

Initial Diagnosis Report.

BEFORE

```
1 // ----- SCRIPT CALCULADORA LEGACY v1.2 -----  
2 // NO TOCAR NADA - FUNCIONA (A VECES)  
3 var buffer = "0";  
4 var memoria = 0;  
5 var ultimo_operador;  
6 function handleNumber(numStr) {  
7   if (buffer === "0") {  
8     buffer = numStr;  
9   } else {  
10    buffer += numStr;  
11  }  
12  updateScreen();  
13 }  
14 function handleSymbol(symbol) {  
15   switch (symbol) {  
16     case 'C':  
17       buffer = "0";  
18       memoria = 0;  
19       ultimo_operador = null;  
20       break;  
21     case '=':  
22       if (ultimo_operador === null) {  
23         return;  
24       }  
25       flushOperation(parseInt(buffer));  
26       ultimo_operador = null;  
27       buffer = "" + memoria;  
28       memoria = 0;  
29       break;  
30     case '+':  
31     case '-':  
32     case '*':  
33     case '/':  
34       handleMath(symbol);  
35       break;  
36   }  
37   updateScreen();  
38 }  
39 function handleMath(symbol) {  
40   if (buffer === '0' && memoria === 0) {  
41     return;  
42   }  
43   var intBuffer = parseInt(buffer);  
44   if (memoria === 0) {  
45     memoria = intBuffer;  
46   } else {  
47     flushOperation(intBuffer);  
48   }  
49   ultimo_operador = symbol;  
50   buffer = "0";  
51 }  
52 function flushOperation(intBuffer) {  
53   if (ultimo_operador === '+') {  
54     memoria += intBuffer;  
55   } else if (ultimo_operador === '-') {  
56     memoria -= intBuffer;  
57   } else if (ultimo_operador === '*') {  
58     memoria *= intBuffer;  
59   } else if (ultimo_operador === '/') {  
60     memoria /= intBuffer;  
61   }  
62 }  
63 function updateScreen(){  
64   var laPantalla = document.getElementById("display");  
65   laPantalla.innerText = buffer;  
66 }  
67 // INICIALIZADOR DE BOTONES  
68 function init(){
```

← → ↻ eslint-online-playground.netlify.app/#eNqVvW1v0zAQ/unfKDptqaDj0OTyLKCKpUUrQMICE/uct28OXZwnLly7b9zjtMmaTclqpx7+65e+7F0ev1OKYH1heC...

ESLint Online Playground

More Plugins More Examples

src
example.js
eslint.config.js
package.json

calculador.js

```
1  
2 // ----- SCRIPT CALCULADORA LEGACY v1.2 -----  
3 // NO TOCAR NADA - FUNCIONA (A VECES)  
4 var buffer = "0";  
5 var memoria = 0;  
6 var ultimo_operador;  
7 function handleNumber(numStr) {  
8   if (buffer === "0") {  
9     buffer = numStr;  
10  } else {  
11    buffer += numStr;  
12  }  
13  updateScreen();  
14 }  
15 function handleSymbol(symbol) {  
16   switch (symbol) {  
17     case 'C':  
18       buffer = "0";  
19       memoria = 0;  
20       ultimo_operador = null;  
21       break;  
22     case '=':  
23       if (ultimo_operador === null) {  
24         return;  
25       }  
26       flushOperation(parseInt(buffer));  
27       ultimo_operador = null;  
28       buffer = "" + memoria;  
29       memoria = 0;  
30       break;  
31     case '+':  
32     case '-':  
33     case '*':  
34     case '/':  
35       handleMath(symbol);  
36       break;  
37   }  
38   updateScreen();  
39 }  
40 function handleMath(symbol) {  
41   if (buffer === '0' && memoria === 0) {  
42     return;  
43   }  
44   var intBuffer = parseInt(buffer);  
45   if (memoria === 0) {  
46     memoria = intBuffer;  
47   } else {  
48     flushOperation(intBuffer);  
49   }  
50   ultimo_operador = symbol;  
51   buffer = "0";  
52 }  
53 function flushOperation(intBuffer) {  
54   if (ultimo_operador === '+') {  
55     memoria += intBuffer;  
56   } else if (ultimo_operador === '-') {  
57     memoria -= intBuffer;  
58   } else if (ultimo_operador === '*') {  
59     memoria *= intBuffer;  
60   } else if (ultimo_operador === '/') {  
61     memoria /= intBuffer;  
62   }  
63 }  
64 function updateScreen(){  
65   var laPantalla = document.getElementById("display");  
66   laPantalla.innerText = buffer;  
67 }  
68 function init(){
```

