

Fundamentos JavaScript

Sesión 2 Semana 1



JavaScript

JavaScript es un lenguaje de programación que nos permite dar instrucciones al ordenador, en este caso al navegador web , para explicarle cómo debe mostrar nuestra página y que debe hacer en determinada situación (si se pulsa un botón, si se rellena un campo, o si pulsamos un enlace



Operadores aritméticos de JavaScript

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
**	Exponentiation (<u>ES2016</u>)
/	Division
%	Modulus (Division Remainder)
++	Increment
--	Decrement

Variables

Las variables permiten guardar información de forma temporal y estarán disponibles hasta que la página web en donde se está ejecutando el código se cierre o se recargue.

Let: Let te permite declarar variables limitando su alcance (scope).

Const: Una "variable" cuyo valor nunca varía recibe el nombre de constante.

Var: var define una variable global o local en una función sin importar el ámbito.

Recordemos...

Una **variable local** es aquella cuyo ámbito se restringe a la función que se declara. Una **variable global** se define fuera del cuerpo de cualquier función.

Comandos a usar en JavaScript

var = variable global – últimamente está perdiendo su uso.

Let = variable local

Number () = convierte texto a número.

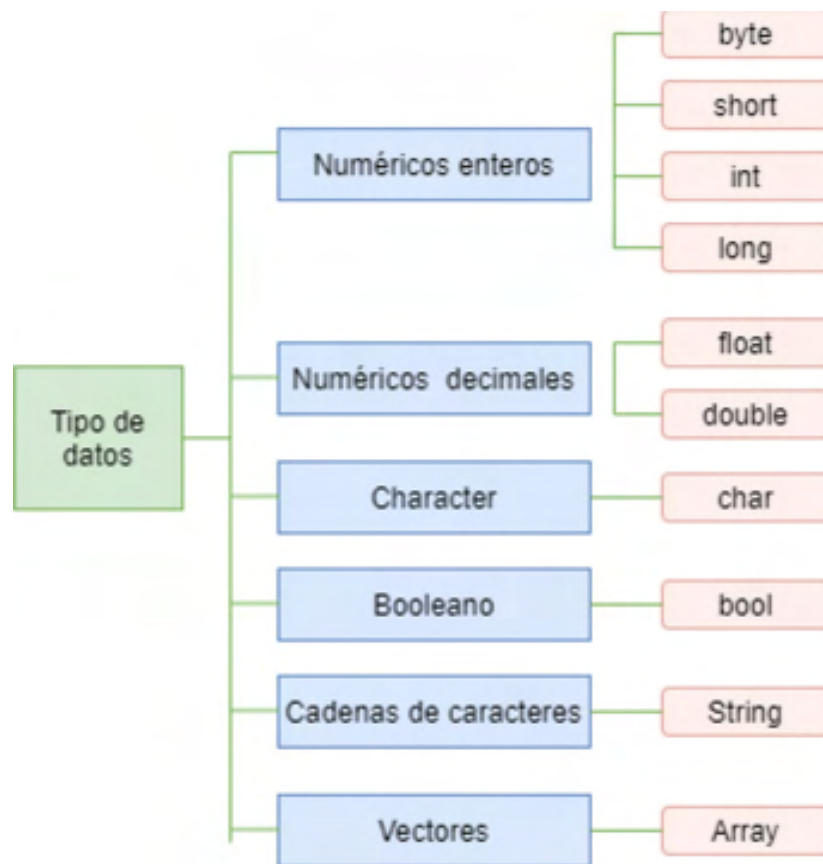
prompt () = Ventana emergente para ingresar datos.

document.write ()= Muestra información en el archivo index.html

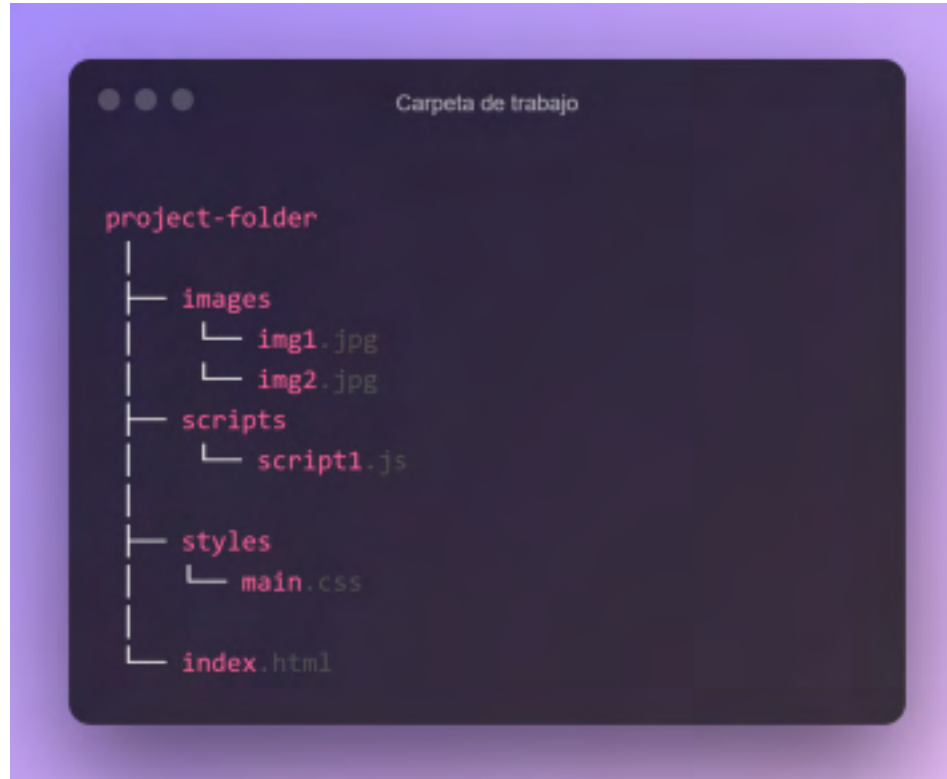
alert() = ventana emergente en el navegador

