

TEMA: Crear un componente Web para realizar operaciones básicas usando Shadow DOM y Bootstrap 5

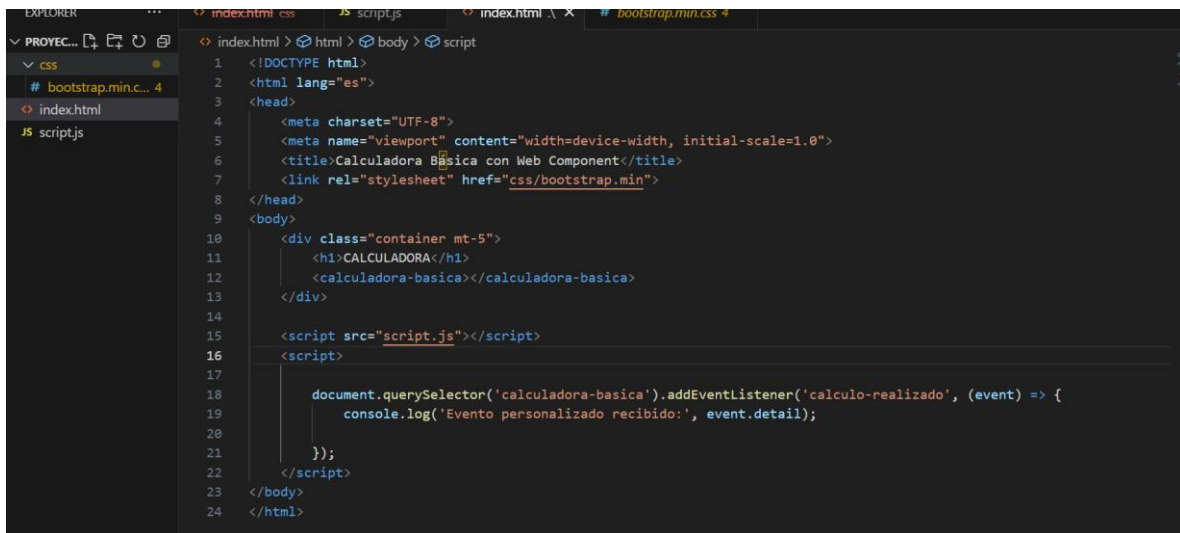
ESTUDIANTE: Nixon Alejandro Bravo Quijije

FECHA: 26-06-25

1. ENLANCE GITHUB

<https://github.com/AlejandroBravo26/Uso-de-Shadow-DOM-y-Bootstrap-5>

2. CODIGO FUENTE



```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Calculadora Básica con Web Component</title>
7   <link rel="stylesheet" href="css/bootstrap.min">
8 </head>
9 <body>
10   <div class="container mt-5">
11     <h1>CALCULADORA</h1>
12     <calculadora-basica></calculadora-basica>
13   </div>
14
15   <script src="script.js"></script>
16   <script>
17
18     document.querySelector('calculadora-basica').addEventListener('calculo-realizado', (event) => {
19       console.log('Evento personalizado recibido:', event.detail);
20     });
21   </script>
22 </body>
23 </html>
```

