

Github Weblink

Vercel

W01-p1: 取得畫面輸入，4個按鈕、2個輸出，共7個。透過console log 印出。

The screenshot displays a web browser window showing a calculator application titled "The Unconventional Calculator". The calculator interface includes a display showing "0" and "Result: 0". Below the display are four buttons: "+", "-", "*", and "/". The browser's developer tools are open, showing the console log with the following output:

```
User Input: ' ' app.js:21
Plus: ' ' app.js:22
Minus: ' ' app.js:23
Times: ' ' app.js:24
Divide: ' ' app.js:25
resultFormula: ' ' app.js:26
resultNumber: ' ' app.js:27
```

The source code in the background shows the following JavaScript logic:

```
1  Variables define.
2
3  const userInput = document.querySelector('#input-number');
4
5  /* Divisors Variables */
6  const divisorPlus = document.querySelector('#btn-add');
7  const divisorMinus = document.querySelector('#btn-subtract');
8  const divisorTimes = document.querySelector('#btn-multiply');
9  const divisorDivide = document.querySelector('#btn-divide');
10
11 /* Results */
12 // When you complete your calculation, The result would be right here.
13 const resultFormula = document.querySelector('#current-calculation');
14 const resultNumber = document.querySelector('#current-result');
15
16
17 /* Log part.
18 */
19 console.log('User Input: ', userInput);
20 console.log('Plus: ', divisorPlus);
21 console.log('Minus: ', divisorMinus);
22 console.log('Times: ', divisorTimes);
23 console.log('Divide: ', divisorDivide);
24 console.log('resultFormula: ', resultFormula);
25 console.log('resultNumber: ', resultNumber);
```

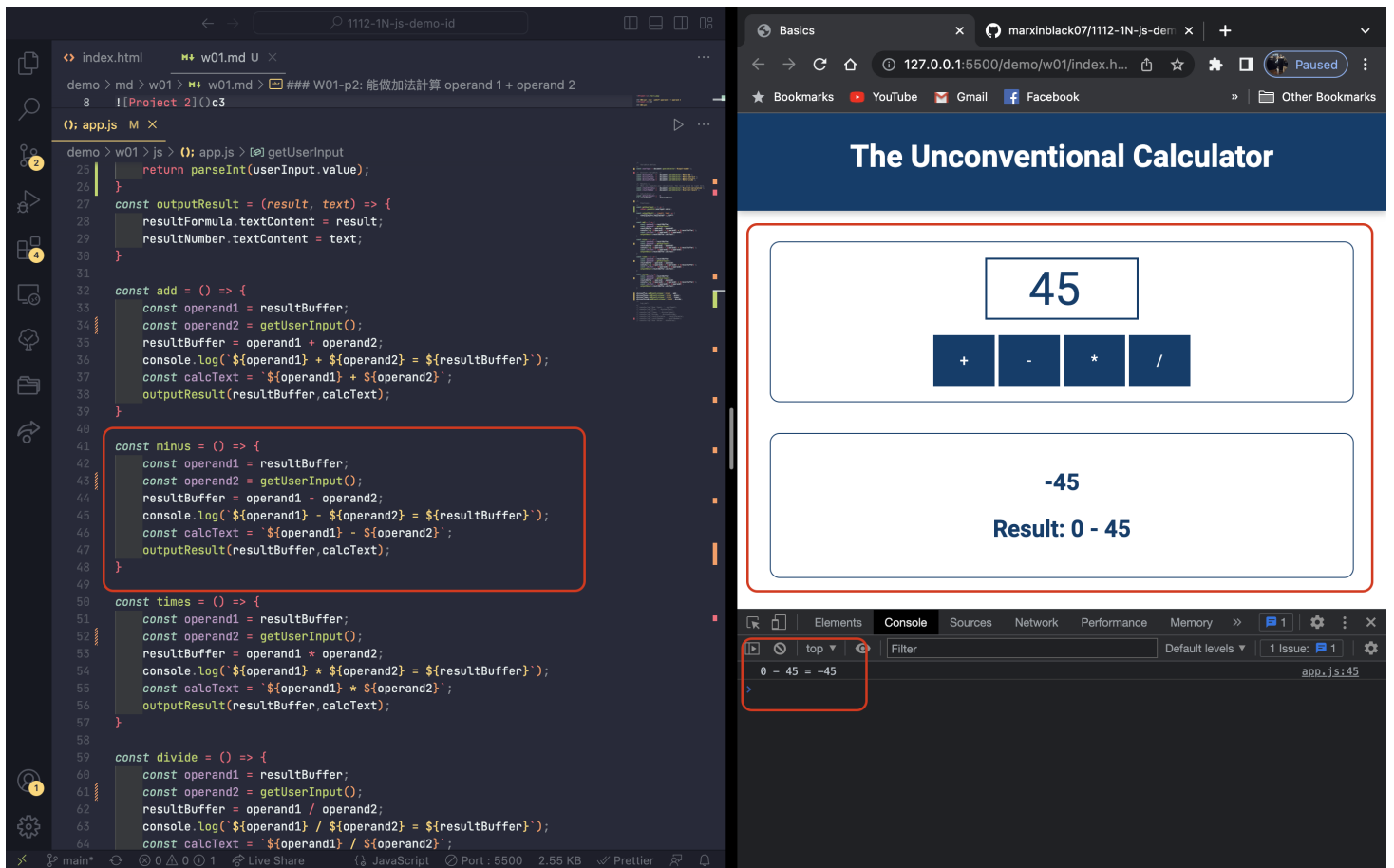
W01-p2: 能做加法計算 operand 1 + operand 2

The screenshot displays a web application titled "The Unconventional Calculator" running in a browser. The application's interface includes a large input field containing the number "12", a row of operation buttons (+, -, *, /), and a result display showing "12" and "Result: 0 + 12".

