

CALCULATOR BASIC TEST

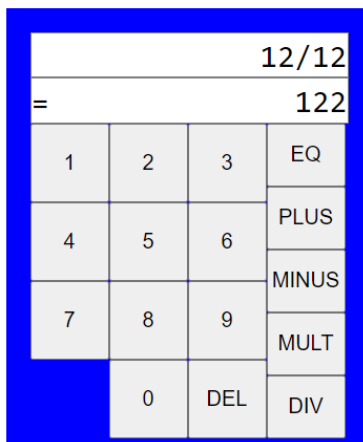
- Verification of all buttons are present, and text written on them is readable – **PASS**
- Check the simple arithmetic operations are working fine +, -, *, / - **PASS**

FUNCTIONAL TEST CASES TESTED

- Check the addition of two number -PASS.
- Check the addition of two negative number -PASS.
- Check the addition of one positive and negative number -PASS.
- Check the subtraction of two integer number -PASS.
- Check the subtraction of one negative and one positive number -PASS.
- Check the subtraction of two negative integer number -PASS.
- Check the multiplication of two integers -PASS.
- Check the multiplication of one negative number and one positive number -PASS.
- Division of two positive one-digit number -PASS

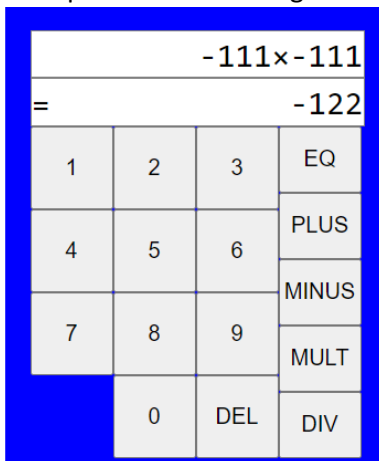
BUGS

- Division only works with one-digit number.



Expected=1

- Multiplication of two negative number.



Expected 12321 or 'ERR'.

- Multiplication of 11 end up in showing **wrong value**.

11×11			
= 111			
1	2	3	EQ
4	5	6	PLUS
7	8	9	MINUS
			MULT
	0	DEL	DIV

Expected **121**

- Division by large number

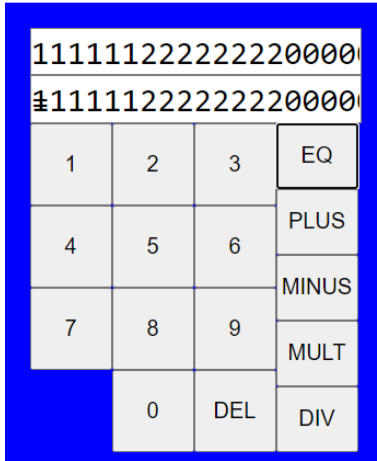
2/123456789			
= 223456789			
1	2	3	EQ
4	5	6	PLUS
7	8	9	MINUS
			MULT
	0	DEL	DIV

Expected 1.6

- Test with random operator multiple time. Instead of showing **ERR** it showing **1**.

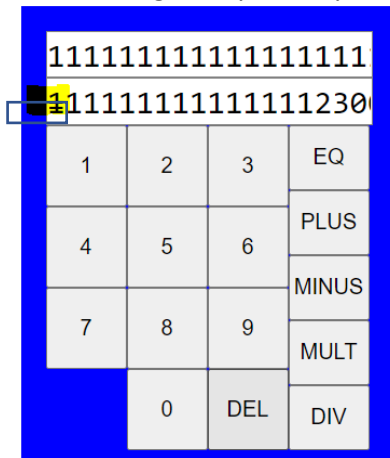
-1×1+2////8+1			
= 1			
1	2	3	EQ
4	5	6	PLUS
			MINUS
7	8	9	MULT
	0	DEL	DIV

- After entering 17 numbers the numbers are not moving forward.