Preview Apply Fix

PROBLEMS

- Strings must use doublequote. (quotes) [17:6-9]
- Strings must use doublequote. (quotes) [22:6-9]
- Strings must use doublequote. (quotes) [31:6-9]
- Strings must use doublequote. (quotes) [32:6-9]
- Strings must use doublequote. (quotes) [33:6-9]
- Strings must use doublequote. (quotes) [34:6-9]
- Strings must use doublequote. (quotes) [41:16-19]
- Strings must use doublequote. (quotes) [54:25-28]
- Strings must use doublequote. (quotes) [56:32-35]
- Strings must use doublequote. (quotes) [58:32-35]
- Strings must use doublequote. (quotes) [60:33-36]

netlify

AFTER

```
1 // ----- SCRIPT CALCULADORA LIBRARY v1.2 -----
2 // NO TOCAR NADA - FUNCIONA (A VECE)
3 var buffer = "0";
4 var memoria = 0;
5 var ultimo_operador;
6 function handleNumber(numStr) {
7   if (buffer === "0") {
8     buffer = numStr;
9   } else {
10    buffer += numStr;
11  }
12  updateScreen();
13 }
14 function handleSymbol(symbol) {
15   switch (symbol) {
16     case "+":
17       buffer = "0";
18       memoria = 0;
19       ultimo_operador = null;
20       break;
21     case "-":
22       if (ultimo_operador === null) {
23         return;
24       }
25       flushOperation(parseFloat(buffer));
26       ultimo_operador = null;
27       buffer = "" + memoria;
28       memoria = 0;
29       break;
30     case "*":
31     case "/":
32       if (ultimo_operador === null) {
33         return;
34       }
35       flushOperation(parseFloat(buffer));
36       ultimo_operador = symbol;
37       buffer = "0";
38     }
39   }
40   function flushOperation(intBuffer) {
41     if (ultimo_operador === "+") {
42       memoria += parseInt(intBuffer);
43     } else if (ultimo_operador === "-") {
44       memoria -= parseInt(intBuffer);
45     } else if (ultimo_operador === "*") {
46       memoria *= parseInt(intBuffer);
47     } else if (ultimo_operador === "/") {
48       memoria /= parseInt(intBuffer);
49     }
50   }
51   function updateScreen(){
52     var document;
53     var laPantalla = document.getElementById("display");
54     laPantalla.innerText = buffer;
55   }
56 }
57 // INICIALIZADOR DE BOTONES
58 function init(){
59   var document;
60   var console;
61   console.log("Calculadora inicializada...");
62   document.querySelector(".buttons").addEventListener("click", function(event) {
63     buttonClick(event.target.innerText);
64   });
65 }
66 function buttonClick(value) {
67   if (isNaN(parseFloat(value))) {
68     handleSymbol(value);
69   } else {
70     handleNumber(value);
71   }
72 }
73 init();
```

ESLint Online Playground [More Plugins](#) [More Examples](#)

calculator.js

```
1 // INICIALIZADOR DE BOTONES
2 function init(){
3   var document;
4   var console;
5   console.log("Calculadora inicializada...");
6   document.querySelector(".buttons").addEventListener("click", function(event) {
7     buttonClick(event.target.innerText);
8   });
9 }
10 function buttonClick(value) {
11   if (isNaN(parseFloat(value))) {
12     handleSymbol(value);
13   } else {
14     handleNumber(value);
15   }
16 }
17 init();
```

PROBLEMS CONSOLE

ESLint

ESLint is a static code analysis tool (known as a linter) for JavaScript and TypeScript. Its main objective is to identify and report problems in your code before it runs.

This includes:

Potential Errors: Detects code patterns that are likely errors (e.g., using a variable before defining it).

Good practices: Helps to follow a set of rules and good programming practices.

Style Issues: Can point out formatting inconsistencies (though for automatically correcting formatting, Prettier is more specific).

Prettier

Prettier is a code formatter. Its sole purpose is to rewrite your code to conform to a set of consistent, predefined style rules.

Unlike ESLint (which looks for errors), Prettier only cares about the appearance of the code:

- Adjusts indentation (spaces or tabs).
- Handles line breaks to respect a maximum width.
- Standardizes the use of quotation marks (single or double).
- Add or remove spaces, semicolons, etc., according to its rules.