

	<b>UNIVERSIDAD GERARDO BARRIOS</b> <b>FACULTAD CIENCIA Y TECNOLOGÍA</b>
<b>CICLO I-2022</b>	<b>PROGRAMACIÓN COMPUTACIONAL IV</b> <b>PRÁCTICA: REPASO JAVASCRIPT</b>

# Introducción

JavaScript es un lenguaje de secuencias de comandos del lado del cliente de código abierto y más popular compatible con todos los navegadores. JavaScript se utiliza principalmente para mejorar la interacción de la página web con los usuarios haciéndola más animada e interactiva. También se utiliza para el desarrollo de juegos y desarrollo de aplicaciones móviles.

JavaScript no es un lenguaje de programación propiamente dicho como C, C++ etc., es un lenguaje script u orientado a documento, como pueden ser los lenguajes de macros que tienen muchos procesadores de texto y planillas de cálculo.

Hasta hace poco no se podía desarrollar programas con JavaScript que se ejecutaran fuera de un Navegador, aunque en este momento comienza a expandirse a otras áreas como la programación en el servidor con Node.js

JavaScript es un lenguaje interpretado que se embebe en una página web HTML. Un lenguaje interpretado significa que a las instrucciones las analiza y procesa el navegador en el momento que deben ser ejecutadas.

```
<> index.html ●
<> index.html > html > 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     <!--El codigo javascript debe ir encerrado entre las etiquetas <script>-->
11     <script>
12         document.write('Programación Computacional IV');
13     </script>
14 </body>
15 </html>
```

# Variables

Las variables se utilizan para almacenar valores (nombre = "Jose") o expresiones (suma = x + y).

Existen 4 formas de declarar una variable en JavaScript:

- var
- let
- const
- sin utilizar una palabra reservada

```
var name= "Gisela";
```

```
let x=22;
```

```
const price=2.50;
```

```
x=30;
```

- var: La palabra clave se usa en todo el código JavaScript desde 1995 hasta 2015.
- let y const: Las palabras clave se agregaron a JavaScript en 2015.
- Variables definidas con let no se pueden volver a declarar.
- Variables definidas con const no se pueden reasignar

## Tipos de variables

Una variable puede almacenar:

Valores Enteros (100, 260, etc.)

Valores Reales (1.24, 2.90, 5.01, etc.)

Cadenas de caracteres ('Juan', 'Compras', 'Listado', etc.)

Valores lógicos (true,false)

Tutorial JavaScript: <https://www.w3schools.com/js/default.asp>

## EJEMPLO

El ejemplo por realizar consiste en la creación de una lista dinámica.

index.html

```
index.html
what > index.html > html > 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>Ejemplo</title>
8
9   <!--incluir archivos css-->
10  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhF1dvKuhfTAU6au08tT94W/HftJDbrCExsU1oBoqy12QvZ6jIN3"
11 </head>
12 <body>
13   <div class="container">
14     <h1>Ejemplo Javascript </h1>
15     <div class="mb-3">
16       <h2>Agregar elemento:</h2>
17       <input type="text" id="addItem" class="form-control">
18       <button id="addItemButton" class="btn btn-success">Add</button>
19       <button id="removeItemButton" class="btn btn-danger">Del Last</button>
20       <button id="removeItem" class="btn btn-danger">Del Selected</button>
21     </div>
22     <div class="mb-3">
23       <button id="hideElements" class="btn btn-secondary"><<</button>
24     </div>
25
26     <div class="mb-3" id="list">
27       <!-- lista de elementos -->
28       <div class="card">
29         <div class="card-body">
30           <ul id="listItems">
31             <!-- aqui se agregaran los elementos -->
32           </ul>
33         </div>
34       </div>
35     </div>
36     <script src="app.js"></script> <!--enlace a archivo js-->
37
38   </div>
39
40   <!--incluir archivo js de bootstrap-->
41   <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js" integrity="sha384-7+zCNj/IqJ95wo16oMtfskbz9ccEh31e0z1H0yDuCQ6wgnyJNSVdrPa03rtrR1zdB" crossorigin="anonymous"></script>
42   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js" integrity="sha384-QJHtvGhmr9XOIpI6VUutG+2QOK9T+ZnN4kz7FNI1RTK3zEFEIsxhlmW15/YESvpZ13" crossorigin="anonymous"></script>
43
44 </body>
45 </html>
```

