



JavaScript



Módulo 8



1. Funciones estándar del lenguaje.
2. Funciones personalizadas: tipología, etapas, argumentos y retorno.
3. Modo de función estricto: “Strict mode”.



Funciones estándar de JavaScript.

JavaScript tiene integradas varias funciones de nivel superior:

La función `eval()` si guardamos código en una cadena y se la pasamos ejecutará el código dentro de la cadena.

La función global `isFinite()` determina si el valor pasado es un número finito. Si es necesario, el parámetro, primero se convierte en un número.



Funciones estándar de JavaScript.

JavaScript tiene integradas varias funciones de nivel superior:

La función `isNaN()` determina si un valor es NaN (not is a number) o no.

La función `parseFloat()` procesa un argumento de cadena y devuelve un número de punto flotante.

La función `parseInt()` procesa un argumento de cadena y devuelve un número entero de la base especificada (la base en los sistemas numéricos matemáticos).



Funciones estándar de JavaScript.

El objeto **Date** nos permite trabajar con fechas:

```
7  
8  const fecha = new Date();  
9
```

Así creamos un nuevo objeto de **Date** con la fecha y hora actuales.



Métodos de obtención de fecha.

Method	Description
<code>getFullYear()</code>	Obtiene el año a 4 dígitos (aaaa)
<code>getMonth()</code>	Obtiene el mes como un número (0-11)
<code>getDate()</code>	Obtiene el día como un número (1-31)
<code>getHours()</code>	Obtiene la hora . (0-23)
<code>getMinutes()</code>	Obtiene el minuto (0-59)
<code>getSeconds()</code>	Obtiene los segundos (0-59)
<code>getMilliseconds()</code>	Obtiene los milisegundos (0-999)
<code>getTime()</code>	Obtiene el tiempo (milisegundos desde el 1ro de enero de 1970)
<code>getDay()</code>	Obtiene la semana como un número (0-6)
<code>Date.now()</code>	Obtiene el tiempo . ECMAScript 5.



Ejercicio.

1. Crear un formato de tiempo así:

Día: 17 Mes: 05 Año: 2022

Y son las: 11:45 horas.

Imprimirlo en la ventana del navegador.



Funciones personalizadas en JavaScript.

Las funciones de JavaScript se definen con la palabra reservada `function`.

Puede utilizar una declaración de función o una expresión de función.

```
15 function nombreFuncion(parametros) {  
16     // Código que será ejecutado  
17     // cada que mandemos a llamar  
18     //la función.  
19 }
```

Veamos algunos ejemplos.

(Incluso mandemos a imprimir la estructura de una función.)



Ejercicios.

1. Crear una función que dado el peso y la altura de una persona imprima en la ventana del navegador su IMC.
2. Hacer una función que reciba un monto y si este excede los \$1000 se aplica un descuento de 7%, imprimir el monto final en la ventana del navegador.



Expresiones de función.

Una función de JavaScript también se puede definir usando una expresión .

Una expresión de función se puede almacenar en una variable:

```
21  
22  var x = function(a){ return a*a};  
23
```

La función anterior es en realidad una función anónima (una función sin nombre).
Veamos algunos ejemplos.



Ejercicio.

1. Crear una expresión de función que reciba una cantidad de segundos y la transforme a su equivalente en minutos con segundos. Ejemplo `t(215)` e imprime: Minutos 3, Segundos: 35.



Funciones autoinvocadas.

Una función se puede ser llamada en automático sin necesidad de escribir la llamada.

```
26  (function () {  
27      let x = "Hola!!"; // Este mensaje sale en automatico.  
28      document.write("<br> --> "+x);  
29  })();
```

Probemos el ejemplo.



Ejercicio.

1. Crear una función autoinvocada que imprima un mensaje deseando buen día, con el día exacto en texto, ejemplo: “Excelente martes para ti”.



Strict mode.

`"use strict";` Define que el código JavaScript debe ejecutarse en "modo estricto".

¿Para que sirve?

El modo estricto cambia la "mala sintaxis" previamente aceptada en errores reales.

Como ejemplo, en JavaScript normal, escribir mal el nombre de una variable crea una nueva variable global. En modo estricto, esto generará un error, lo que imposibilitará la creación accidental de una variable global.

Hagamos un ejemplo con una función.