



Introducción a JavaScript

JavaScript para la web

Unidad	Clases (sincrónico)	Autoaprendizaje (asincrónico)	Tutoría (sincrónico)
Introducción a JavaScript	2 horas	Desde 6 horas	2 horas
Condiciones	2 horas	Desde 6 horas	2 horas
Funciones	2 horas	Desde 6 horas	2 horas
Arreglos y objetos	2 horas	Desde 6 horas	2 horas
Métodos de arreglos	2 horas	Desde 6 horas	2 horas
APIs	2 horas	Desde 6 horas	2 horas
<i>Receso</i>	<i>0 horas</i>	<i>0 horas</i>	<i>0 horas</i>



**Activen las cámaras los que puedan y
pasemos asistencia**

***Utiliza
Javascript
para obtener y
modificar
elementos del
DOM a partir
de un
requerimiento***

- Unidad 1: Introducción a JavaScript.
- Unidad 2: Condiciones.
- Unidad 3: Funciones.
- Unidad 4: Arreglos y objetos.
- Unidad 5: Métodos de arreglos.
- Unidad 6: APIs



Te encuentras aquí





Inicio

{desafío}
latam_



/ Ejecutar javascript en la consola del navegador */*

/ Crear un script que ejecute operaciones matemáticas */*

/ Manipular el DOM de una página web con JavaScript: */*

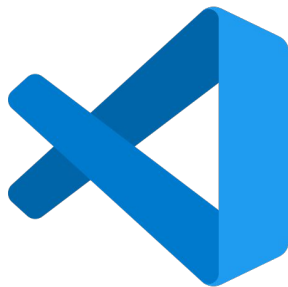
- Cambiar el contenido de un elemento
- Cambiar el estilo de un elemento

/ Cambiar el contenido o estilo en función del contenido ingresado en el prompt */*

Objetivos

Un editor de código

Podemos escribir JavaScript en muchos editores de código. En este curso utilizaremos el editor **VSCode**



Un navegador

JavaScript puede ser ejecutado en casi todos los navegadores. En este curso ocuparemos **Firefox**.





Desarrollo

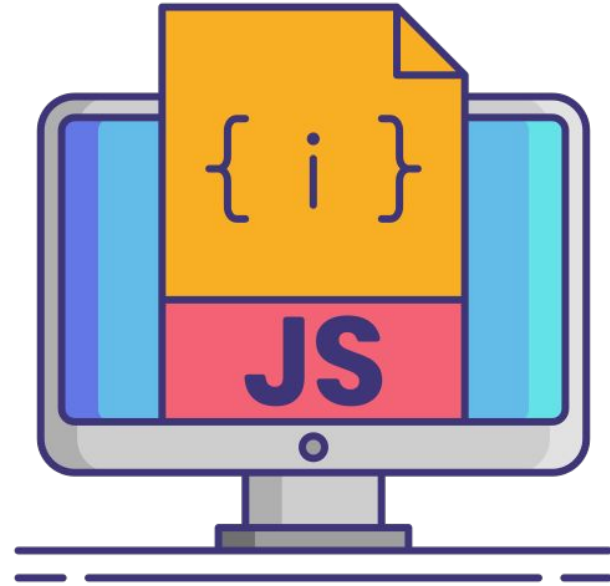
{desafío}
latam_



¿Qué es JavaScript?

JavaScript es un lenguaje de programación que nos permite agregar interacción a una página web

Mientras que HTML y CSS se encargan del esqueleto y la apariencia, JavaScript se enfoca en la lógica detrás de la interfaz, permitiendo que el usuario interactúe con el sitio web y estas reacciones.



Ejemplos JavaScript

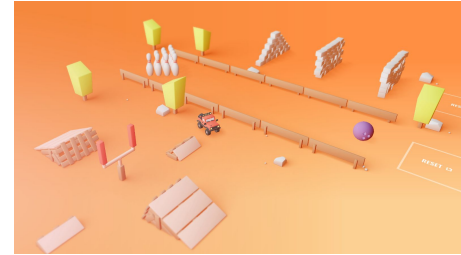
Para ver ejemplos de qué se puede hacer con JavaScript, visita los siguientes sitios web y observa cómo reaccionan con movimientos del mouse y el uso del teclado.