Console.log() = Imprime en consola

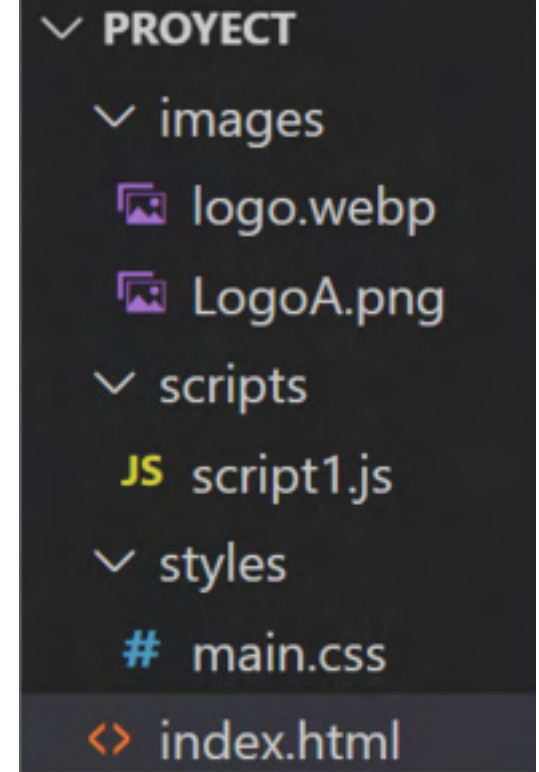
Tipo de datos



Estructura de trabajo



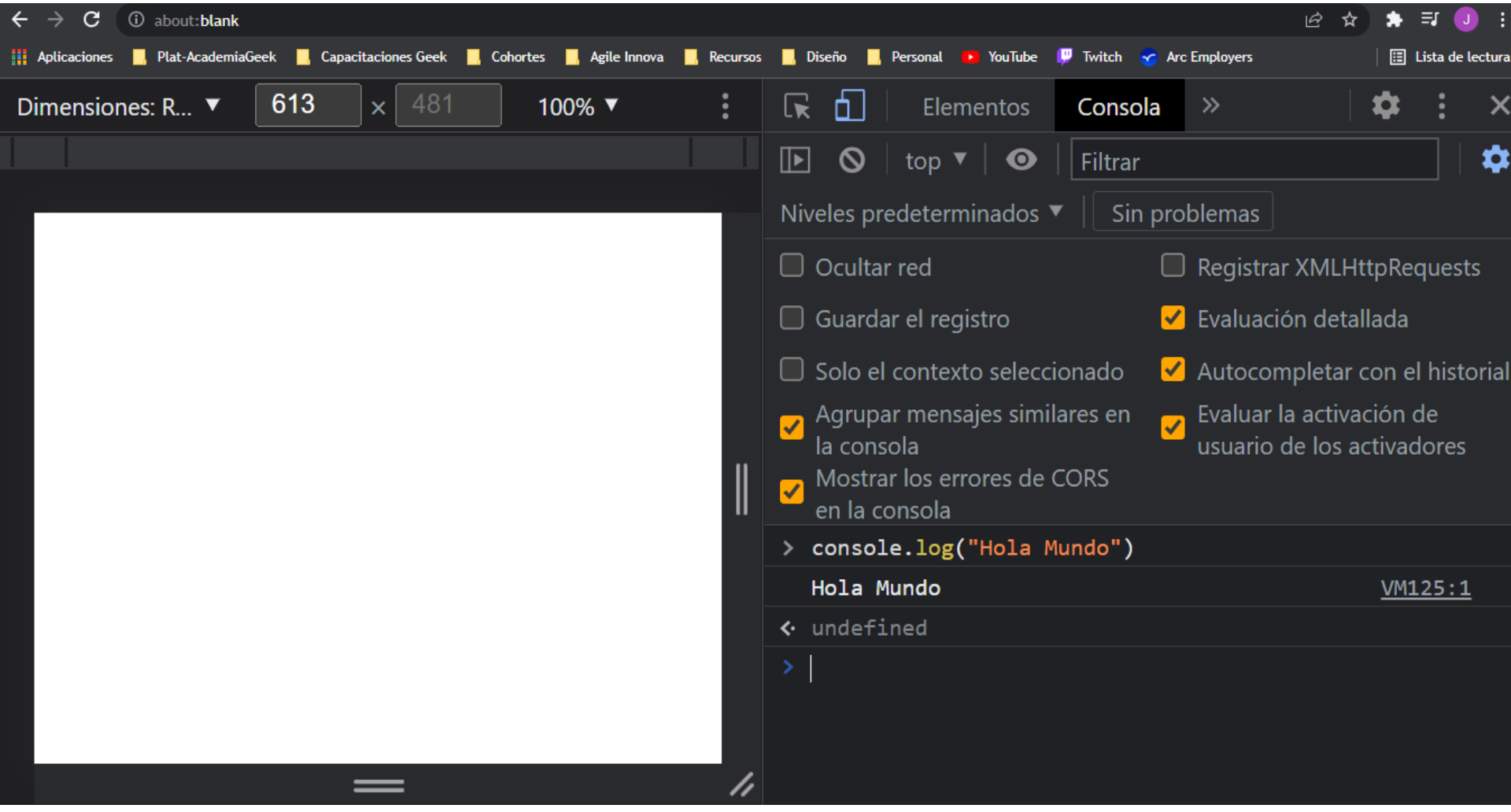
- Los nombres de archivo irán siempre en minúsculas, sin tildes, sin espacios y sin caracteres especiales.
- Usaremos guiones para separar palabras
- Los nombres de archivo siempre en inglés
- El archivo de estilos principal se llamará style.css, o main.css
- El archivo HTML principal se llamará index.html



