

Universidad Mariano Gálvez de Guatemala  
Sede Villa Nueva, Jornada Matutina  
Ingeniería en Sistemas de Información y Ciencias de la Computación  
Desarrollo Web  
Docente: Ing. Carlos Alejandro Arias

## **Laboratorio 3 – Sitio JavaScript**

Angelo Diancarlo Quezada Gallardo 5090-21-9826

Villa Nueva, 31 de julio de 2024

## **Introducción**

Durante este laboratorio, nos centramos en la creación de una calculadora interactiva utilizando HTML, CSS y JavaScript. Adicionalmente, estilizamos la página web con un diseño vintage para mejorar la experiencia del usuario. Este informe detalla los conceptos aprendidos, los desafíos enfrentados, y las soluciones implementadas durante el desarrollo del proyecto.

## Contenido

### HTML y Estructura del Documento

#### Estructura del Documento:

- Se utilizó HTML para definir la estructura básica de la página, incluyendo el encabezado (<header>), el cuerpo principal (<main>), y el pie de página (<footer>).
- Las etiquetas semánticas mejoraron la accesibilidad y organización del contenido.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Calculadora en Consola</title>
  <link rel="stylesheet" href="./css/style.css">
</head>
<body>
  <h1>Calculadora en Consola con JavaScript</h1>
  <p>Abre la consola del navegador para interactuar con la calculadora.</p>
  <div id="resultado"></div>
  <script src="./js/calculadora.js"></script>
</body>
</html>
```

## CSS y Estilización

#### Estilos Básicos:

- Aplicamos estilos CSS para dar un aspecto vintage a la página. Utilizamos una paleta de colores suaves y fuentes clásicas.
- Se aplicaron propiedades de display, grid, flex, y position para organizar los elementos en la página.

```

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-family: 'Georgia', serif;
  background-color: #f4ecd8;
  color: #5a4d41;
  padding: 20px;
}

h1 {
  font-family: 'Courier New', Courier, monospace;
  font-size: 2.5em;
  color: #3b3027;
  text-align: center;
  margin-bottom: 20px;
  text-shadow: 2px 2px #d2c7b3;
}

```

## Diseño Responsivo:

- Se utilizaron unidades relativas y propiedades CSS para asegurar que el diseño fuera adaptativo y se viera bien en diferentes tamaños de pantalla.

```

p {
  font-size: 1.2em;
  margin-bottom: 20px;
  text-align: center;
  color: #6b5f55;
}

#resultado {
  background-color: #fff8e8;
  border: 1px solid #d2c7b3;
  padding: 20px;
  margin: 0 auto;
  max-width: 600px;
  font-size: 1.1em;
  line-height: 1.6;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

```

## JavaScript y Funcionalidad Interactiva

### Lógica de la Calculadora:

- Implementamos una función calcular que realiza operaciones matemáticas básicas (suma, resta, multiplicación, división).
- La función maneja casos especiales, como la división por cero, para evitar errores.

```
// Función para realizar las operaciones
function calcular(operador, num1, num2) {
  switch (operador) {
    case '+':
      return num1 + num2;
    case '-':
      return num1 - num2;
    case '*':
      return num1 * num2;
    case '/':
      if (num2 === 0) {
        return 'Error: División por cero';
      }
      return num1 / num2;
    default:
      return 'Operador no válido';
  }
}
```

### Interacción con el Usuario:

- Utilizamos prompt para capturar la entrada del usuario (operador y números) y alert para mostrar resultados.
- Se añadió un ciclo while para permitir al usuario realizar múltiples operaciones hasta que decida cancelar.

```
// Función principal para interactuar con el usuario
function iniciarCalculadora() {
  let continuar = true;
  let resultados = [];
  while (continuar) {
    let operador = prompt('Ingrese el operador (+, -, *, /):');
    if (operador === null) break; // Salir si el usuario cancela

    let num1 = prompt('Ingrese el primer número:');
    if (num1 === null) break; // Salir si el usuario cancela
    num1 = parseFloat(num1);

    let num2 = prompt('Ingrese el segundo número:');
    if (num2 === null) break; // Salir si el usuario cancela
    num2 = parseFloat(num2);
```

### Manipulación del DOM:

- Los resultados se muestran dinámicamente en la página web utilizando document.getElementById y la propiedad textContent.

```
// Agregar el resultado a la lista de resultados
resultados.push(`El resultado de ${num1} ${operador} ${num2} es: ${resultado}`);

// Mostrar todos los resultados acumulados en el HTML
document.getElementById('resultado').innerHTML = resultados.join('<br>');

// Preguntar al usuario si desea realizar otra operación
continuar = confirm('¿Desea realizar otra operación?');
```

### Registro de Resultados:

- Todos los resultados de las operaciones se acumulan y se muestran en el HTML para mantener un historial de las operaciones realizadas.

```
let resultado = calcular(operador, num1, num2);
console.log(`El resultado de ${num1} ${operador} ${num2} es: ${resultado}`);
alert(`El resultado de ${num1} ${operador} ${num2} es: ${resultado}`);
```

Link de GitHub Sitio:

<https://github.com/And0811/Labos/tree/3f82592eef6f077ee17df9b6cbc22c6c71578586/sitio6>

## **Conclusión**

El código HTML presentado estructura una página web sencilla con secciones dedicadas a la portada, navegación, información sobre un perro llamado Toby, y un formulario de contacto. Incluye elementos multimedia como imágenes y un video, y

está diseñado para ser responsivo y accesible en dispositivos móviles gracias a la meta etiqueta viewport.