[Kuon Yagi](#)

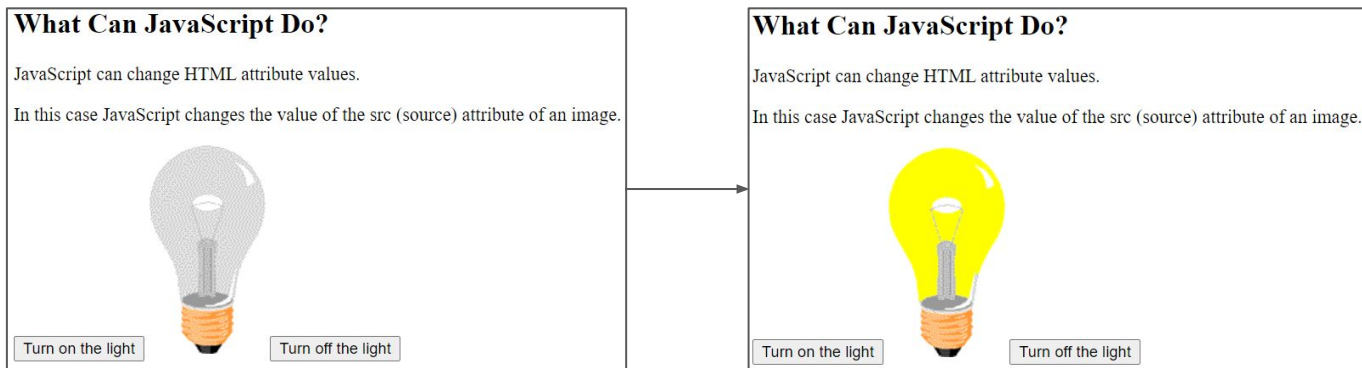
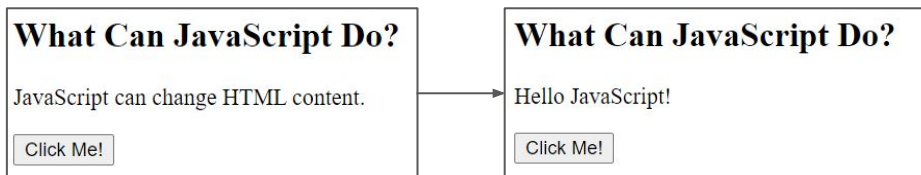


[Moon Farmer](#)



[Bruno Simon](#)

En [W3School](https://www.w3schools.com) contamos con una serie de ejemplos de lo que podemos hacer con JavaScript.



Mostrando el contenido

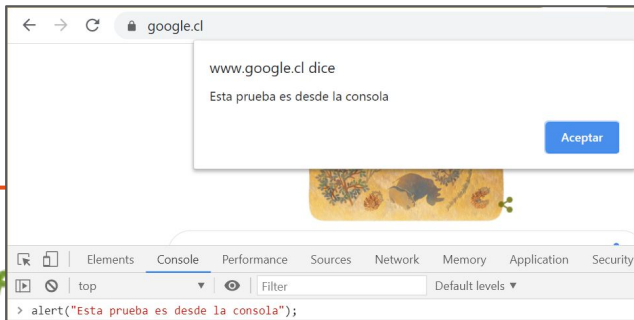
Con Alert

1. Abrir el inspector de elementos
2. Ir al tab console

Ahora podemos mostrar el contenido:

```
alert("Esta prueba es desde la consola");
```

Presiona Enter y verás lo siguiente:



Con console.log

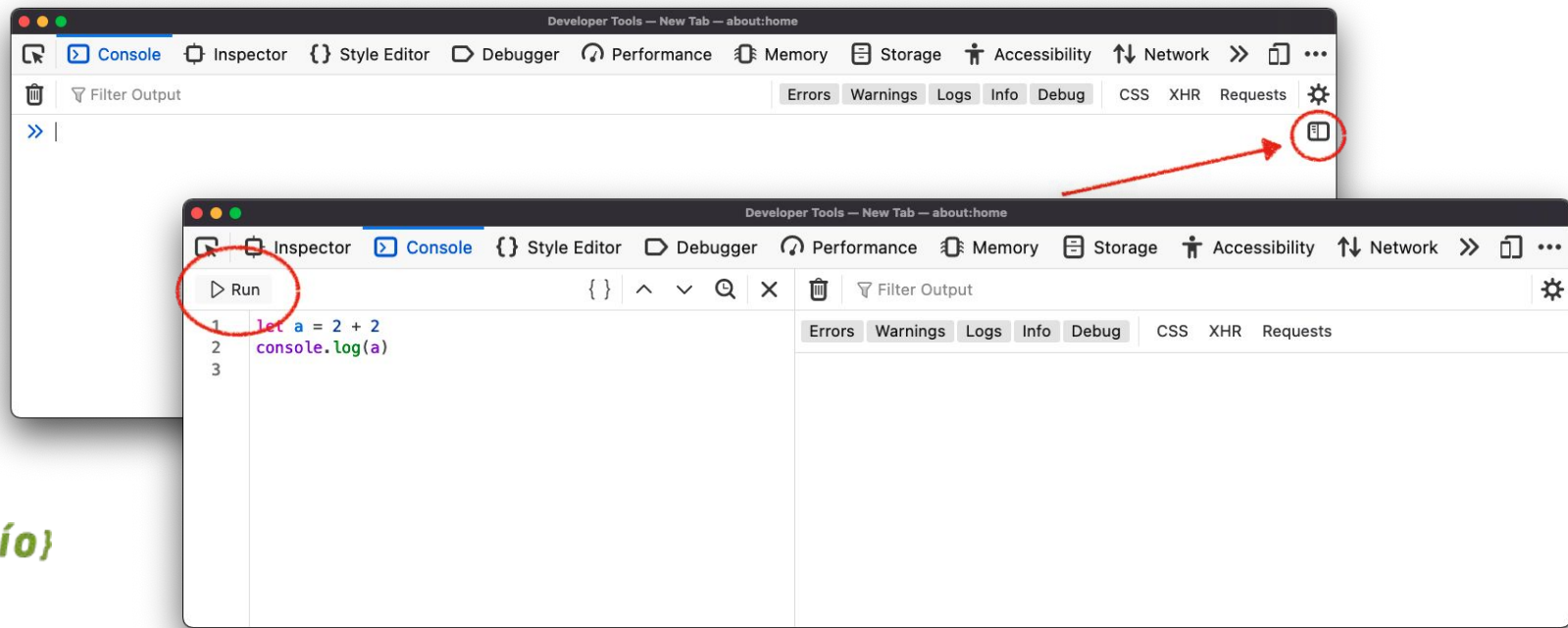
En la misma consola podemos probar:

```
console.log("Esta prueba es desde la consola");
```

Esto es muy útil para hacer pruebas, lo ocuparemos frecuentemente a lo largo del curso.