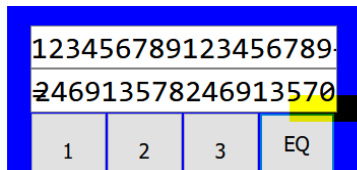


DESIGN ISSUES

- Numbers are not separated by comma.
- Over writing of = symbol by 1



- The end of the EQ button and result area is not aligned properly.



- Multiplication of two large integers $123456789 \times 210987654000 = 26047858281483006000$

The screenshot shows a web browser with the URL `outputjsbin.com/hudape/1/`. On the left is a calculator interface with a display showing `123456789x21098765-260478582814830060`. The calculator has buttons for digits 0-9, `DEL`, `DIV`, `MULT`, `MINUS`, `PLUS`, `EQ`, and a `Wappalizer` button. On the right is the browser's developer console, showing the following JavaScript code and its output:

```

< >
let input=document.getElementById('expression').value
< undefined
input
< "123456789x210987654000"
let output=document.querySelector('div#output').innerText
< undefined
output
< "26047858281483006000"
123456789x210987654000
26047858281483006000

```

Problem: Result is not completely **visible**; the output textbox cannot be moved.

SECURITY

- Usage of **EVAL ()**

The screenshot shows a web browser with the URL `outputjsbin.com/hudape/1/`. On the left is a calculator interface with a display showing `2/2+3x2-1` and the result `6`. The calculator has buttons for digits 0-9, `DEL`, `DIV`, `MULT`, `MINUS`, `PLUS`, `EQ`, and a `Wappalizer` button. On the right is the browser's developer tools, showing the source code for the calculator. The code is as follows:

```

113 c = $("#expression.display");
114 d = $("#output.display");
115 $(".inputs").delegate("numeric",
116   var a = $(this).val();
117   b = a;
118   c.val(b);
119   d.text("");
120 });
121 $(".inputs").delegate("operator",
122   var a = $(this).val();
123   switch (a) {
124     case "+":
125       break;
126     default:
127       b += a, c.val(b), d.
128   }
129 });
130 $(".inputs").delegate("[value='<
131 b = b.substr(0, b.length - 1
132 c.val(b);
133 d.text("");
134 });
135 $(".inputs").delegate("[value='<
136   try {
137     var a = c.val();
138     a = a.replace(/u0007/g,
139     a = a.replace(/\\s/, ""
140     a = a.replace(/\\s/, ""
141     var f = eval(a);
142     d.text(f);
143   } catch (e) {
144     debugger;
145     d.text("ERR")
146   }
147 })
148 });
149 </script>
150

```

TEST CASES

ADD

```
Running: Test 1 addition of two positive integer number [111+111=222]
✓ Element <div[contains(text(), "222")]> was visible after 16 milliseconds.
OK. 1 assertions passed. (885ms)
Running: Test 2 addition of two negative number [-111+-111=-222]
✓ Element <div[contains(text(), "-222")]> was visible after 7 milliseconds.
OK. 1 assertions passed. (447ms)
Running: Test 3 addition of one negative and one positive number [-111+111=0]
✓ Element <div[contains(text(), "0")]> was visible after 9 milliseconds.
OK. 1 assertions passed. (381ms)
Running: Test 4 addition of two ten digit number [123456789+123456789=246913578]
✓ Element <div[contains(text(), "246913578")]> was visible after 13 milliseconds.
OK. 1 assertions passed. (4.478s)
Running: Test 5 addition of two fifteen digit number [123456789+123456789=246913578]
✓ Element <div[contains(text(), "24691357824690")]> was visible after 14 milliseconds.
OK. 1 assertions passed. (6.583s)
```