The source code for the application is visible on the left, showing the JavaScript logic. The `add` function is highlighted, which takes the current value in the result buffer and the user input, calculates the sum, and updates the display. The code also includes functions for `getUserInput`, `outputResult`, `minus`, and `times`.

The browser's console on the right shows the output of the `add` function: `0 + 12 = 12`.

W01-p3:能做減法計算 operand 1 - operand 2



W01-p4:能做乘法計算 operand 1 * operand 2

The image displays a web application titled "The Unconventional Calculator" running in a browser. The calculator interface shows a display with the number 72, a row of buttons for addition (+), subtraction (-), multiplication (*), and division (/), and another display showing 0. Below this, the text "Result: 0 * 72" is visible. The browser's console shows the output of the JavaScript code: "0 * 72 = 0".

The JavaScript code in the background is as follows:

```
demo > md > w01 > js > 0: app.js > @: getUserInput
const add = () => {
  33 const operand1 = resultBuffer;
  34 const operand2 = getUserInput();
  35 resultBuffer = operand1 + operand2;
  36 console.log(`${operand1} + ${operand2} = ${resultBuffer}`);
  37 const calcText = `${operand1} + ${operand2}`;
  38 outputResult(resultBuffer, calcText);
  39 }
40
41 const minus = () => {
  42 const operand1 = resultBuffer;
  43 const operand2 = getUserInput();
  44 resultBuffer = operand1 - operand2;
  45 console.log(`${operand1} - ${operand2} = ${resultBuffer}`);
  46 const calcText = `${operand1} - ${operand2}`;
  47 outputResult(resultBuffer, calcText);
  48 }
49
50 const times = () => {
  51 const operand1 = resultBuffer;
  52 const operand2 = getUserInput();
  53 resultBuffer = operand1 * operand2;
  54 console.log(`${operand1} * ${operand2} = ${resultBuffer}`);
  55 const calcText = `${operand1} * ${operand2}`;
  56 outputResult(resultBuffer, calcText);
  57 }
58
59 const divide = () => {
  60 const operand1 = resultBuffer;
  61 const operand2 = getUserInput();
  62 resultBuffer = operand1 / operand2;
  63 console.log(`${operand1} / ${operand2} = ${resultBuffer}`);
  64 const calcText = `${operand1} / ${operand2}`;
  65 outputResult(resultBuffer, calcText);
  66 }
67
68
69 divisorPlus.addEventListener('click', add);
70 divisorMinus.addEventListener('click', minus);
71 divisorTimes.addEventListener('click', times);
```

W01-p5:能做除法計算 operand 1 / operand 2

The image displays a web application titled "The Unconventional Calculator" running in a browser. The calculator interface shows the number 87 in the display, with buttons for addition (+), subtraction (-), multiplication (*), and division (/). Below the display, the text "0" and "Result: 0 / 87" are visible. The browser's console shows the output of the division operation: "0 / 87 = 0".

The source code for the application is shown in a code editor. The code defines a function `divide` that calculates the division of two operands. The function is as follows:

```
const divide = () => {
  const operand1 = resultBuffer;
  const operand2 = getUserInput();
  resultBuffer = operand1 / operand2;
  console.log(`${operand1} / ${operand2} = ${resultBuffer}`);
  const calcText = `${operand1} / ${operand2}`;
  outputResult(resultBuffer, calcText);
}
```

The code also includes event listeners for the division button, which calls the `divide` function when clicked.

W01-p6:能做四則計算，加減乘除都執行一遍，可任意順序、結果要正確。

index.htmlw01.md

demo > md > w01 > w01.md > ### W01-p6:能做加法計算

1[[6github WebLink](https://aithub.com/marxinblack07/

0): app.js M X

demo > w01 > js > 0: app.js > [e] getUserInput

31

32const add = () => {

33const operand1 = resultBuffer;

34const operand2 = getUserInput();

35resultBuffer = operand1 + operand2;

36console.log(`\${operand1} + \${operand2} = \${resultBuffer}`);

37const calcText = `\${operand1} + \${operand2}`;

38outputResult(resultBuffer, calcText);

39}

40

41const minus = () => {

42const operand1 = resultBuffer;

43const operand2 = getUserInput();

44resultBuffer = operand1 - operand2;

45console.log(`\${operand1} - \${operand2} = \${resultBuffer}`);

46const calcText = `\${operand1} - \${operand2}`;

47outputResult(resultBuffer, calcText);

48}

49

50const times = () => {

51const operand1 = resultBuffer;

52const operand2 = getUserInput();

53resultBuffer = operand1 * operand2;

54console.log(`\${operand1} * \${operand2} = \${resultBuffer}`);

55const calcText = `\${operand1} * \${operand2}`;

56outputResult(resultBuffer, calcText);

57}

58

59const divide = () => {

60const operand1 = resultBuffer;

61const operand2 = getUserInput();

62resultBuffer = operand1 / operand2;

63console.log(`\${operand1} / \${operand2} = \${resultBuffer}`);

64const calcText = `\${operand1} / \${operand2}`;

65outputResult(resultBuffer, calcText);

66}

67

68

69divisorPlus.addEventListener('click', add);

Basics

marxinblack07/1112-1N-js-demo

127.0.0.1:5500/demo/w01/index.h...

Paused

BookmarksYouTubeGmailFacebookOther Bookmarks

The Unconventional Calculator

50

+ - * /

1

Result: 50 / 50

ElementsConsoleSourcesNetworkPerformanceMemory

Filter

Default levels1 Issue: 1

0 + 20 = 20app.js:36

20 - 10 = 10app.js:45

10 * 5 = 50app.js:54

50 / 50 = 1app.js:63