

script\codigo2.js

```
360 // CALCULADORA
361
362 // Declaración de variables para almacenar los valores y operaciones
363 let operacionActual = ""; // Almacena la cadena de la operación en curso
364 let resultado = null; // Almacena el resultado de la última operación
365
366 // Función para actualizar el display con la operación actual o el resultado
367 function actualizarDisplay() {
368     let valorDisplay = operacionActual;
369     // Si no hay una operación en curso, pero hay un resultado, muestra el resultado
370     if (valorDisplay === "" && resultado !== null) {
371         valorDisplay = formatearNumero(resultado);
372     } else if (valorDisplay !== "") {
373         // Si hay una operación en curso, muestra la operación formateada
374         valorDisplay = formatearOperacion(operacionActual);
375     }
376     // Establece el valor del display
377     document.getElementById("display").value = valorDisplay;
378 }
379
380 // Función para formatear el número con separadores de miles y decimales
381 function formatearNumero(numero) {
382     // Utiliza Intl.NumberFormat para formatear números según el estándar alemán
383     return new Intl.NumberFormat("de-DE", { maximumFractionDigits: 20 }).format(
384         numero
385     );
386 }
387
388 // Función para formatear la operación, manteniendo operadores intactos
389 function formatearOperacion(operacion) {
390     // Divide la operación en números y operadores
391     return (
392         operacion
393         .split(/([\+\-\*\\/\(\)])/)
394         // Formatea cada número manteniendo los operadores como están
395         .map(elem =>
396             isNaN(elem) || elem === "" ? elem : formatearNumero(Number(elem))
397         )
398         .join("")
399     ); // Une de nuevo todos los elementos formateados
400 }
401
402 // Función para añadir un número o símbolo a la operación
403 function agregarASecuencia(caracter) {
404     // Si hay un resultado y se inicia una nueva operación, usa ese resultado
405     if (
406         resultado !== null &&
407         (operacionActual === "" || isNaN(operacionActual))
408     ) {
409         operacionActual = resultado.toString();
410         resultado = null;
411     }
412     // Añade el nuevo carácter a la operación actual
413     operacionActual += caracter;
414     // Actualiza el display
415     actualizarDisplay();
416 }
```

```

417
418 // Función para calcular el resultado de la operación
419 function calcular() {
420     // Verifica si hay una operación para calcular
421     if (operacionActual !== "") {
422         try {
423             // Realiza el cálculo utilizando eval y maneja errores
424             resultado = eval(operacionActual.replace(/,/g, ""));
425             operacionActual = "";
426         } catch (error) {
427             resultado = "Error";
428         }
429         // Resetea la operación actual y actualiza el display
430         actualizarDisplay();
431     }
432 }
433
434 // Función para limpiar la calculadora y empezar de nuevo
435 function limpiar() {
436     operacionActual = "";
437     resultado = null;
438     actualizarDisplay();
439 }
440
441 // Función para borrar el último carácter ingresado o retroceder en el resultado
442 function borrar() {
443     // Verifica si hay una operación en curso
444     if (operacionActual !== "") {
445         // Borra el último carácter de la operación actual
446         operacionActual = operacionActual.slice(0, -1);
447     } else if (resultado !== null) {
448         // Si no hay operación en curso, pero hay un resultado, permite retroceder en el resultado
449         let resultadoStr = resultado.toString();
450         if (resultadoStr !== "Error") {
451             // Convierte el resultado a string y borra el último carácter
452             resultadoStr = resultadoStr.slice(0, -1);
453             // Convierte de nuevo a número y actualiza el resultado
454             resultado = resultadoStr === "" ? null : Number(resultadoStr);
455         } else {
456             // Si el resultado es "Error", simplemente limpia el resultado
457             resultado = null;
458         }
459     }
460     // Actualiza el display
461     actualizarDisplay();
462 }
463
464 // Funciones para manejar los botones de la interfaz
465 function presionarNumero(numero) {
466     agregarASecuencia(numero);
467 }
468
469 function presionarOperador(operador) {
470     agregarASecuencia(operador);
471 }
472
473 // Evento para manejar la entrada del teclado
474 document.addEventListener("keydown", function (event) {
475     // Captura los eventos de teclado y llama a las funciones correspondientes
476     if ("0123456789+*-/().".includes(event.key)) {

```

```
477     agregarASecuencia(event.key);
478 } else if (event.key === "Enter" || event.key === "=") {
479     event.preventDefault();
480     calcular();
481 } else if (event.key === "Backspace") {
482     borrar();
483 } else if (event.key === "Escape") {
484     limpiar();
485 }
486 });
487
488 // FIN CALCULADORA
```

```

146 <h5 id="calculadora" class="titulo-practica">CALCULADORA</h5>
147
148 <!-- Estructura principal de la Calculadora -->
149 <div id="calculadora">
150   <!-- Display donde se muestra la entrada y el resultado -->
151   <input type="text" id="display" disabled>
152
153   <!-- Primera fila: números 7, 8, 9 y operador de división -->
154   <div class="fila">
155     <button onclick="presionarNumero('7')">7</button>
156     <button onclick="presionarNumero('8')">8</button>
157     <button onclick="presionarNumero('9')">9</button>
158     <button onclick="presionarOperador('/')">/</button>
159   </div>
160
161   <!-- Segunda fila: números 4, 5, 6 y operador de multiplicación -->
162   <div class="fila">
163     <button onclick="presionarNumero('4')">4</button>
164     <button onclick="presionarNumero('5')">5</button>
165     <button onclick="presionarNumero('6')">6</button>
166     <button onclick="presionarOperador('*')">*</button>
167   </div>
168
169   <!-- Tercera fila: números 1, 2, 3 y operador de resta -->
170   <div class="fila">
171     <button onclick="presionarNumero('1')">1</button>
172     <button onclick="presionarNumero('2')">2</button>
173     <button onclick="presionarNumero('3')">3</button>
174     <button onclick="presionarOperador('-')">-</button>
175   </div>
176
177   <!-- Cuarta fila: número 0, punto decimal, operador de suma y botón de igual -->
178   <div class="fila">
179     <button onclick="presionarNumero('0')">0</button>
180     <button onclick="presionarNumero('.')">.</button>
181     <button onclick="presionarOperador('+')">+</button>
182     <button onclick="calcular()">=</button>
183   </div>
184
185   <!-- Quinta fila: botones para paréntesis, borrar y limpiar -->
186   <div class="fila">
187     <button onclick="presionarOperador('(')">(</button>
188     <button onclick="presionarOperador(')')">)</button>
189     <button onclick="borrar()">←</button>
190     <button onclick="limpiar()">C</button>
191   </div>
192 </div>

```