



JavaScript

Introducción a JavaScript

***Construir una página web
utilizando scripts en
Javascript y la biblioteca
jQuery para manipular el
comportamiento de la
interfaz de usuario y agregar
funcionalidades***

{desafío}
latam_

- Unidad 1: HTML y CSS
- Unidad 2: Bootstrap
- Unidad 3: JavaScript
- Unidad 4: Terminal, Git, GitHub y GitHub Pages



Te encuentras aquí



¿Qué aprenderás en esta sesión?

- *Implementa scripts locales y externos en un documento HTML, para integrar JavaScript a un proyecto web.*
- *Emplea declaraciones, valores, expresiones y operadores, de acuerdo a la sintaxis de JavaScript.*

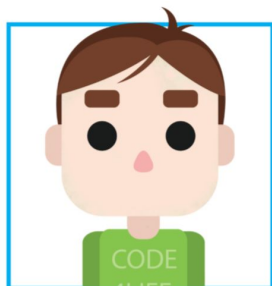
`/* ¿Qué es JavaScript? */`

¿Qué es JavaScript?

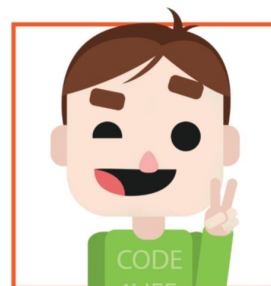
- JavaScript es un lenguaje mediante el cual se crean documentos que contienen instrucciones (scripts), que se ejecutan cuando se carga una página web (al leer la página) o cuando se produce un suceso determinado (eventos, como hacer click sobre un vínculo).



HTML
(Estructura)



CSS
(Apariencia)



JAVASCRIPT
(Animación)

¿Qué es JavaScript?

Algunos ejemplos de JavaScript



Fuente: [TheFWA](#).



Fuente: [Promoflex](#).



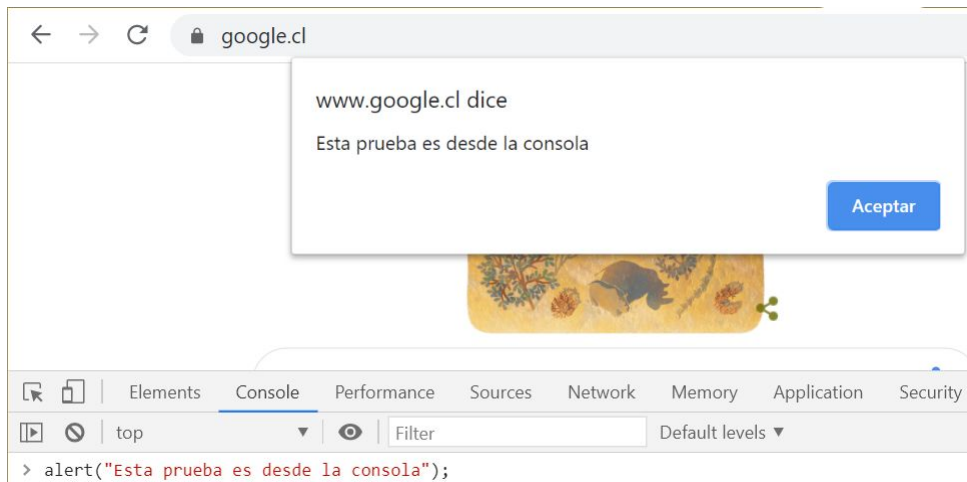
Fuente: [Pinterest](#).

`/* Agregando JS */`

Agregando JS

Inspector de elementos

- En la pestaña **console**, podremos ejecutar código JavaScript. Por ejemplo, al ejecutar el código: `alert("Esta prueba es desde la consola");` se mostrará lo siguiente:



Agregando JS

Desde el mismo archivo HTML

- Para hacer esto, una de las opciones que tenemos es incluir el código en una etiqueta `<script>` ... `<script>`. Ésta podemos colocarla tanto dentro de la etiqueta `<head>` como de la etiqueta `<body>`.

```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Probando JS</title>
  </head>
  <body>

    <script>
      alert("Esta prueba es desde la etiqueta
script");
    </script>
  </body>
</html>
```

Agregando JS

Desde un archivo JS

- Podemos escribir JavaScript en un archivo externo al HTML. Para vincularlo a la etiqueta `<script>` le debemos añadir el atributo **src** con valor de la ruta del archivo.

```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Probando JS</title>
  </head>
  <body>

    <script
src="assets/js/script.js"></script>
  </body>
</html>
```