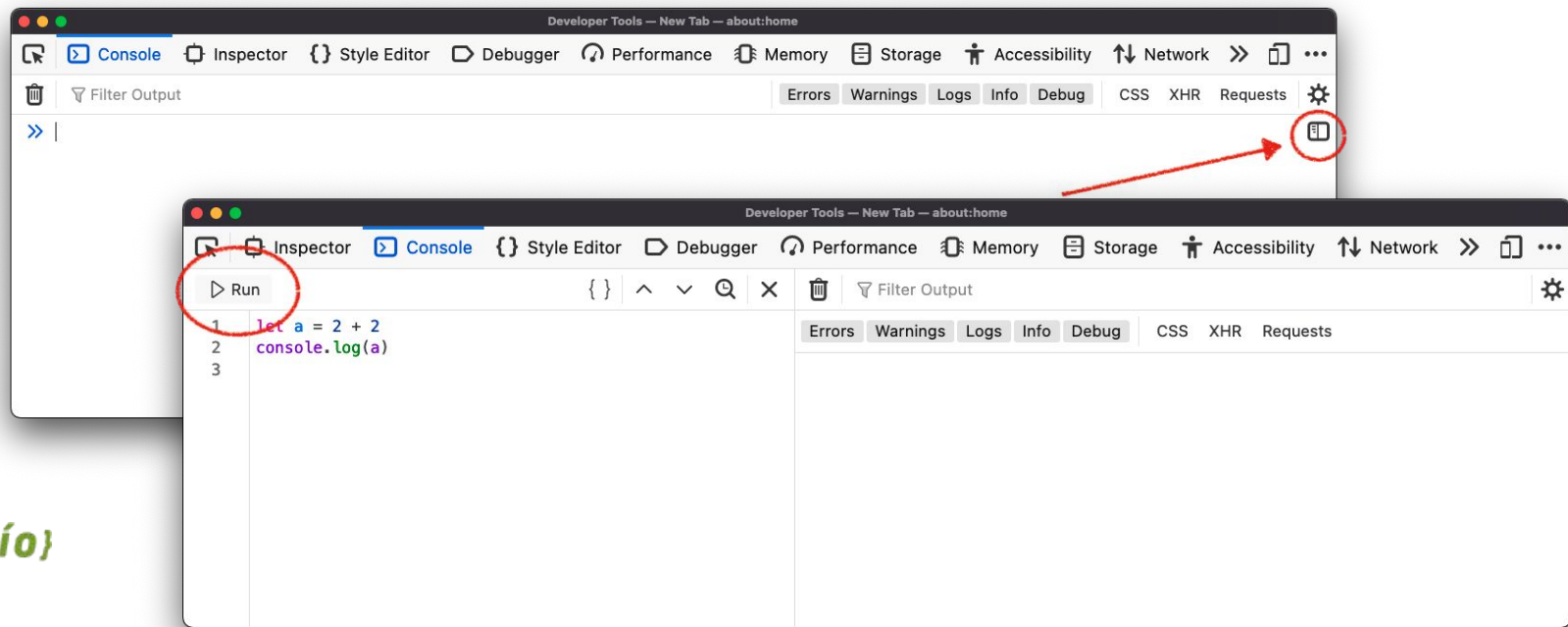
Activar modo multilinea de la consola

El modo multilinea de Firefox nos permite escribir múltiples líneas de código y luego ejecutarlas. Para activarlo debemos hacer clic en el siguiente ícono:



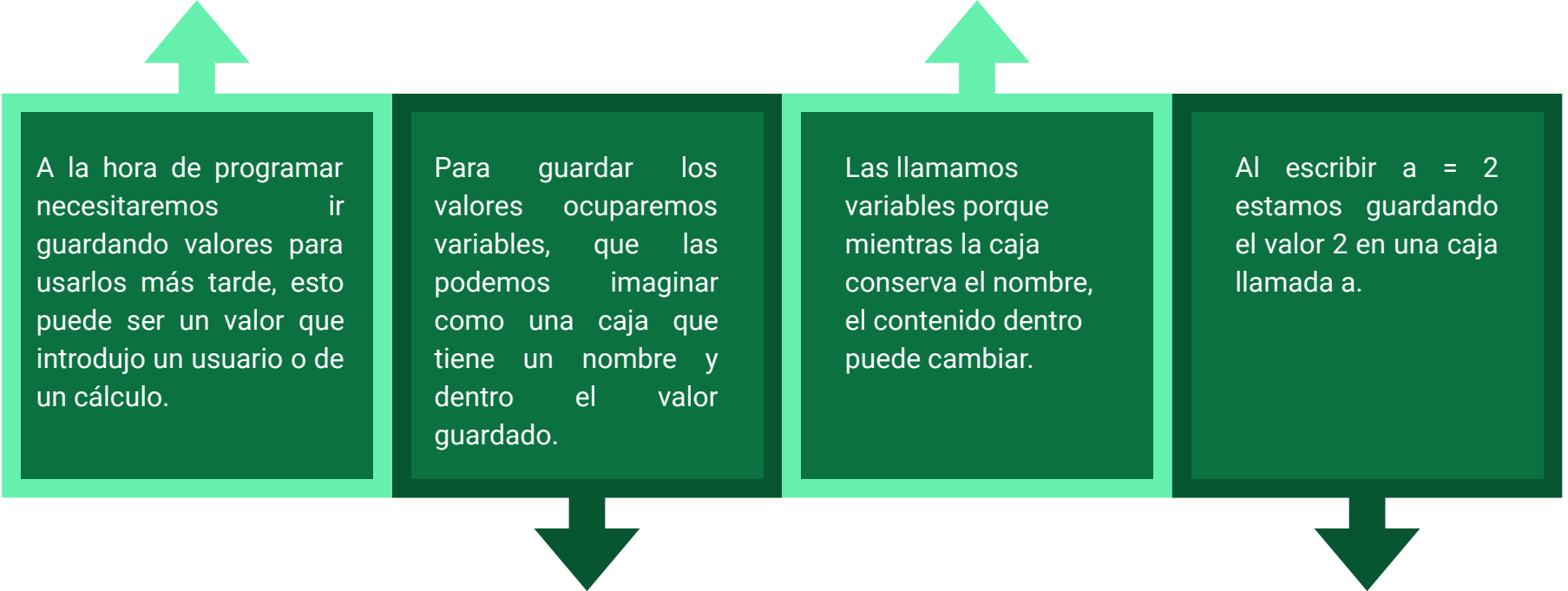
Ventaja del modo multilínea

Este modo se asemeja mucho más a la forma en que más adelante escribiremos nuestros programas en JavaScript. Por lo tanto, lo activaremos por ahora.



