

«Talento Tech»

Desarrollo Web 2

Clase 02





Clase N° 2 | Primeros pasos en JS

Temario:

- Funciones nativas
- Primeros pasos en JavaScript
- Variables y constantes
- Concatenación de variables
- Sumas y restas



Funciones Nativas

JavaScript como lenguaje de programación nos presenta una **serie de funciones propias** como vimos la clase pasada, nos será de mucha utilidad para agregarle dinamismo y funcionalidad a nuestro sitio web. A partir de la clase de hoy comenzaremos a ver cómo lo lograremos.

Comenzaremos a trabajar desde la consola de nuestro navegador, y a partir de allí insertamos el código, tal como vimos la clase pasada.

Funciones nativas de JavaScript

Son funciones ya definidas, y cada función nativa se usa para un fin distinto.

Estas son algunas de las más comunes:

- **alert()**: Función nativa diseñada para que, al escribir un mensaje dentro del paréntesis, ese mensaje se muestre en formato de cuadro de diálogo en nuestra página web.

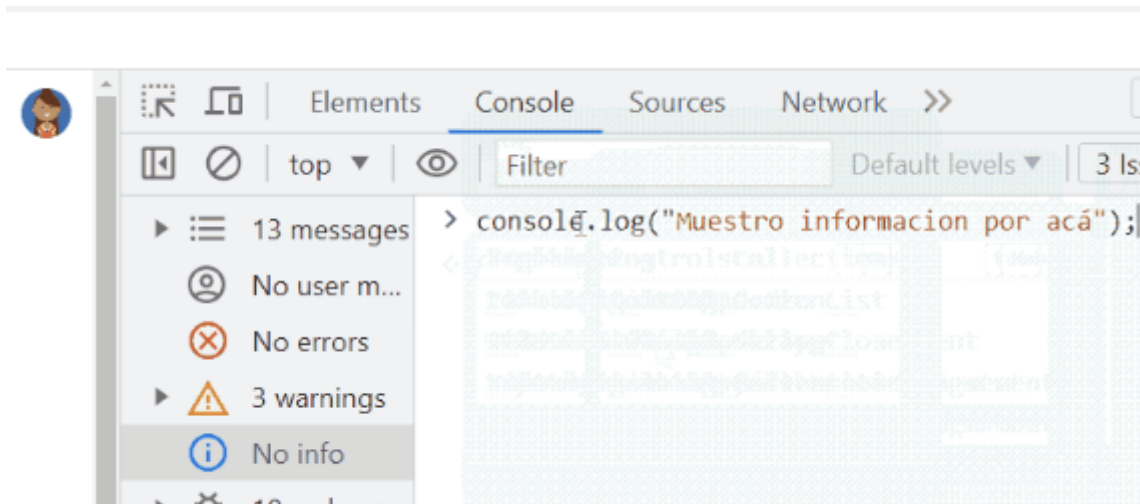
Ejemplo:

```
alert("Esta es mi primera alerta!");
```

- **console.log()**: Función nativa útil para imprimir un mensaje en la consola de nuestro navegador.

Ejemplo:

```
console.log("Hola mundo");
```



→ **prompt()**: Función nativa utilizada para hacerle una pregunta al usuario, posee un campo donde éste puede escribir la respuesta. dicha respuesta suele almacenarse en una variable para poder trabajarla luego

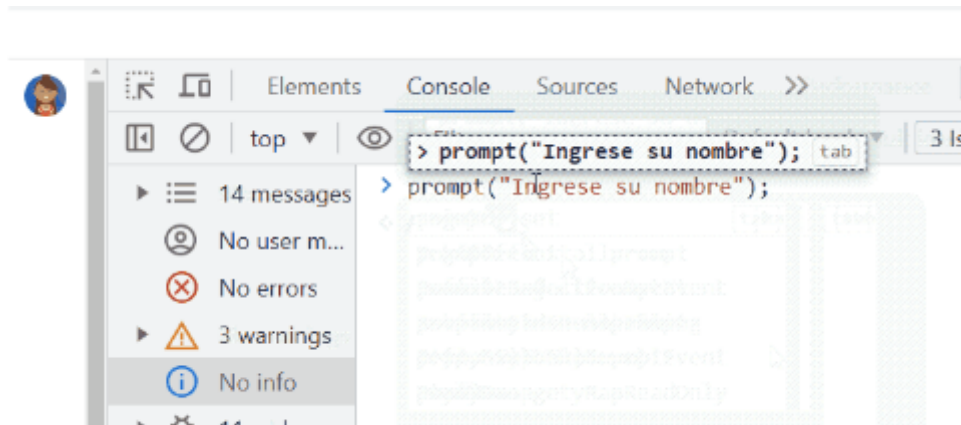
Ejemplos:

```
prompt("¿Cuál es tu nombre de pila?");
```

Aquí, la respuesta del usuario no se almacena en ningún lugar.

```
let nombre = prompt("¿Cuál es tu nombre de pila?");
```

Por el contrario, en este caso la respuesta del usuario queda almacenada en la variable llamada "nombre".



Primeros pasos en JavaScript

En **Javascript** cuando hablamos de **datos** nos referimos a los valores que pueden guardarse y utilizarse en la ejecución de un programa.