Herramientas de trabajo

- Visual Studio Code <https://code.visualstudio.com/>
- Git <https://git-scm.com/>
- Wakatime <https://wakatime.com/>





← → ↻ ⓘ about:blank

🔍 🔗 ☆ ⚙️ ☰ J ⋮

📁 Aplicaciones 📁 Plat-AcademiaGeek 📁 Capacitaciones Geek 📁 Cohortes 📁 Agile Innova 📁 Recursos 📁 Diseño 📁 Personal 📁 YouTube 📁 Twitch 📁 Arc Employers

📄 Lista de lectura

Hola Mundo

🖱️ 📄 Elementos **Consola** >> ⚙️ ⋮ ✕

🎬 🚫 top ▼ 👁️ Filtrar 1 oculto ⚙️

Niveles predeterminados ▼ Sin problemas

- | | |
|---|---|
| <input type="checkbox"/> Ocultar red | <input type="checkbox"/> Registrar XMLHttpRequests |
| <input type="checkbox"/> Guardar el registro | <input checked="" type="checkbox"/> Evaluación detallada |
| <input type="checkbox"/> Solo el contexto seleccionado | <input checked="" type="checkbox"/> Autocompletar con el historial |
| <input checked="" type="checkbox"/> Agrupar mensajes similares en la consola | <input checked="" type="checkbox"/> Evaluar la activación de usuario de los activadores |
| <input checked="" type="checkbox"/> Mostrar los errores de CORS en la consola | |

> document.write("Hola Mundo")

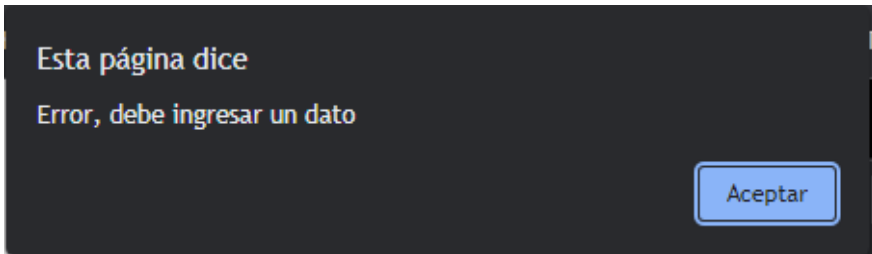
<< undefined

>

- Permite hacerle pruebas a nuestro código y verificar cómo está funcionando nuestro código de JavaScript.

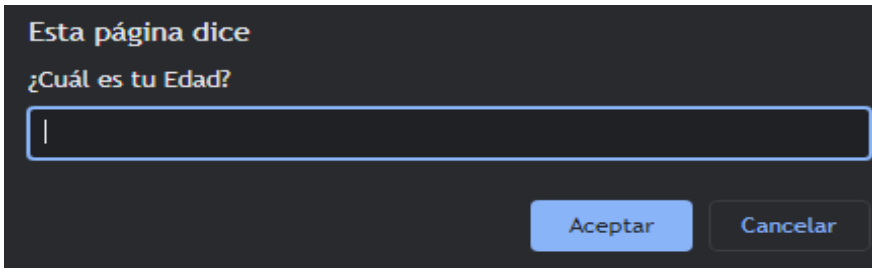
Alert():

```
> alert('Error, debe ingresar un dato')  
>
```



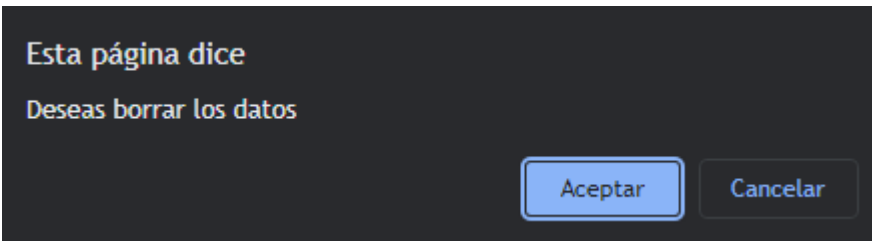
Prompt():

```
> prompt('¿Cuál es tu Edad?')  
>
```