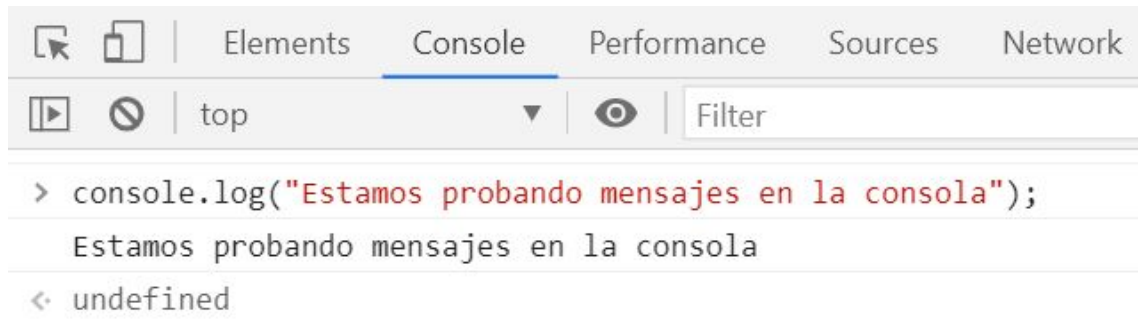
`/* Console log */`

Console log

- Existe otra función que aprenderemos, el llamado `console.log()`; Con él se puede imprimir lo que necesitemos, pero en la consola. Veamos un ejemplo:

```
console.log("Estamos probando mensajes en la consola");
```

Nos mostrará el siguiente mensaje:



/* Declaraciones y sintaxis */

Declaraciones y sintaxis

- Un programa es una lista de "instrucciones" para ser "ejecutadas" por un computador.
- En un lenguaje de programación, estas instrucciones de programación se denominan declaraciones.
- Un script de JavaScript es una lista de declaraciones.



Declaraciones y sintaxis

Declaraciones de JavaScript

- Las declaraciones de JavaScript se componen de:



Declaraciones y sintaxis

Sintaxis

- La sintaxis de JavaScript tiene sus propias reglas, pero dependerá del tipo de declaración que corresponda. Los punto y coma (;) sirven para separar las declaraciones de JavaScript. Si bien ahora ya no son 100% obligatorios, nos ayudarán a ordenar nuestro propio código y a evitar errores.
- Es importante tener en cuenta los siguientes puntos en JavaScript:
 - Es sensible a las mayúsculas y minúsculas;
 - No toma en cuenta los espacios en blanco ni los saltos de línea.

Declaraciones y sintaxis

Valor

- **Literales:** Como se escriba, dependerá del tipo de dato.
- **Variables:** Se emplea para almacenar y hacer referencia a un valor.

```
100
```

```
0.5
```

```
"Hola a todos"
```

```
let numero = 100;
```

```
let numero2 = 0.5;
```

```
let frase = "Hola a  
todos";
```

Declaraciones y sintaxis

Operadores y expresiones

- **Operadores:** Son símbolos que ocupamos para interactuar con las variables.
- Veremos 3 tipos:
 1. Operadores de asignación
 2. Operadores aritméticos
 3. Operadores de comparación
- **Expresiones:** Cualquier unidad de código válida que se resuelve en un valor. Es una combinación de valores, variables y operadores, que computa un valor. El cómputo se llama evaluación.

```
x = 7
```

```
3 + 4
```

Declaraciones y sintaxis

Palabras claves y comentarios

- **Palabras claves:** Son identificadores que tienen un significado especial en JavaScript. Las declaraciones de JavaScript a menudo comienzan con una palabra clave para identificar la acción que se realizará.
- Ejemplo:
 - Abstract
 - Arguments
 - Boolean
 - Continue

- **Comentarios:**

- No todas las declaraciones de JavaScript son "ejecutadas";
- El código después de // o entre /* y */ se trata como un comentario;
- Los comentarios son ignorados, y no serán ejecutados;
- Los comentarios son muy útiles para documentar el código.

Demostración "Primeros pasos con JavaScript"



Ejercicio guiado

Primeros pasos con JavaScript

- En el siguiente ejercicio, iremos experimentando con JavaScript, para ir teniendo mayor familiaridad con su declaración y sintaxis. De esta manera, iremos adquiriendo práctica, para agregar aspectos dinámicos a nuestros sitios web.
- Lo primero que haremos será crear un proyecto nuevo, donde realizaremos las pruebas pertinentes para conocer el lenguaje de JavaScript, paso a paso.
- Como ya lo hemos hecho durante el módulo crearemos un directorio de carpetas, con sus archivos ordenados y clasificados.



Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 1:** crea una nueva carpeta donde más nos acomode. Esta se llamará `probando-js`.
- **Paso 2:** entra a VSC. Crea el `index.html`, en la raíz de la carpeta `probando-js`.
- **Paso 3:** crea la carpeta `assets` presionando "botón derecho" y luego nueva carpeta.
- **Paso 4:** dentro de `assets` crea la carpeta `js`.