/* Variables */

Variables



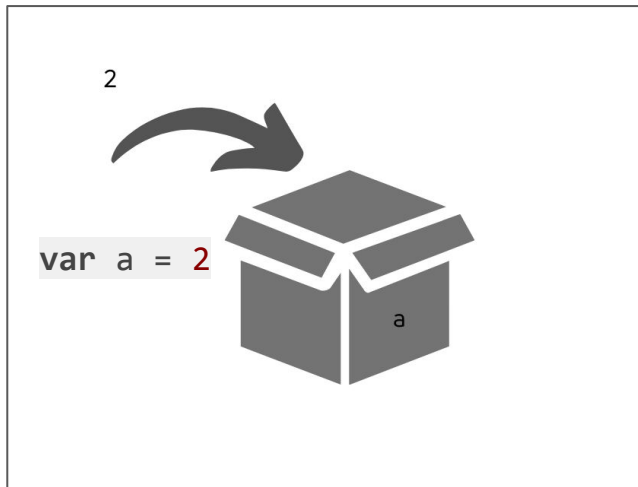
A la hora de programar necesitaremos ir guardando valores para usarlos más tarde, esto puede ser un valor que introdujo un usuario o de un cálculo.

Para guardar los valores ocuparemos variables, que las podemos imaginar como una caja que tiene un nombre y dentro el valor guardado.

Las llamamos variables porque mientras la caja conserva el nombre, el contenido dentro puede cambiar.

Al escribir `a = 2` estamos guardando el valor 2 en una caja llamada a.

Creando nuestra primera variable

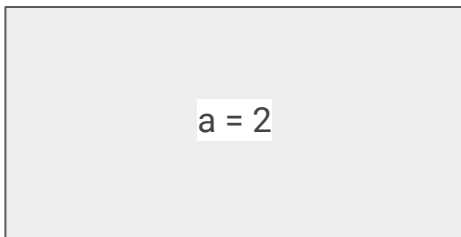


Probemos escribir en la consola del inspector de elementos:

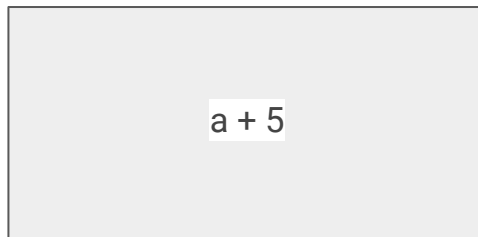
```
a = 2  
alert(a + 2)
```

Conceptos importantes sobre variables

Esto es **asignar** un valor a una variable

A light gray rectangular box with a thin black border. Inside the box, the text 'a = 2' is centered in a monospaced font.

Esto es **ocupar** una variable (para que funcione tiene que tener un valor asignado)

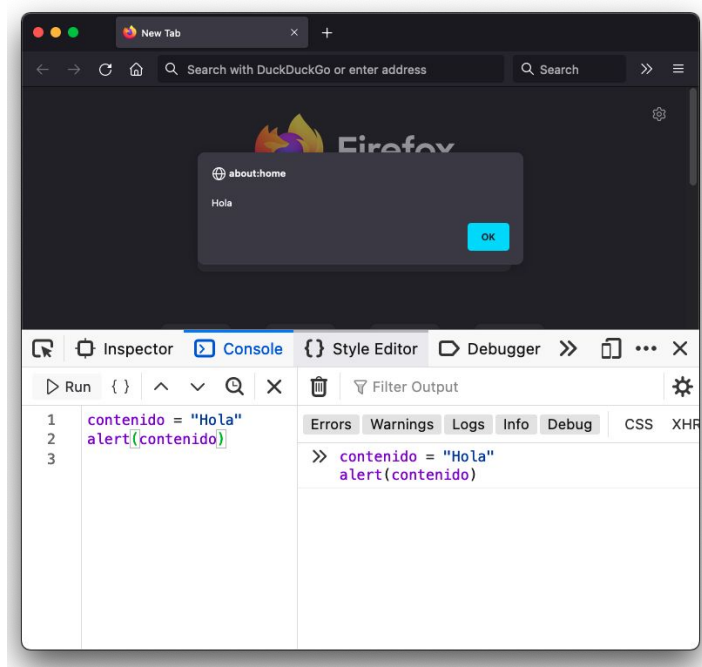
A light gray rectangular box with a thin black border. Inside the box, the text 'a + 5' is centered in a monospaced font.

Si ocupamos una variable sin definirla obtendremos un error.
En Javascript al asignar un valor a una variable, esta se define automáticamente.

Guardando texto en variables

Probemos escribir en la consola del inspector de elementos.

```
var contenido = "hola"  
alert(contenido)
```



Texto vs Variables

Cuando guardemos texto en una variable, necesitaremos envolverlo en comillas simples o dobles. Podemos usar cualquiera de las dos.

