*:



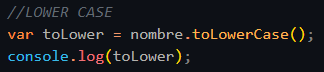
*Consola:*



* toLowerCase()

Transforma las letras de un string en minúsculas.

*Script*:



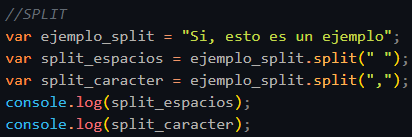
*Consola*:



* split()

Transforma un string en un array a partir de un carácter o espacio de división.

*Script*:



*Consola:*



* + Se puede limitar el tamaño del nuevo array.

*Script:*



*Consola*:

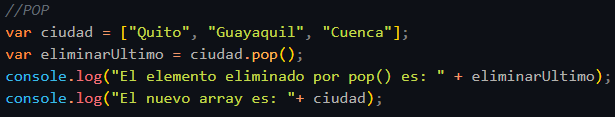


1. Métodos para Arrays

* pop()

Elimina el último elemento de un array.

*Script*:



*Consola*:

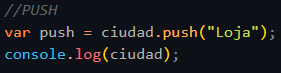


* push()

Agrega un nuevo elemento al Array.

*Script*:





*Consola*:



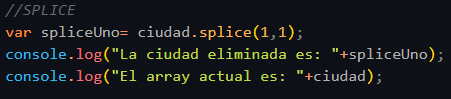
* splice()

Elimina uno o varios elementos de un Array. Splice(posición\_del\_item,numero\_de\_elementos\_a\_borrar).

* + Eliminar un elemento.

*Script:*





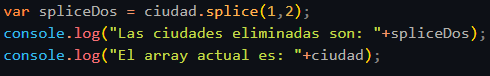
*Consola:*



* + Eliminar dos o más elementos.

*Script*:





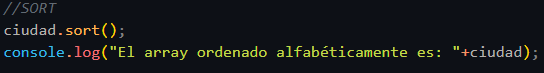
*Consola*:



* sort()

Ordena alfabéticamente un array.

*Script*:



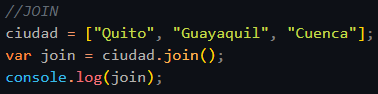
*Consola*:



* join()

Transforma un array en string.

*Script*:

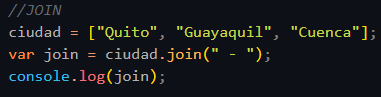


*Consola:*



* + Se puede incluir un string adicional dentro de la transformación.

*Script*:



*Consola*:

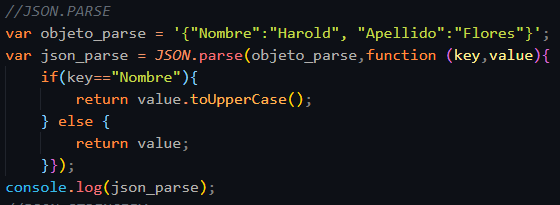


1. Métodos para objetos

* JSON.parse()

Transforma un string en un objeto javascript. Esta función recibe dos parámetros: el string y una función. La función es opcional y sirve para realizar cálculos u otras operaciones entre los datos del nuevo objeto por ejemplo:

*Script*:



*Consola*:

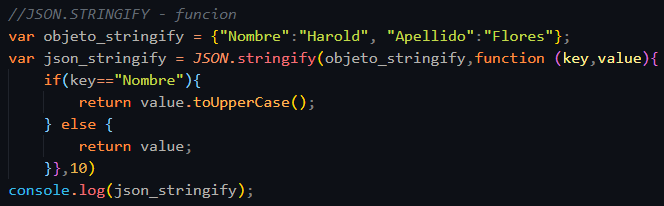


* JSON.stringify()

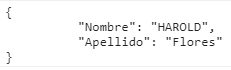
Transforma un objeto javascript en un string. Esta función recibe tres parámetros: el objeto, una función o array y los espacios antes de cada elemento. La función es opcional al igual que los espacios.

* + Con función como segundo parámetro.

*Script*:

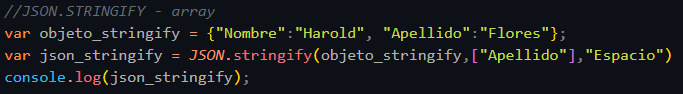


*Consola:*



* + Con array como segundo parámetro.

*Script*:



*Consola:*



1. Tiempo y fechas

* New Date()

Obtiene la fecha y la hora actual.