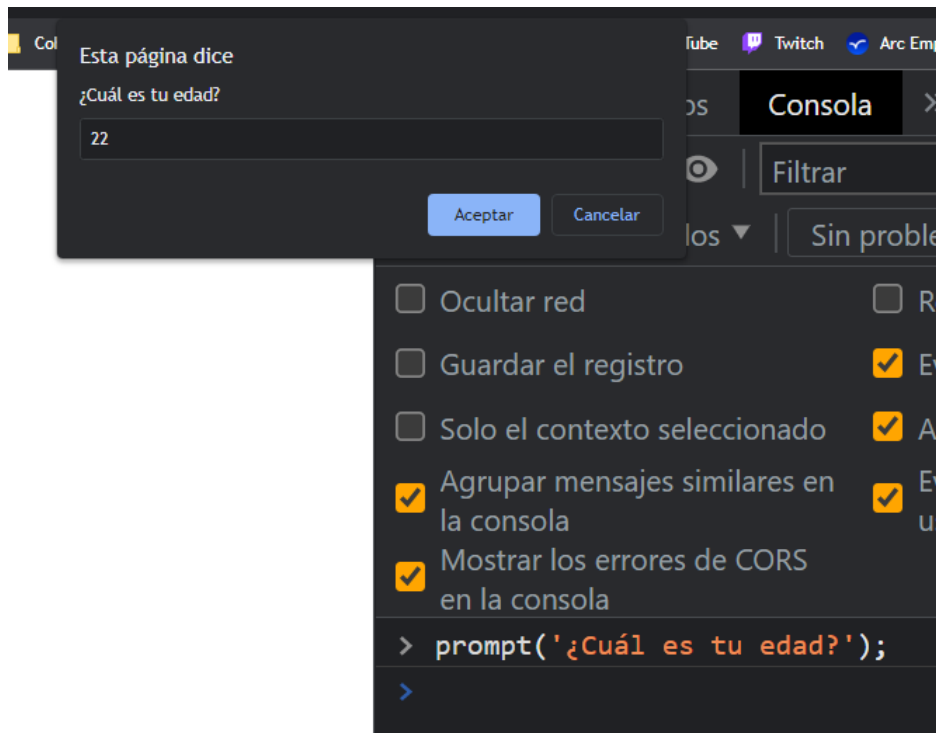


Confirm():

```
> confirm('Deseas borrar los datos')  
< true
```

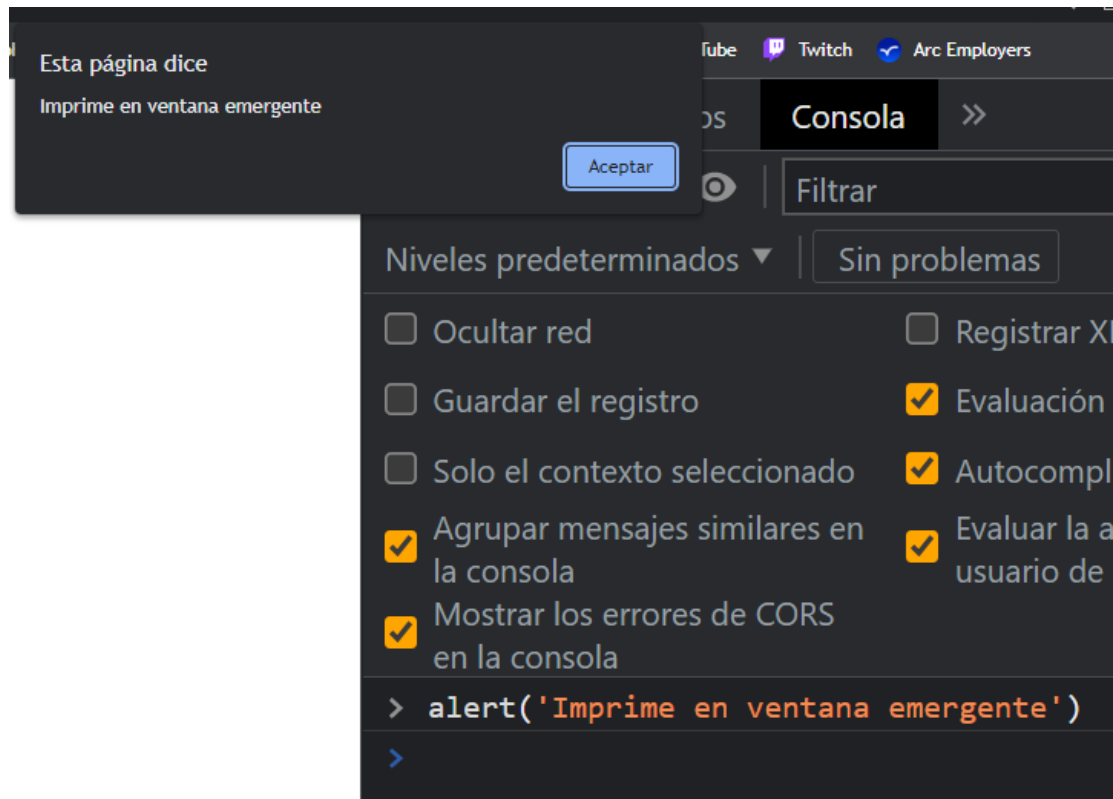


Consola Chrome



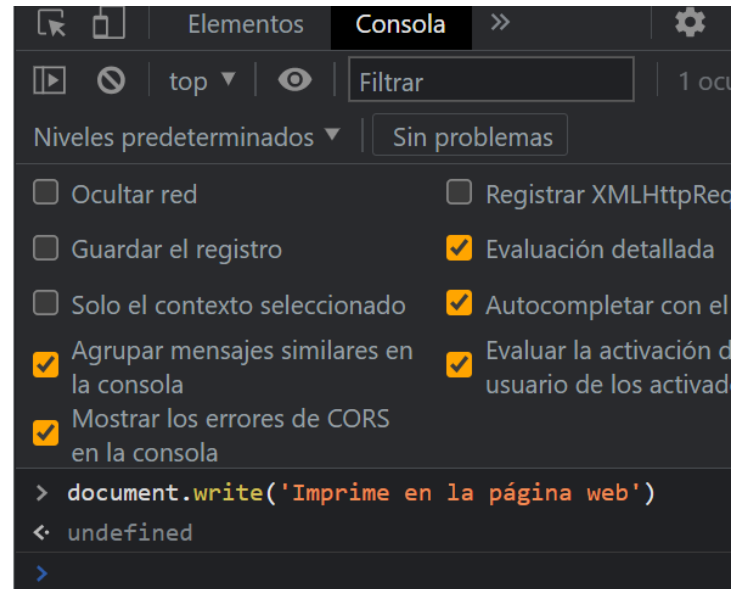
```
> prompt('¿Cuál es tu edad?');  
< '22'  
>
```

Imprimir Información



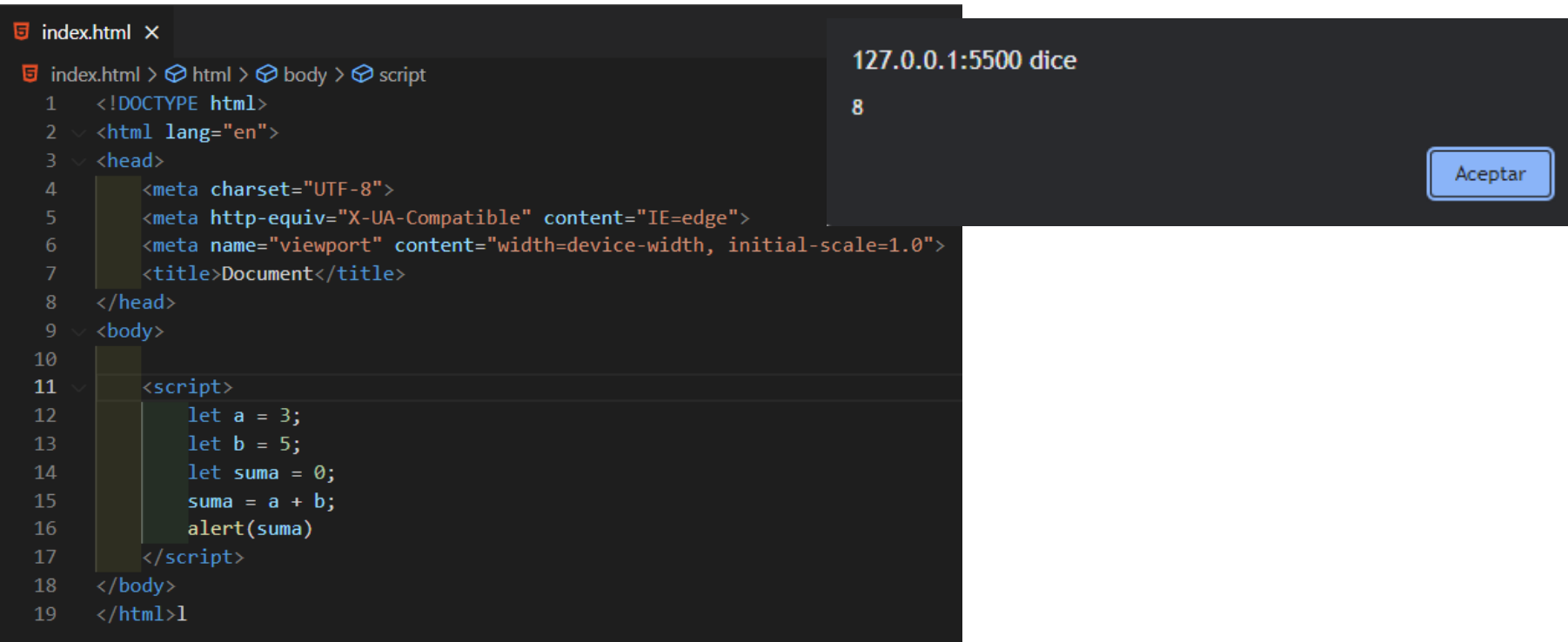
```
> console.log('Imprime en la consola de chrome')
Imprime en la consola de chrome          VM877:1
< undefined
> |
```

Imprime en la página web



Nuestro primer código en JavaScript

1. JS dentro del mismo archivo HTM



```
index.html ×  
index.html > html > body > script  
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  
11 <script>  
12   let a = 3;  
13   let b = 5;  
14   let suma = 0;  
15   suma = a + b;  
16   alert(suma)  
17 </script>  
18 </body>  
19 </html>
```

127.0.0.1:5500 dice
8

Aceptar

2. JS dentro del mismo archivo HTML

```
index.html x
index.html > html > body > script
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
11  <script>
12      let msg = "¿Está Seguro?";
13      prompt(msg);
14  </script>
15 </body>
16 </html>
```

127.0.0.1:5500 dice

¿Está Seguro?