Index.html

JS scriptjs > ...

```
1 class CalculadoraBasica extends HTMLElement {
2   constructor() {
3     super();
4     // Creamos el Shadow DOM
5     this.attachShadow({ mode: 'open' });
6     this.historial = []; // Para el extra de historial
7
8     // Construimos el contenido del Shadow DOM
9     this.render();
10  }
11
12  render() {
13    this.shadowRoot.innerHTML = `
14      <style>
15        /* Estilos propios del componente (opcional, si no usas 100% Bootstrap) */
16        .card {
17          margin-top: 20px;
18        }
19        .form-group {
20          margin-bottom: 1rem;
21        }
22        #resultado {
23          font-size: 1.5rem;
24          font-weight: bold;
25          color: #007bff; /* Color Bootstrap primary */
26        }
27        .error-message {
28          color: #dc3545; /* Color Bootstrap danger */
29          font-weight: bold;
30        }
31      </style>
32      <link rel="stylesheet" href="css/bootstrap.min.css">
33
34      <div class="card shadow-sm p-4">
35        <div class="card-body">
36          <h5 class="card-title text-center mb-4">Calculadora</h5>
37
38          <div class="form-group">
39            <label for="numero1">Número 1:</label>
40            <input type="number" class="form-control" id="numero1" placeholder="Ingrese el primer número">
41          </div>
42
43          <div class="form-group">
44            <label for="numero2">Número 2:</label>
45            <input type="number" class="form-control" id="numero2" placeholder="Ingrese el segundo número">
46          </div>
47
48          <div class="form-group">
49            <label for="operacion">Operación:</label>
50            <select class="form-select" id="operacion">
51              <option value="suma">Suma (+)</option>
52              <option value="resta">Resta (-)</option>
53              <option value="multiplicacion">Multiplicación (<math>*</math>)</option>
54              <option value="division">División (<math>/</math>)</option>
55            </select>
56          </div>
57        </div>
58      </div>
59    `;
60  }
```

```

57
58         <button class="btn btn-primary w-100 mt-3" id="calcular">Calcular</button>
59
60         <div class="mt-4 text-center">
61             <p class="mb-1">Resultado:</p>
62             <div id="resultado" class="alert alert-info"></div>
63             <div id="errorMessage" class="error-message"></div>
64         </div>
65
66         <div id="historialContainer" class="mt-4 border-top pt-3" style="display: none;">
67             <h6 class="text-center">Historial de Operaciones</h6>
68             <ul id="historialList" class="list-group list-group-flush">
69                 <li>
70                     </li>
71             </ul>
72         </div>
73     `;
74
75     this.attachEventListeners();
76 }
77
78 attachEventListeners() {
79     //Elementos del Shadow DOM
80     const numero1Input = this.shadowRoot.getElementById('numero1');
81     const numero2Input = this.shadowRoot.getElementById('numero2');
82     const operacionSelect = this.shadowRoot.getElementById('operacion');
83     const calcularButton = this.shadowRoot.getElementById('calcular');
84
85     const resultadoDiv = this.shadowRoot.getElementById('resultado');
86     const errorMessageDiv = this.shadowRoot.getElementById('errorMessage');
87     const historialContainer = this.shadowRoot.getElementById('historialContainer');
88     const historialList = this.shadowRoot.getElementById('historialList');
89
90     calcularButton.addEventListener('click', () => {
91         errorMessageDiv.textContent = '';
92         resultadoDiv.textContent = '';
93         resultadoDiv.classList.remove('alert-danger');
94
95         const num1 = parseFloat(numero1Input.value);
96         const num2 = parseFloat(numero2Input.value);
97         const operacion = operacionSelect.value;
98
99         // Validaciones
100         if (isNaN(num1) || isNaN(num2)) {
101             errorMessageDiv.textContent = 'Por favor, ingrese números válidos en ambos campos.';
102             resultadoDiv.classList.add('alert-danger');
103             return;
104         }
105
106         let resultado;
107         let operacionSimbolo;
108
109         switch (operacion) {
110             case 'suma':
111                 resultado = num1 + num2;
112                 operacionSimbolo = '+';

```

```

111         operacionSimbolo = '-';
112         break;
113     case 'resta':
114         resultado = num1 - num2;
115         operacionSimbolo = '-';
116         break;
117     case 'multiplicacion':
118         resultado = num1 * num2;
119         operacionSimbolo = '*';
120         break;
121     case 'division':
122         if (num2 === 0) {
123             errorMessageDiv.textContent = 'Error: No se puede dividir por cero.';
124             resultadoDiv.classList.add('alert-danger');
125             return;
126         }
127         resultado = num1 / num2;
128         operacionSimbolo = '/';
129         break;
130     default:
131         errorMessageDiv.textContent = 'Operación no válida.';
132         resultadoDiv.classList.add('alert-danger');
133         return;
134 }
135
136 resultadoDiv.textContent = resultado;
137 resultadoDiv.classList.remove('alert-info');
138 resultadoDiv.classList.add('alert-success');

```

```

139
140     //Agrega al historial
141     const operacionTexto = `${num1} ${operacionSimbolo} ${num2} = ${resultado}`;
142     this.historial.push(operacionTexto);
143     this.actualizarHistorial(historialList, historialContainer);
144
145
146     this.dispatchEvent(new CustomEvent('calculado-realizado', {
147         detail: {
148             numero1: num1,
149             numero2: num2,
150             operacion: operacion,
151             resultado: resultado,
152             timestamp: new Date().toISOString()
153         },
154         bubbles: true,
155         composed: true
156     }));
157 });
158 }
159
160
161 actualizarHistorial(listElement, containerElement) {
162     listElement.innerHTML = '';
163     if (this.historial.length > 0) {
164         containerElement.style.display = 'block';
165         this.historial.forEach((item, index) => {
166             const listItem = document.createElement('li');
167
168             listItem.classList.add('list-group-item');
169             listItem.textContent = `${index + 1}. ${item}`;
170             listElement.appendChild(listItem);
171         });
172     } else {
173         containerElement.style.display = 'none';
174     }
175 }
176
177 // Definimos el custom element
178 customElements.define('calculadora-basica', CalculadoraBasica);

```

Script.js

3. PROYECTO EJECUTANDO

CALCULADORA

Calculadora

Número 1:
6

Número 2:
5

Operación:
Multiplicación (*)

Calcular

Resultado:
30

CALCULADORA

Calculadora

Número 1:
3

Número 2:
9

Operación:
Suma (+)

Calcular

Resultado:
12

Historial de Operaciones

1. $6 * 5 = 30$
2. $3 + 9 = 12$

División por cero

CALCULADORA

Calculadora

Número 1:

6

Número 2:

0

Operación:

División (/)



Calcular

Resultado:

Error: No se puede dividir por cero.