*Script*:



*Consola*:



* getFullYear()

Obtiene el año actual.

*Script:*



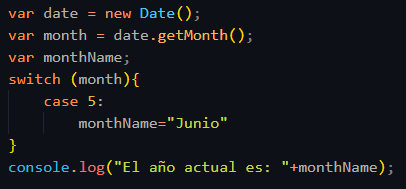
*Consola*:



* getMonth()

Devuelve el número del mes, tomando en cuenta que comienza en cero y termina en once.

*Script*:



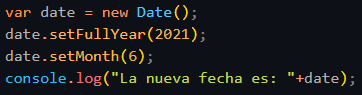
*Consola*:



* setFullYear() - setMonth()

Cambiar los valores de la fecha actual a cualquiera que se elija.

*Script*:



*Consola*:

****

1. Métodos matemáticos

* Math.random()

Te devuelve un número aleatorio entre 0 y 1.

*Script*:

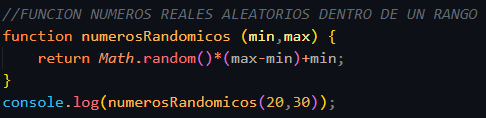


*Consola*:



* + Función para devolver un número real aleatorio dentro de un rango.

*Script*:

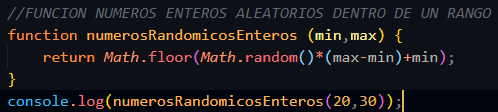


*Consola*:



* + Función para devolver un numero entero dentro de un rango.

*Script*:



*Consola*:



* Math.max()

Devuelve el valor más alto entre un conjunto de valores.

*Script*:

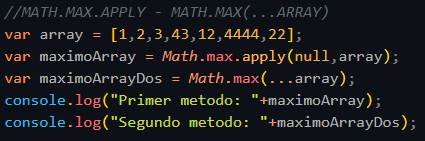


*Consola*:



* + Para devolver el valor máximo de un array podemos hacerlo de dos formas:

*Script*:



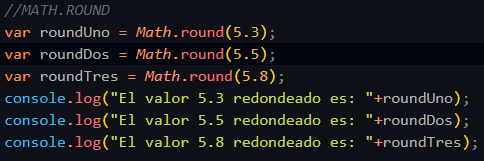
*Consola*:



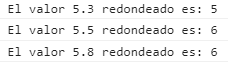
* Math.round()

Devuelve el valor entero más cercano al número real solicitado.

*Script*:



*Consola*:

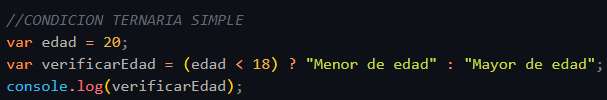


1. Condición Ternaria

Esta operación contiene tres operandos en el cual: el primero es una condición (true o false), el segundo es la sentencia a ejecutar si la condición es verdadera, el tercero es la sentencia a ejecutar si la condición es falsa.

* + Condición ternaria simple: Contiene solo una condición.

*Script*:

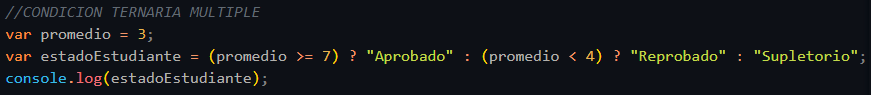


*Consola*:



* + Condición ternaria múltiple: Contiene dos o más condiciones.

*Script*:



*Consola*:

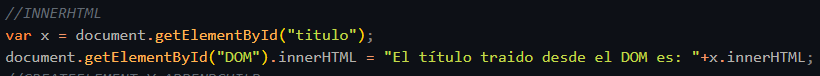


1. HTML DOM (Document Object Model)

Con el modelo de objetos, JavaScript permite crear HTML dinámico:

* Métodos:
  + getElementById(*id*): Encuentra un elemento por su id**.**
  + getElementsByTagName(*Tag\_name*): Encuentra un elemento por su etiqueta.
  + getElementsByClassName(*Class\_name*): Encuentra un elemento por su clase.
* Cambiar elementos HTML:
  + Elemento.innerHTML: Cambia el contenido del elemento HTML buscado.
  + Elemento.attribute: Cambia el atributo del elemento HTML buscado.
  + Elemento.style*.property:* Cambia el estilo del elemento HTML buscado.
  + Elemento.setAttribute(Atributo,valor): Cambia el valor de un atributo del elemento HTML buscado.
* Agregar y eliminar elementos HTML:
  + Document.createElement(Elemento): Crea un elemento HTML.
  + Document.removeChild(Elemento): Elimina un elemento HTML.
  + Document.appendChild(Elemento): Añade un elemento HTML.
  + Document.replaceChild(nuevo,antiguo): Reemplaza un elemento HTML.
  + Document.write(texto): Escribe directamente en el HTML.
* Algunos ejemplos:
  + **getElementById() con innerHTML**