Aceptar

Cancelar

3. Crear archivo HTML - index.html

index.html ✕

index.html > html > head > meta

```
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
11 </body>
12 </html>l
```

2. Crear archivo JavaScript - script.js

index.html

script.js



script.js > ...

```
1  const mensaje = "Este es un mensaje de ejemplo";  
2  
3  console.log(mensaje)
```

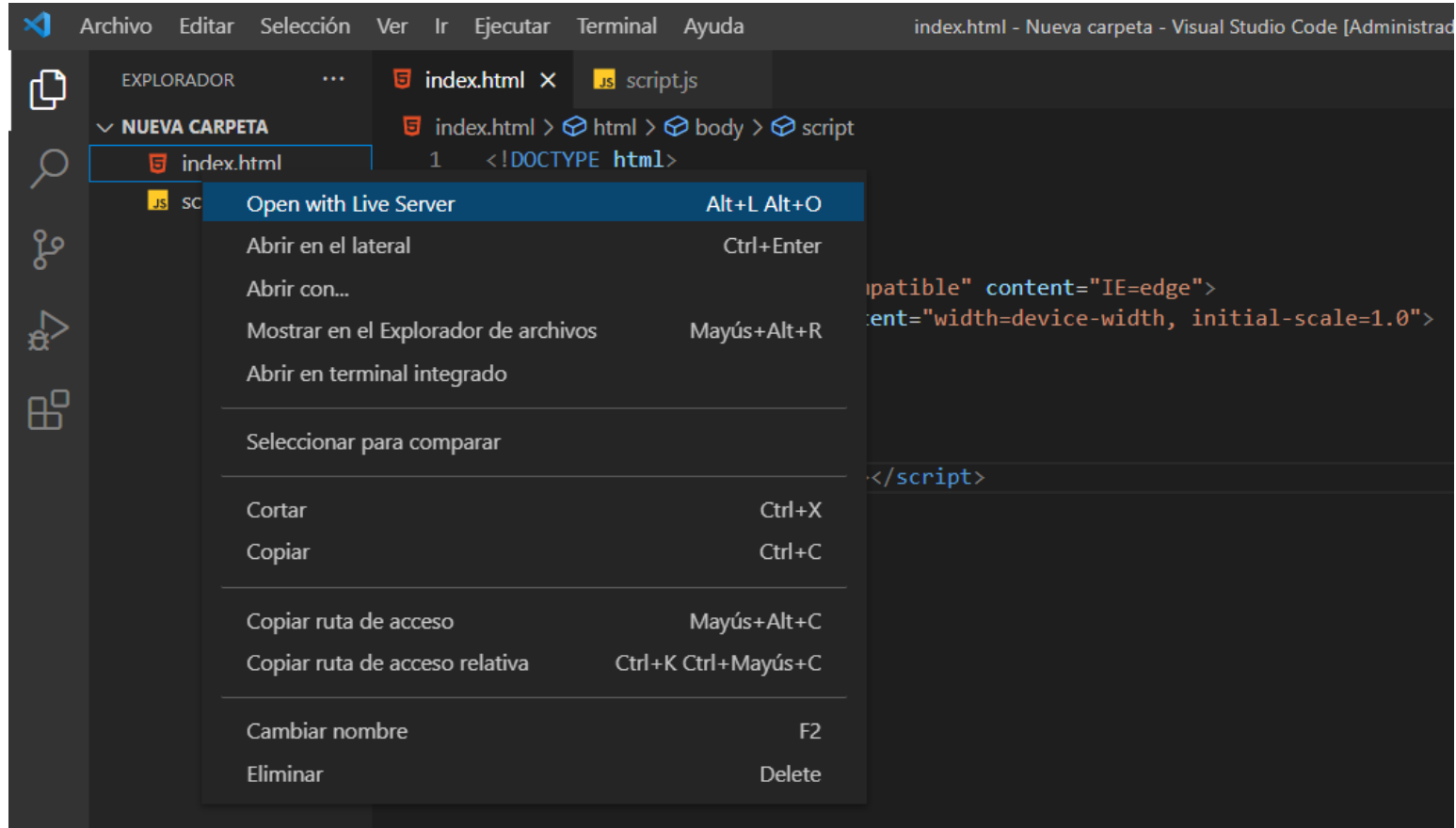
3. Vincular el archivo JavaScript con el archivo HTML