Cuando no utilizamos comillas, javascript interpreta ese texto como si fuera el nombre de una variable o de una instrucción de JavaScript.

Probemos escribir en la consola del inspector de elementos.

Sintaxis

Así como en el lenguaje existe la gramática, los lenguajes tienen una sintaxis declarada por sus creadores o instituciones que deben cumplirse para su correcta ejecución.

Antes de conocer esta sintaxis y empezar a programar con JavaScript es importante entender algunos conceptos:

- Un programa consiste en una secuencia de "instrucciones" para ser ejecutadas por un computador.
- Las instrucciones deben ser gramaticalmente correctas para que el computador las entienda.

Errores de sintaxis

- ¿Qué sucede si escribimos `3 = a`?
- ¿O si utilizamos una variable no definida?
Ejemplo: `alert(zq)`

Probemos en el inspector de elementos



Ejercicio

Realicemos un ejercicio en que guardemos en una variable nuestro nombre y ocupemos *alert* para imprimirlo en el navegador:

1. Abre el navegador.
2. Abre la consola del navegador en el inspector de elementos.
3. Define una variable que almacene tu nombre, recuerda usar las comillas simples o dobles sobre tu nombre.
4. Utiliza el *alert* para mostrar el valor guardado de la variable.

Ejercicio ¡Manos al teclado!



`/* Ejecutar javascript en la consola del navegador */` 

`/* Crear un script que ejecute operaciones matemáticas
*/`

`/* Manipular el DOM de una página web con JavaScript:
*/`

- Cambiar el contenido de un elemento
- Cambiar el estilo de un elemento

`/* Cambiar el contenido o estilo en función del
contenido ingresado en el prompt */`

Objetivos

`/* Sintaxis de JavaScript */`

Escribiendo código JavaScript en el editor

Para escribir JavaScript dentro del HTML podemos ocupar la etiqueta ***<script>***

```
<h1>JavaScript</h1>

<script>

  alert("Aprende Haciendo")

</script>
```

Podemos cargar javascript de un archivo externo utilizando el atributo **src** para especificar la ubicación del archivo.

```
<h1>JavaScript</h1>

<script src="index.js"></script>
```

¿Dónde guardar los archivos JS?

Al igual que cuando trabajamos con CSS, podemos guardar los archivos en una subcarpeta (o sub-subcarpeta):

proyecto

- assets
 - css
 - js
 - script.js

```
<h1>JavaScript</h1>
```

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

Variables en JavaScript

Cuando escribimos programas ocupamos variables para almacenar valores que necesitemos ocupar en diferentes situaciones.

```
numero =  
100  
numeroDecimal =  
0.5  
frase = "Hola a todos"
```

¿Dónde agregar la etiqueta script?

Se puede agregar dentro del head, o en el body como cualquier etiqueta html.

Por ahora lo agregaremos al final justo antes de cerrar el body.

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7     <title>Document</title>
8   </head>
9   <body>
10    <p>
11      Lorem ipsum dolor sit amet consectetur adipisicing elit. Architecto
12      magnam, repudiandae sint optio, dolorem assumenda dignissimos a amet
13      doloribus eaque voluptatem blanditiis laboriosam quis mollitia? Beatae
14      eius enim minima id?
15    </p>
16    <script> alert("hola") </script>
17  </body>
18 </html>
19
```

Calculando la suma de dos valores

Creemos un programa que muestra la suma de los números guardados en dos variables.

```
<script>  
numero1 = 100  
numero2 = 50  
alert(numero1 + numero2)  
</script>
```

`/* Ejecutar javascript en la consola del navegador */` ✓

`/* Crear un script que ejecute operaciones matemáticas */` ✓

`/* Manipular el DOM de una página web con JavaScript: */`

- Cambiar el contenido de un elemento
- Cambiar el estilo de un elemento

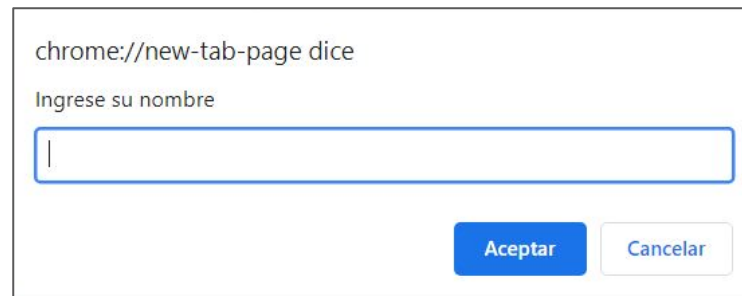
`/* Cambiar el contenido o estilo en función del contenido ingresado en el prompt */`

Objetivos

Prompt

Al momento de crear una variable, puede que necesitemos que su valor o contenido sea definido por el usuario, para esto podemos ocupar ***prompt***. El valor ingresado lo guardaremos en una variable.

```
nombre = prompt("Ingrese su nombre")
```



The image shows a browser dialog box with the title "chrome://new-tab-page dice". Inside the dialog, the text "Ingrese su nombre" is displayed above a text input field. The input field is empty with a vertical cursor. At the bottom right of the dialog, there are two buttons: "Aceptar" (Accept) in a blue box and "Cancelar" (Cancel) in a white box with a blue border.

Concatenación

La concatenación consiste en la unión de textos cuyo objetivo es producir textos compuestos, de esta forma podemos juntar uno existente con otro que ingrese el usuario vía prompt.

```
nombre = prompt("Ingrese su nombre")  
alert("Hola " + nombre)
```

Sumando números introducidos por prompt

```
num1 = prompt("Ingrese el primer número")  
num2 = prompt("Ingrese el segundo número")  
  
alert("El resultado es: " + resultado)
```

¿Qué sucede ahora si intentamos sumar 2 números? Probemos.

Transformando un texto en un número

En javascript existen los tipos de datos, "2" es un texto, 2 es un entero. Cuando sumamos 2 textos se juntan o concatenan.

```
num1 = prompt("Ingrese el primer número")  
num2 = prompt("Ingrese el segundo número")  
  
resultado = Number(num1) + Number(num2)  
  
alert("El resultado es: " + resultado)
```

chrome://new-tab-page dice

El resultado es: 6

Aceptar

Ejercicio

Crear un script en donde le pidamos al usuario que ingrese 2 números a través del *prompt* y posteriormente imprimamos con el *alert* la suma de ambos:

1. Crea un html base, al final (antes del body) introduce la etiqueta script
2. Dentro del script utiliza *prompt* para solicitarle al usuario que ingrese el primer número. Deberás almacenarlo en una variable *num1*.
3. Usa el *prompt* para solicitarle al usuario que ingrese el segundo número. Deberás almacenarlo en una variable *num2*.
4. Con el *alert* muestra la suma de ambos números, recuerda transformarlos a números antes de sumarlos.

{desafío}
latam_

Ejercicio

¡Manos al teclado!



`/* Ejecutar javascript en la consola del navegador */` ✓

`/* Crear un script que ejecute operaciones matemáticas */` ✓

`/* Manipular el DOM de una página web con JavaScript: */`

- Cambiar el contenido de un elemento
- Cambiar el estilo de un elemento

`/* Cambiar el contenido o estilo en función del contenido ingresado en el prompt */`

Objetivos

Seleccionando elementos del DOM

Con JavaScript podemos manipular los elementos que creamos en HTML y cambiar su contenido, sus estilos e incluso el valor de sus atributos. Para hacer esto debemos obtener un elemento y luego modificarlo.

En este caso, ocuparemos `querySelector`:

```
elemento = document.querySelector("p")
```

Seleccionando elementos del DOM

Además, con **querySelector** podemos seleccionar en base a etiquetas, clases e id.

```
<nav class="menu">Menú</nav>

<section id="Hero-Section">
  ...
</section>

<footer>Pie de página</footer>
```

```
menu = document.querySelector(".menu")

heroSection =
document.querySelector("#Hero-Section")

pieDePagina =
document.querySelector("footer")
```


Manipulando el DOM

Una vez que tenemos seleccionado el elemento podemos modificarlo utilizando `.innerHTML`

```
<h1> </h1>
```

```
titulo = document.querySelector("h1")  
titulo.innerHTML = "Este contenido fue agregado con JavaScript"
```



¿Por qué agregamos el script al final?

En esta parte del curso estamos manipulando el DOM, o sea elementos de la página web, pero no podemos obtener o manipular un elemento que el navegador todavía no ha leído. Es por esto que estamos agregando la etiqueta script al final del documento.