app.js

### Definición de constantes

```
//ocultar o mostrar div que contiene la lista
const hide = document.querySelector('#hideElements');
const listDiv = document.querySelector('#list');

//constantes para agregar elementos a la lista
const addItemInput = document.querySelector('#addItem'); //input
const addItemButton = document.querySelector('button#addItemButton'); //button

//eliminar ultimo item
const removeItemButton = document.querySelector('button#removeItemButton');
//eliminar elemento seleccionado
const removeItem = document.querySelector('button#removeItem');
```

### Definición de eventos a botones

Mostrar y ocultar div que contiene la lista de elementos

```
//mostrar y ocultar lista
hide.addEventListener('click', () => {
  if (listDiv.style.display == 'none') {
    listDiv.style.display = 'block';
    hide.textContent = '<<';
  } else {
    listDiv.style.display = 'none';
    hide.textContent = '>>';
  }
});
```

Agregar elementos a la lista

```
//agregar elementos a la lista
addItemButton.addEventListener('click', () => {
  let list = document.querySelector('ul');
  let li = document.createElement('li');
  li.textContent = addItemInput.value;
  list.appendChild(li); //agregar elemento a la lista
  addItemInput.value="";
});
```

Eliminar ultimo elemento agregado a la lista

```
//eliminar ultimo elemento de la lista
removeItemButton.addEventListener('click', () => {
  let list = document.querySelector('ul');
  let li = document.querySelector('li:last-child');
  list.removeChild(li);
});
```

**Eliminar elemento seleccionado**

Obtener index del elemento seleccionado

```
function findIndex(elem) {
  var i, len = items.length;
  for(i=0; i<len; i++) {
    if (items[i]===elem) {
      return i;
    }
  }
  return -1;
}

//obtener index of element in list
var list = document.getElementById("listItems"),
    items = list.getElementsByTagName("li");
list.onclick = function(e) {
  var event = e || window.event,
      src = event.target || event.srcElement;
  var myIndex = findIndex(src);
  // alert(myIndex);
  index=myIndex;
  console.log(myIndex);

  //cambiar color del elemento seleccionado
  if (event.target.tagName === 'LI' ) {
    list.querySelectorAll('li').forEach(el=>el.classList.remove('alert','alert-success')); //se agregan clases bootstrap para los elementos
    items[index].classList.add('alert','alert-success');
  }
};
```

```

//eliminar elemento seleccionado
removeItem.addEventListener('click', () => {
  let showDiv = document.getElementById('list');
  let divAlert = document.createElement('div');
  let label = document.createElement('label');
  label.innerHTML= 'Se elimino el elemento '+index;
  divAlert.classList.add('alert', 'alert-danger');//clase de la alerta

  items[index].parentNode.removeChild(items[index]);
  divAlert.appendChild(label);
  showDiv.appendChild(divAlert);
});

```

## Ejercicios complementarios

1. Modificar el ejercicio anterior de tal manera que en el alert deberá mostrar el texto del elemento eliminado.
2. Publicar el proyecto en github pages de javascript
3. Haciendo uso de directivas, represente los datos academicos de cada integrante del equipo, aplique un estilo diferente para cada caso
4. Crear una variable que contenga codigo HTML, en el cual represente los nombres de los integrantes con las etiquetas <h2> y texto en cursiva, debera mostrarlo en utilizando v-html
5. Declarar un arreglo con datos de mascotas, con los datos: nombre y edad, mostrar el contenido del array dentro de una lista <ul>
6. Publicar el proyecto en GitHub
7. Una vez finalizados los ejercicios deberá compartir el enlace al proyecto en GitHub