SUBTRACT

```
Running: Test 1 subtraction of two positive integer number [111-111=0]
✓ Element <div[contains(text(), "0")]> was visible after 18 milliseconds.
OK. 1 assertions passed. (1.485s)
Running: Test 2 subtraction of one negative and positive integer [-111-111=0]
✓ Element <div[contains(text(), "-222")]> was visible after 9 milliseconds.
OK. 1 assertions passed. (519ms)
Running: Test 3 subtraction of two negative integer [-111-0-111=0]
✓ Element <div[contains(text(), "-222")]> was visible after 17 milliseconds.
OK. 1 assertions passed. (2.895s)
Running: Test 4 subtraction of two ten digit number [123456789+123456789=246913578]
✓ Element <div[contains(text(), "0")]> was visible after 19 milliseconds.
OK. 1 assertions passed. (4.971s)
Running: Test 5 subtraction of two fifteen digit number [123456789+123456789=246913578]
✓ Element <div[contains(text(), "0")]> was visible after 19 milliseconds.
OK. 1 assertions passed. (7.26s)
```

MULTIPLICATION

```
Using: firefox (83.0) on windows 10.0 platform.

Running: Test 1 multiplication of two integer [222,222=0]

✓ Element </div[contains(text(), "49284")]> was visible after 33 milliseconds.

OK. 1 assertions passed. (3.92s)
Running: Test 2 multiplication of one negative and one positive integer [-111,111=0]

✓ Element </div[contains(text(), "-49284")]> was visible after 16 milliseconds.

OK. 1 assertions passed. (1.278s)
Running: Test 3 Multiplication of a number by 0 [1234567890*0=0]

✓ Element </div[contains(text(), "0")]> was visible after 21 milliseconds.

OK. 1 assertions passed. (3.375s)
Running: Test 4 Multiplication of a number by fraction on the right side [2/3*123456789]

✓ Element </div[contains(text(), "82304526")]> was visible after 17 milliseconds.

OK. 1 assertions passed. (3.615s)
Running: Test 5 Multiplication of a number by fraction on the left side [2/3*123456789]

✓ Element </div[contains(text(), "82304526")]> was visible after 17 milliseconds.

OK. 1 assertions passed. (3.568s)
Running: Test 6 Result of a previous operation when the result is 0 [0*6=0]

✓ Element </div[contains(text(), "0")]> was visible after 16 milliseconds.

OK. 1 assertions passed. (1.845s)

OK. 6 total assertions passed (23.673s)
```

DIVISION

```
Running: Test 1 division of two positive integer [222,222=0]

✓ Element </div[contains(text(), "1")]> was visible after 30 milliseconds.

OK. 1 assertions passed. (1.336s)
Running: Test 2 division of two negative integers[-222,-222=0]

✓ Element </div[contains(text(), "1")]> was visible after 18 milliseconds.

OK. 1 assertions passed. (592ms)
Running: Test 3 division of one negative integer and one positive integer[-2,2=0]

✓ Element </div[contains(text(), "-1")]> was visible after 46 milliseconds.

OK. 1 assertions passed. (647ms)
Running: Test 4 divide 0 by a integer divisor [0/200=0]

✓ Element </div[contains(text(), "0")]> was visible after 20 milliseconds.

OK. 1 assertions passed. (1.975s)
Running: Test 5 divide a negative dividend by a positive divisor [-1500/2000=-0.75]

✓ Element </div[contains(text(), "-0.75")]> was visible after 18 milliseconds.

OK. 1 assertions passed. (3.222s)
Running: Test 6 division by 0 [1500/0=ERR]

✓ Element </div[contains(text(), "Infinity")]> was visible after 16 milliseconds.

OK. 1 assertions passed. (2.441s)
Running: Test 7 division by two floating numbers[1/2/1/5]

✓ Element </div[contains(text(), "0.1")]> was visible after 17 milliseconds.

OK. 1 assertions passed. (2.232s)
```

EXPRESSION

```
[Expression\expression] Test Suite
=====
i Connected to localhost on port 4444 (4353ms).
  Using: firefox (85.0) on windows 10.0 platform.

Running: Test 1 expression [2/2+3×2-1=6]

✓ Element <div[contains(text(), "6")]> was visible after 19 milliseconds.

OK. 1 assertions passed. (1.182s)
Running: Test 2 expression

No assertions ran.
```