```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7      <title>Document</title>
8    </head>
9    <body>
10     <p>
11       Lorem ipsum dolor sit amet consectetur adipisicing elit. Architecto
12       magnam, repudiandae sint optio, dolorem assumenda dignissimos a amet
13       doloribus eaque voluptatem blanditiis laboriosam quis mollitia? Beatae
14       eius enim minima id?
15     </p>
16     <script> alert("hola") </script>
17   </body>
18 </html>
19
```

Ejercicio

Realicemos un ejercicio en donde haya que completar un párrafo con contenido que ingrese el usuario.

El Arrendador declara que es propietario del inmueble, ubicado en _____, departamento _____, de la comuna _____.

El Arrendador declara que es propietario del inmueble, ubicado en Miguel Claro 5442, departamento 234, de la comuna Providencia.

Ejercicio ¡Manos al teclado!



Ejercicio

1. En un archivo base HTML:
 - a. Utiliza la etiqueta *p* escribe la porción del contrato de arriendo usando etiquetas *span* en los espacios por llenar
 - b. Agrega identificadores a las etiquetas *span* con un nombre que hagan referencia a su contenido
 - c. Agrega la etiqueta *script* al final del documento html (antes de `</html>`)

Ejercicio

¡Manos al teclado!



Ejercicio

2. Dentro la etiqueta script previamente agregada:
 - a. Utiliza `prompt` y guarda la información ingresada por el usuario en las variables `info1` e `info2`.
 - b. Utiliza el método `querySelector` para encontrar los elementos que vas a modificar y guarda esa información en las variables `ele1` y `ele2`.
 - c. Utiliza `ele1.innerHTML(info1)` para cambiar el valor.
3. Observa el resultado en el navegador.

Ejercicio

¡Manos al teclado!



`/* Ejecutar javascript en la consola del navegador */` ✓

`/* Crear un script que ejecute operaciones matemáticas
*/` ✓

`/* Manipular el DOM de una página web con JavaScript:
*/` ✓

- Cambiar el contenido de un elemento
- Cambiar el estilo de un elemento

`/* Cambiar el contenido o estilo en función del
contenido ingresado en el prompt */`

Objetivos

Manipulando el DOM

De la misma manera que modificamos el contenido de una etiqueta, también podemos cambiar los estilos. Por ejemplo, teniendo la siguiente etiqueta h3:

```
<h3> Desafío Latam </h3>
```

```
var titulo = document.querySelector("h3")  
titulo.style.color = "#7ba238"
```



Manipulando el DOM

Existe una pequeña diferencia entre cómo llamamos las propiedades de CSS en JavaScript a cómo lo escribimos directamente en un documento CSS.

Por ejemplo, la propiedad *border-radius* la escribimos separando las palabras con un guión medio.

```
border-radius: 50%
```


CamelCase

En JavaScript cuando queramos ocupar una propiedad de CSS que utilice 2 o más palabras, deberemos recurrir al Camel Case, es decir, que tendremos que juntar las palabras y a partir de la segunda palabra cada una empezará con mayúscula.

```
imagen.style.borderRadius = "50%"
```

```
texto.style.fontFamily = "Fantasy"
```

```
texto.style.fontSize= "10px"
```

Ejemplos:

borderRadius

fontFamily

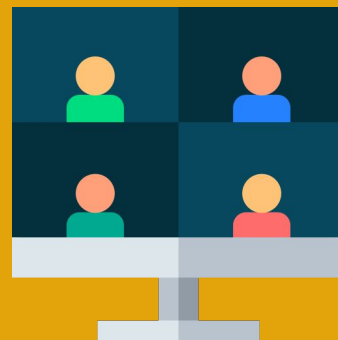
fontSize

Ejercicio

Realicemos un ejercicio en donde redondemos una imagen con CSS aplicando un *border-radius* definido por el usuario

1. En un archivo HTML nuevo, crea una base con una etiqueta *img* que ocupe una imagen con dimensiones cuadradas (<https://placedog.net/500/500>)
2. Utilizar el método `querySelector` para seleccionar la etiqueta *img*
3. Utiliza el *prompt* para solicitar al usuario el valor de la propiedad *border-radius* (tip: utilizar *camelCase*)
4. Observa el resultado en el navegador

Ejercicio Guiado ¡Todos juntos!



`/* Ejecutar javascript en la consola del navegador */` ✓

`/* Crear un script que ejecute operaciones matemáticas
*/` ✓

`/* Manipular el DOM de una página web con JavaScript:
*/` ✓

- Cambiar el contenido de un elemento
- Cambiar el estilo de un elemento

`/* Cambiar el contenido o estilo en función del
contenido ingresado en el prompt */` ✓

Objetivos



Cierre

{desafío}
latam_



¿Existe algún concepto que no
hayas comprendido?
¿Por qué crees que te resulta
difícil?

Reflexionemos

- Revisar la guía que trabajarán de forma autónoma.
- Revisar en conjunto el desafío.

¿Qué sigue?



*Academia de
talentos digitales*

www.desafiolatam.com



/DesafioLatam



/DesafioLatam



/DesafioLatam



/DesafioLatam