Hacer este ejercicio por cada proyecto ayudará a tener una estructura de carpetas ordenada,

Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 5:** ahora al index.html, dale la estructura base de un archivo HTML.

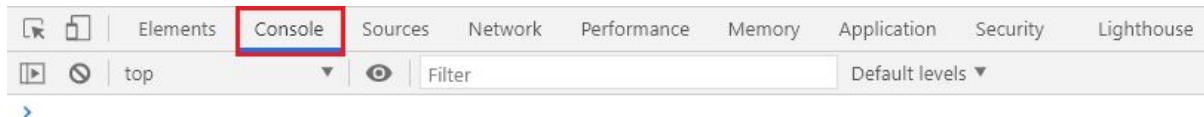
```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Probando JS</title>
  </head>
  <body>

  </body>
</html>
```

Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 6:** guarda y abre el archivo con el navegador Chrome. Por ahora, no mostrará nada, ya que no hay contenido, pero luego deberás ir refrescando el sitio con el código que agregues.
- **Paso 7:** ahora, te invitamos a experimentar JavaScript con la consola del navegador, ingresa al inspector de elementos y dirígete a la pestaña “Console”.



Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 8:** escribe en la consola lo siguiente `alert("Estoy ejecutando la función alert");` presiona enter para ejecutarlo. Podremos observar que nos aparece la alerta.
- **Paso 9:** Observa ahora cómo se comporta `console.log`. Escribe en la consola lo siguiente: `console.log("Estoy ejecutando la función console.log");` presiona enter para ejecutarlo. Podremos observar que aparece el mensaje, pero esta vez en nuestra consola:

Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 10:** realiza este mismo ejercicio, pero desde el archivo index.html. Dentro del tag `<body>` agrega dos líneas, una para la función `alert()` y otra para `console.log()`, como lo muestra el siguiente ejemplo:

```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Probando JS</title>
  </head>
  <body>
    <script>
      alert("Estoy ejecutando la función alert");
      console.log("Estoy ejecutando la función console.log");
    </script>
  </body>
</html>
```

Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 11:** Guarda y abre el archivo en el navegador. Observa que inmediatamente muestra la alerta.
- Una vez que haces clic en aceptar, verifica la consola. Observa que también está ahí el mensaje.
- **Paso 12:** ahora haz lo mismo, pero desde el archivo script.js. Elimina el código que acabas de agregar y agrega las referencias al archivo externo:

```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Probando JS</title>
  </head>
  <body>
    <script
src="assets/js/script.js"></script>
  </body>
</html>
```

Ejercicio guiado

Primeros pasos con JavaScript

- **Paso 13:** En el archivo script.js escribe el siguiente código:

```
alert("Estoy ejecutando la función  
alert");  
console.log("Estoy ejecutando la función  
console.log");
```

- Observa que no es necesario volver a agregar la etiqueta **<script>**, solo agregaremos aquí el código JavaScript que queramos ejecutar.
- Si guardas y vuelves a cargar el archivo, verás que mantiene el mismo comportamiento que la prueba anterior, pero ahora has llamado a un archivo como referencia desde el documento HTML.

Ejercicio "Código sencillo"



Ejercicio propuesto

- A continuación te mostramos el código JavaScript relativo a expresiones y operadores.
- En el siguiente capítulo podremos experimentar con mayor profundidad estos aspectos, a través de las variables y tipos de datos, pero por el momento, sigamos experimentando con la consola.
- Ejecuta los siguientes códigos (línea por línea) por consola:



Ejercicio propuesto

```
3 + 4  
x = 3 + 4  
x  
y = x  
y  
y = y+y  
y  
x  
z = "Hola"  
z = z + y  
console.log(z);  
alert(z);
```



Resumen

- La función `console.log()`; permite imprimir lo que necesitemos, pero en la consola.
- Una variable se emplea para almacenar y hacer referencia a un valor.
- Los operadores son símbolos con los que se pueden realizar operaciones con variables, valores literales o variables y valores literales.
- Una expresión es cualquier unidad de código válida que se resuelve en un valor.
- Las palabras clave son identificadores (token) que tienen un significado especial en JavaScript.
- Un comentario es el código después de `//` o entre `/ *` y `* /` el cual no es ejecutado.

¿Cuáles son las 3 formas de trabajar con JavaScript?





Próxima sesión...

- *Utilizar correctamente las variables y tipos de datos de JavaScript, para realizar operaciones de asignación, aritméticas y de comparación. Implementa adecuadamente la librería jQuery en un proyecto web, utilizando recursos locales y externos.*

{desafío}
latam_

*Academia de
talentos digitales*