index.html X JS script.js

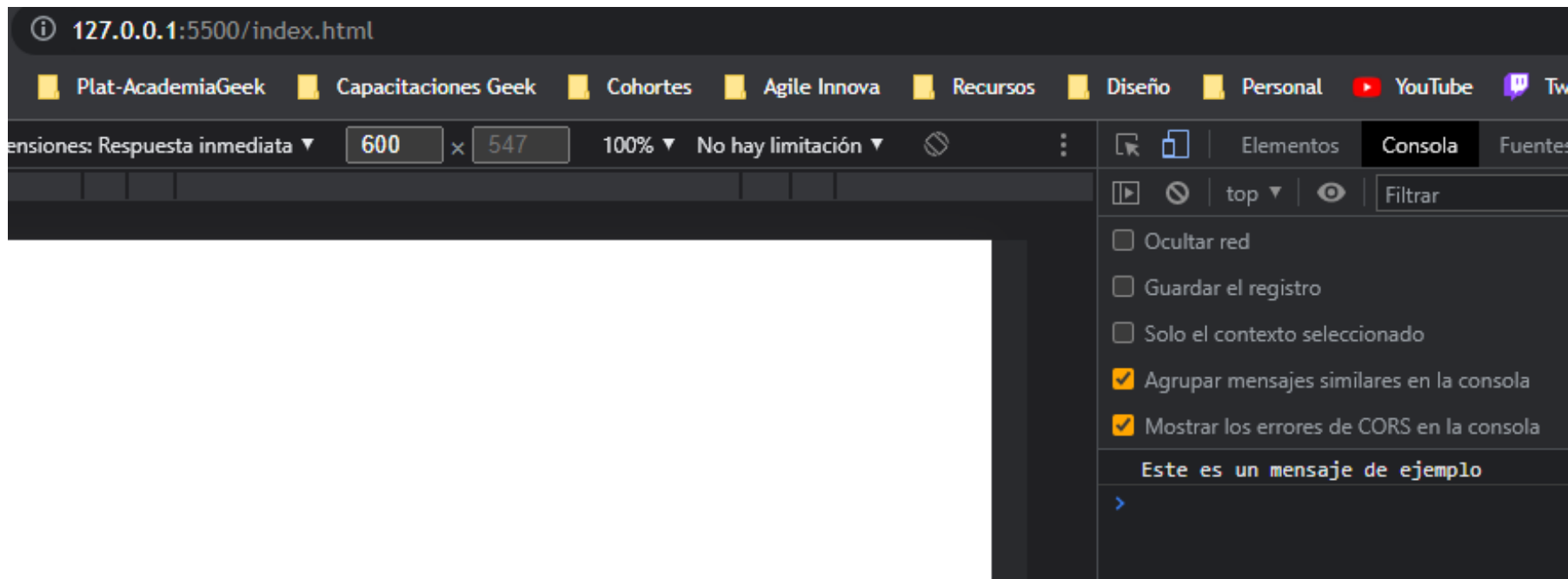
index.html > html > body > script

```
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
11  <script src="./script.js"></script>
12 </body>
13 </html>
```

4. Ejecutamos el programa (Live Server)

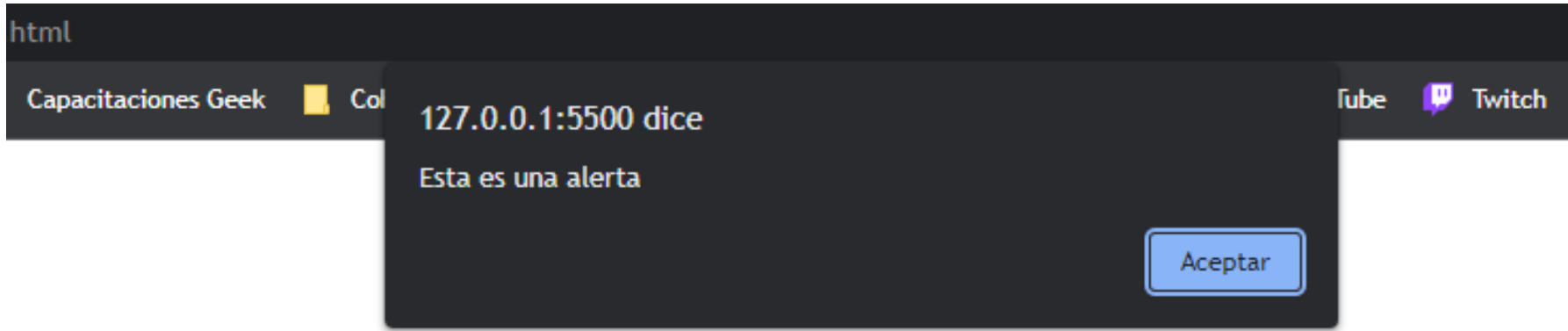


5. Resultados

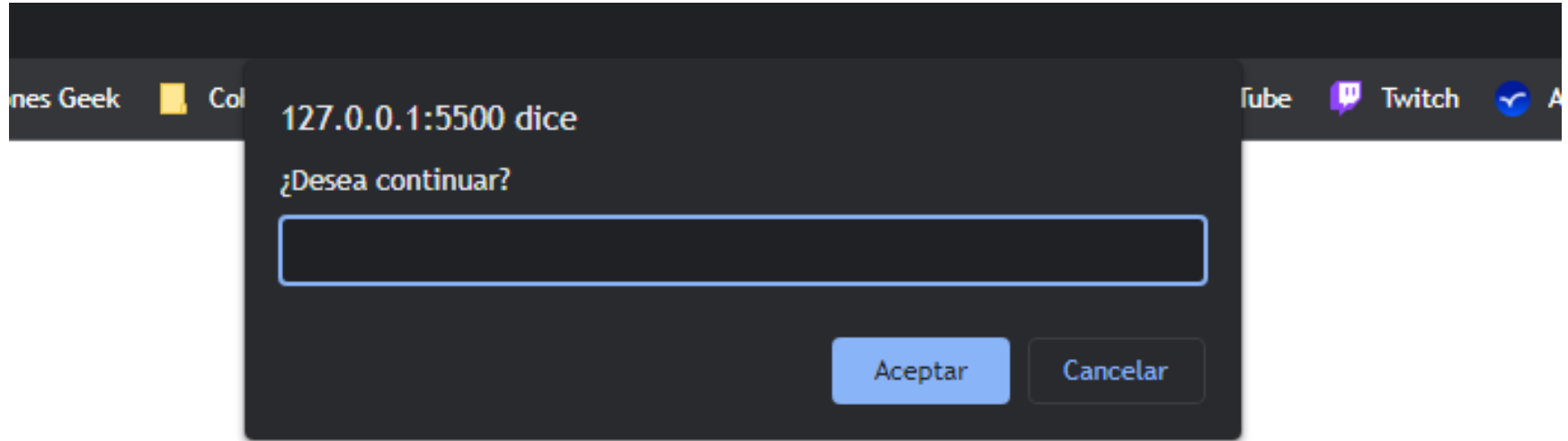


Ejercicio

1. Realizar los cambios necesarios en el archivo script.js para mostrar el siguiente mensaje en el navegador



2. Realizar los cambios necesarios en el archivo script.js para mostrar el siguiente mensaje en el navegador



Seleccionar un elemento 4

En JavaScript podemos seleccionar un elemento de varias formas:

1. Por su Id:

```
//Accedemos directamente al elemento deseado por su id única  
var elem = document.getElementById("id_del_elemento");
```

2. Por su Clase:

```
//Obtenemos un array de elementos con esa misma clase  
var elem = document.getElementsByClassName("clase_del_elemento");
```

3. Por su etiqueta:

```
//Obtenemos un array de elementos con esa misma etiqueta
```

```
var elem = document.getElementsByTagName("etiqueta_del_elemento");
```

document.getElementById()

getElementById() permite obtener la referencia a un elemento de la página mediante el id de dicho elemento.

Propiedades:

- 1. innerHTML: Cambiar contenido**
- 2. style.fontSize: Cambiar fuente.**
- 3. Style.display: Ocultar elementos**

Cambiar tamaño de letra

HTML

```
...  <> index.html X  JS script.js

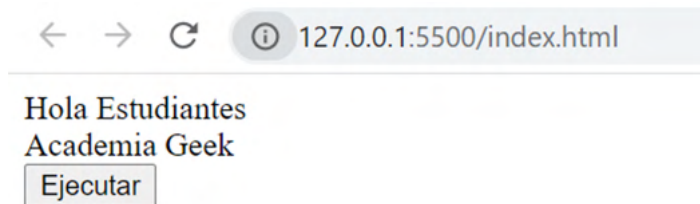
index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Document</title>
8  </head>
9  <body>
10     <div id="demo">Hola Estudiantes</div>
11     <div id="demo2">Academia Geek</div>
12     <button type="button" onclick="Enviar()">Ejecutar</button>
13     <script src="script.js"></script>
14 </body>
15 </html>
```

JavaScript

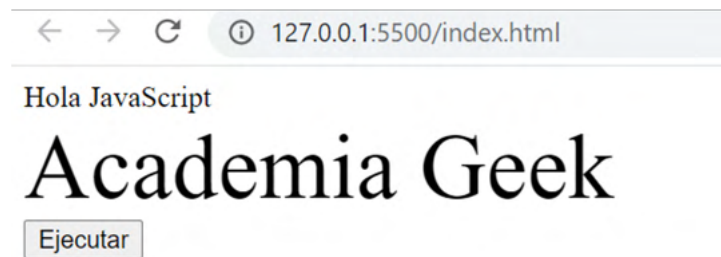


```
<> index.html JS script.js X  
JS script.js > Enviar  
1 function Enviar(){  
2     document.getElementById('demo').innerHTML = 'Hola JavaScript';  
3     document.getElementById('demo2').style.fontSize = '50px'  
4 }
```

Resultados Navegador



Resultados luego de hacer clic sobre el botón Ejecutar



Mostrar y ocultar elementos

HTML

```
<!-- Mostrar o ocultar elementos-->
<div id="elemento" class="container text-center py-5">
|   <h1>Elementos de clase</h1>
</div>

<div id="" class="container text-center py-5">
|   <input type="button" value="Mostrar" onclick="mostrar()">
|   <input class="btn btn-success" type="button" value="Ocultar" onclick="Ocultar()">
</div>
```


Script:

```
function mostrar(){  
    document.getElementById("elemento").style.display= "block";  
}
```

```
function Ocultar(){  
    document.getElementById("elemento").style.display= "none";  
}
```

document.querySelector()

El método `querySelector ()` devuelve el primer elemento que coincide con un selector o selectores CSS especificados en el documento

```
const mainTitle = document.querySelector('h1');
```



```
var el = document.querySelector(".miClase");
```

Ejemplo 1

```
<> index.html M X
<> index.html > html > body > script
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>Mensaje 1</p>
11
12     <button onclick="myFuncion()">Botón</button>
13
14     <script>
15         function myFuncion(){
16             document.querySelector("p").style.background = "blue"
17         }
18
19     </script>
20 </body>
21 </html>
```

Ejemplo 2

<> index.html M X

<> index.html > html > body > p.ejemplo

```
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 class="ejemplo">Mensaje 2</p>
11
12     <button onclick="myFuncion()">Botón</button>
13
14     <script>
15         function myFuncion(){
16             document.querySelector("p.ejemplo").style.background = "red"
17         }
18     </script>
19 </body>
20 </html>
```

document.querySelectorAll()

El método querySelectorAll () devuelve todos los elementos del documento que coinciden con un selector o selectores CSS especificados

```
<> index.html M X
<> index.html > html > body > div.ejemplo
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   <div class="ejemplo">Ejemplo 1</div>
11   <div class="ejemplo">Ejemplo 2</div>
12   <div class="ejemplo">Ejemplo 3</div>
13   <div class="ejemplo">Ejemplo 4</div>
14   <p id="demo"></p>
15
16   <button onclick="myFuncion()">Botón</button>
17
18   <script>
19     function myFuncion(){
20       var x = document.querySelectorAll(".ejemplo");
21       document.getElementById("demo").innerHTML = x.length;
22     }
23   </script>
24 </body>
25 </html>
```

Fundamentos de JavaScript

Sesión 2 Semana 1