Para almacenarlos utilizamos variables, que funcionan a modo de cajas, es decir nos permiten guardarlos y utilizarlos cuando lo deseemos. Estos datos pueden ser de diferentes tipos y se los utiliza para representar información.

Tipos de Datos

En Programación, existen diferentes tipos de datos y los principales son:

Number	1 2 3 -5 -9,7	Número
Strings	Hola, Juan, El uso de ""	Texto
Boolean	True-False	Verdadero o Falso
Undefined	(Aún no está definido)	Indefinido
Null	Valor Nulo	Nulo

Para cada tipo de dato se determina que tipo de operación se pueden realizar:

Number	Suma, resta, multiplicación, división, módulo
Strings	concatenación, métodos
Float	Suma, resta, multiplicación, división, módulo
Boolean	Evaluación verdadero-falso
Undefined	Indefinido
Null	Nulo

Undefined: es que a una variable no tiene asignado ningún valor. Es como tener una caja vacía.

Null: pertenece a un tipo de dato particular que se llama objetos (ya vamos a ver qué significa esto) y especifica que algo no tiene un valor, y es distinto a undefined.

Variables y constantes

Para comprender estos conceptos avanzaremos con un ejemplo:

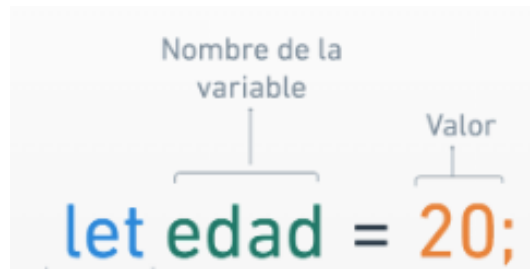
Si en una caja insertamos inicialmente el valor "Hola", y al rato reemplazamos este valor por el valor "chau", el contenido de la caja **varía**, por lo tanto, la llamamos variable.





La palabra reservada para asignar una variable es "**let**".

Por ejemplo:



Al decir "let edad", estamos creando una **Variable** que se llama "edad".

-¿Qué es aquello que voy a insertar en la caja "edad"?

-Voy a insertar el número 20.

En cambio, si el contenido de la caja no puede modificarse nunca, y solamente hay un dato que queda estático dentro de ella, el contenido de la caja queda **Constante** y le asignamos ese nombre.

La palabra reservada para asignar una constante es "**const**".

const pi = 3,14159



Creación de una caja o variable constante llamada "pi".

A dicha caja, le inserto el valor "3,14159"

Concatenación de variables

La concatenación de Variables es una acción fundamental para combinar o unir variables de tipo string (texto), esto nos permite formar una cadena más grande o combinar un **String** con Números para mostrar resultados permitiendo unir diferentes valores.

! Importante: Para guardar texto en una Variable debemos colocarles **comillas**.

```
nombre = "Juan";  
apellido = "Perez Gonzalez";
```

En **JavaScript**, hay diferentes formas de concatenar variables para formar strings más grandes. Aquí hay algunas formas comunes de hacerlo:

- **Operador "+":**

Se utiliza **para concatenar strings** en JavaScript.

```
let nombre = "Juan";  
let apellido = "Perez Gonzalez";  
console.log(nombre + " " + apellido);
```

Resultado:

Juan Perez Gonzalex

- **Plantillas de cadenas (template literals):**

Se colocan directamente las variables dentro de la cadena utilizando `${}`. Esto hace que el código sea más claro y fácil de leer. Para que esto sea posible es necesario hacer uso de comillas invertidas ``.

```
let nombre = "Jimena";  
let apellido = "Gonzalez";  
console.log(`Bienvenidos: ${nombre} ${apellido}`)
```


Otro ejemplo:

```
let nombre = prompt ("Inserte su nombre");  
let apellido = prompt ("Inserte su apellido");  
let nombreCompleto = nombre + " " + apellido;  
console.log("Su nombre completo es: " + nombreCompleto)
```

Resultado:

Su nombre completo es: Marcelo Perez

Operaciones de suma y resta con JavaScript

Ahora que ya conocemos las funciones nativas de **JavaScript** y tuvimos una introducción acerca de variables y constantes, es momento de relacionar todo lo que vimos en esta clase N° 2.

Ejemplo 1:

```
let resultado = 2 + 3;  
console.log("El resultado de la suma es: " + resultado);
```

Resultado:

El resultado de la suma es: 5

Ejemplo 2:

```
let primerNumero = 3;  
let segundoNumero = 1;  
let resta = primerNumero - segundoNumero;  
console.log("El resultado de la resta es: " + resta);
```

Resultado:

El resultado de la resta es: 2



? Pregunta de reflexión o de llamado de atención

En esta clase vimos los tipos de datos que se utilizan en **JavaScript**.

¿En qué caso se te ocurre que conviene utilizar una constante (const) o una variable (let)?

Considerando la concatenación de variables en **JavaScript**, ¿puedes mencionar una situación práctica en la que sería útil utilizar la concatenación para formar cadenas de texto más grandes en tu página web?



Buenos Aires
aprende
Agencia de Actividades para el Futuro

BA Buenos
Aires
Ciudad