*Script***:**



*Html*:

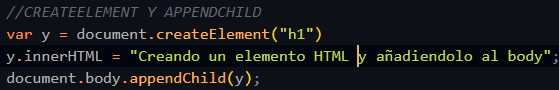


*Navegador*:



* + **createElement() y appendChild()**

*Script:*

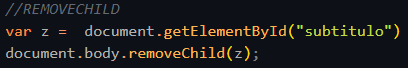


*Navegador*:



* + **removeChild()**

*Script:*

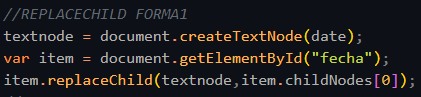


*Html:*



* + **replaceChild()**

*Script:*



*Html*:



*Navegador:*

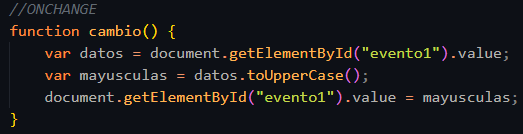


1. Eventos

* onchange()

Se ejecuta cuando el valor de una etiqueta HTML cambia.

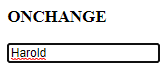
*Script*:

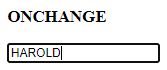


*Html*:



*Navegador*:

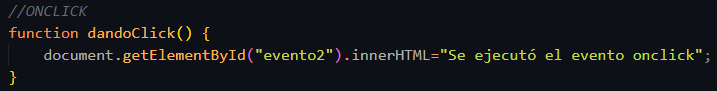




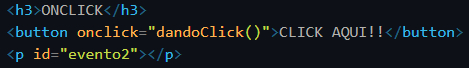
* onclick()

Se ejecuta cuando se da click sobre alguna etiqueta HTML.

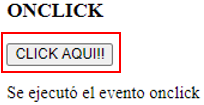
*Script*:



*Html*:



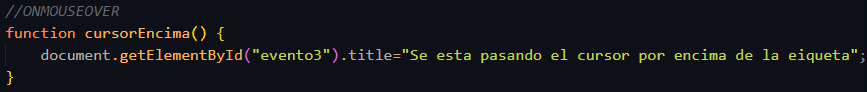
*Navegador*:



* onmouseover()

Se ejecuta cuando se pasa el cursor del mouse por la etiqueta HTML seleccionada.

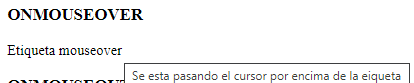
*Script*:



*Html*:



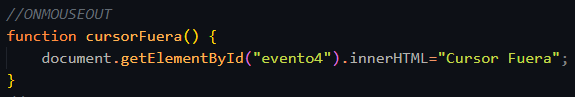
*Navegador*:



* onmouseout()

Se ejecuta cuando el cursor del mouse sale de la etiqueta HTML.

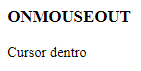
*Script*:



*Html*:



*Navegador*:





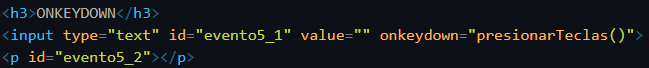
* onkeydown()

Se ejecuta al presionarse una tecla dentro de la etiqueta HTML seleccionada.

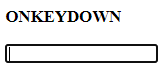
*Script*:

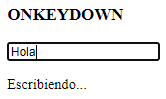


*Html*:



*Navegador*:

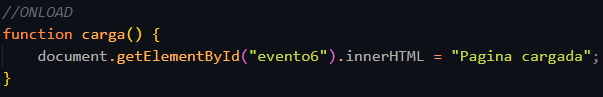




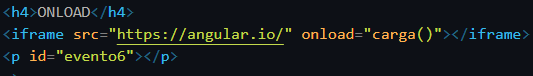
* onload()

Se ejecuta cuando se termina de cargar la etiqueta HTML seleccionada.

*Script*:



*Html*:



*Navegador*:

