

JavaScript. Guía de Referencia.

2.1.- Palabras reservadas.

break, case, catch, continue, default, delete, do, else, finally, for, function, if, in, instanceof, new, return, switch, this, throw, try, typeof, var, void, while, with.

2.2.- Variables.

[var] nombreVariable [=valor]

Secuencias de escape.

\n	Nueva línea
\t	Tabulador
\'	Comilla simple
\"	Comilla doble
\\	Barra inclinada

2.2.1.- Arrays

- o Conjunto de valores separados por comas y encerrados entre corchetes.

2.3.- Operadores.

- De asignación =
 - o Almacena en una variable el resultado de evaluar una expresión

- Lógicos

Operador	Significado
!	Not
&&	And
	Or

- Aritméticos

Operador	Significado
+	Suma
-	Resta
*	Multiplicación
/	División
%	Resto de una división entera
++	Incremento
--	Decremento

- De comparación

==	Igual a
!=	Distinto de
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que

- De concatenación +

- o Este operador permite concatenar cadenas de caracteres.

2.4.- Mensajes.

- `alert(texto_del_mensaje);`
o Visualiza una ventana con un mensaje que muestra información al usuario.
- `prompt(texto_del_mensaje,valor_por_defecto);`
o Visualiza una ventana que permite al usuario la introducción de datos.
- `confirm(texto_del_mensaje);`
o Visualiza un mensaje junto con 2 botones: Aceptar y Cancelar.
 - El primer botón devuelve al script un valor `true` y el segundo un valor `false`.

2.5.- Estructuras de control del flujo de ejecución.

2.5.1.- Sentencia `if`

```
if (expresión lógica){
    bloque de sentencias 1;
}
else{
    bloque de sentencias 2;
};
```

2.5.2.- Sentencia `while`.

```
while(expresión lógica)
do{
    bloque de sentencias;
};
```

2.5.3.- Sentencia `do...while`.

```
do{
    bloque de sentencias;
}while(expresión lógica);
```

2.5.4.- Sentencia `for`.

```
for(valor_inicial;expresión_lógica;sentencia_de_actualización)
{
    bloque de sentencias;
};
```

2.6.- JavaScript orientado a objetos.

objeto = `new` Clase();

2.6.1.- La clase **String**.

- **length** Propiedad que contiene la longitud de una cadena de texto.
- **charAt(posicion)** Aplicado a una cadena de texto, obtiene el carácter que se encuentra en la posición indicada.
 - o La primera posición es 0.
- **concat()** Equivale al operador **+**
- **indexOf("carácter")** Calcula la primera posición en la que se encuentra el carácter indicado dentro de la cadena de texto. Si la cadena no contiene el carácter, el método devuelve el valor **-1**.
- **lastIndexOf("carácter")** Ídem la última posición.
- **toLowerCase()** Transforma todos los caracteres de la cadena a minúsculas:
- **toUpperCase()** Ídem a mayúsculas.
- **substring(inicio, final)** Extrae una porción de una cadena de texto. El segundo parámetro es opcional. Si sólo se indica el parámetro *inicio*, la función devuelve la parte de la cadena original correspondiente desde esa posición hasta el final:

2.6.2.- La clase **Math**.

- **NaN** Valor devuelto por JavaScript cuando obtiene un valor no numérico o no definido (*Not a Number*).
 - o **toFixed(dígitos)**, Método que devuelve el número original con tantos decimales como los indicados por el parámetro **dígitos** realizando los redondeos necesarios.

2.6.3.- La clase **Array**.

- **length** Propiedad que contiene el número de componentes de un array.
- **pop()** Devuelve el último elemento del array y lo elimina. El array original se modifica y su longitud disminuye en 1 elemento.
- **push(elemento)**, Añade un elemento al final del array. El array original se modifica y su longitud aumenta en 1 elemento.
- **reverse()**, Modifica un array colocando sus elementos en el orden inverso a su posición original:
- **shift()** Devuelve el primer elemento del array y lo elimina. El array original se modifica y su longitud disminuye en 1 elemento.
- **unshift(elemento)** Añade un elemento al principio del array. El array original se modifica y aumenta su longitud en 1 elemento.
- **split("carácter")** Es un método de la clase **String**. Convierte una cadena en un array de cadenas separando la cadena inicial a partir del carácter especificado.
 - o Este método puede utilizarse para convertir una palabra en un array de caracteres.

2.6.4.- La clase **Date**.

- **getDate()** Devuelve el día del mes.
- **getDay()** Devuelve el día de la semana en forma de número.

- o El primer día de la semana (domingo) es el 0.
- `getFullYear()` Devuelve el año en forma de número con 4 cifras.
- `getHours()` Devuelve la hora.
- `getMinutes()` Devuelve el minuto.
- `getMonth()`. Devuelve el mes en forma de número.
 - o El primer mes (enero) es el 0.
- `getSeconds()` Devuelve el segundo.
- `setDate(valor)`, `setFullYear(valor)`, `setHours(valor)`, `setMinutes(valor)`, `setMonth(valor)`, `setSeconds(valor)` Permiten almacenar en el objeto sobre el que se ejecutan el día del mes, el año, etc.

2.7.1- Funciones Predefinidas.

- `parseInt(valor)`. Devuelve la parte entera de un número sin redondear.
 - o Si el primer carácter de la cadena es no numérico devuelve `NaN`.
- `parseFloat(valor)`. Análogo a `parseInt()` para números decimales.
- `isNaN(expresión)` Devuelve `true` si el resultado de evaluar la expresión es `NaN` y `false` en caso contrario.

3.1.- Algunos eventos.

- `onclick`: Evento que se produce cuando se hace click con el ratón sobre un elemento.
- `onsubmit`: Evento que se produce cuando se va a enviar un formulario.

4.2.- Acceso a los controles de un formulario.

- Objeto `document`. Creado por el navegador cuando carga una página web. A través de él puede accederse a cualquiera de las etiquetas que forman la página.
- Array `options`. Array que crea el navegador por cada control `<select>`. Contiene una referencia a cada uno de los controles `<option>` que lo forman.
- Propiedad `checked`, aplicada a controles de `"radio"` y `"checkbox"` devuelve `true` si el control ha sido seleccionado y `false` en caso contrario.
- Propiedad `selectedIndex`, aplicada a controles `<select>`, permite obtener el índice de la opción seleccionada. El primer índice es 0.
- Propiedad `text`, aplicada a controles `"option"`, permite obtener su contenido.
- Propiedad `value`, aplicada a controles `"text"` y `"textarea"`, contiene el dato tecleado. Aplicada a controles `"radio"`, `"checkbox"` y `"option"`, contiene el valor de su atributo `value`.
- Método `getElementById("valor")`, perteneciente al objeto `document`, obtiene una referencia al control cuyo valor de atributo `id` coincida con el valor especificado.
- Método `getElementsByTagName("valor")`, perteneciente al objeto `document`, obtiene un array con una referencia a cada uno de los controles cuyo valor de atributo `name` coincida con el especificado.