

W01-P1: 取得畫面輸入，4 個按鈕，兩個輸出，共 7 個，透過 console.log 印出

The screenshot displays a web development environment with three main components:

- VS Code Editor (Left):** Shows the `app.js` file with JavaScript code for a calculator. A red box highlights the `console.log` statements used for debugging: `console.log('userInput', userInput);`, `console.log('addBtn', addBtn);`, `console.log('subtractBtn', subtractBtn);`, `console.log('multiplyBtn', multiplyBtn);`, `console.log('divideBtn', divideBtn);`, `console.log('currentResultOutput', currentResultOutput);`, and `console.log('currentCalculationOutput', currentCalculationOutput);`.
- Browser (Right):** Displays the web application titled "The Unconventional Calculator". The UI includes an input field, four buttons (+, -, *, /), a display showing "0", and a "Result: 0" label. A red box highlights the HTML elements in the DOM: `<input type="number" id="input-number">`, `<button type="button" id="btn-add">+</button>`, `<button type="button" id="btn-subtract">-</button>`, `<button type="button" id="btn-multiply">*</button>`, `<button type="button" id="btn-divide">/</button>`, `0`, and `<h2 id="current-calculation">0</h2>`.
- Browser Console (Bottom Right):** Shows a red error message: "Unchecked runtime.lastError: The message port closed before a response was received." at `index.html:1`.

W01-P2: 能做加法計算 operand1 + operand2

The screenshot shows the development environment for 'The Unconventional Calculator'. On the left, the VS Code editor displays the JavaScript code for the application. A red box highlights the `add` function, which takes the current result and a new operand, calculates their sum, and updates the UI. The code includes comments for logging user input, button clicks, and the current calculation output.

```
const defaultResult = 0;
let currentResult = defaultResult;

function getUserInput() {
  return parseInt(userInput.value);
}

function outputResult(result, text) {
  currentResultOutput.textContent = result;
  currentCalculationOutput.textContent = text;
}

// operand1 op operator operand2 0 + 5
function add() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 + operand2;
  console.log(`${operand1} + ${operand2} = ${currentResult}`);
  const calcText = `${operand1} + ${operand2}`;
  outputResult(currentResult, calcText);
}

addBtn.addEventListener('click', add);
```

On the right, the web browser shows the application running at `127.0.0.1:5502/index.html`. The calculator interface has a display showing '10', a row of buttons including '+', '-', '*', and '/', and a result box showing '5 + 10' and 'Result: 15'. The browser's console shows the log output: `0 + 5 = 5` and `5 + 10 = 15`.

W01-P3: 能做減法計算 operand1 - operand2

The screenshot shows the development environment for 'The Unconventional Calculator' with subtraction functionality. On the left, the VS Code editor displays the JavaScript code. A red box highlights the `sub` function, which calculates the difference between the current result and a new operand. The code includes comments for logging user input, button clicks, and the current calculation output.

```
function sub() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 - operand2;
  console.log(`${operand1} - ${operand2} = ${currentResult}`);
  const calcText = `${operand1} - ${operand2}`;
  outputResult(currentResult, calcText);
}

function mul() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 * operand2;
  console.log(`${operand1} * ${operand2} = ${currentResult}`);
  const calcText = `${operand1} * ${operand2}`;
  outputResult(currentResult, calcText);
}

addBtn.addEventListener('click', add);
subBtn.addEventListener('click', sub);
mulBtn.addEventListener('click', mul);
```

On the right, the web browser shows the application running at `http://127.0.0.1:5500/demo/w01/index.html`. The calculator interface has a display showing '2', a row of buttons including '+', '-', '*', and '/', and a result box showing '-5 - 2' and 'Result: -7'. The browser's console shows the log output: `0 - 5 = -5` and `-5 - 2 = -7`.

W01-P4: 能做乘法計算 $\text{operand1} * \text{operand2}$

The screenshot shows the development environment for 'The Unconventional Calculator'. In Visual Studio Code, the `mul()` function is highlighted in red, showing the logic for multiplying two operands. The browser window displays the calculator interface with the number 10 entered. The multiplication button is highlighted in red, and the result -35 * 10 = -350 is shown. The console log shows the calculation steps.

```
function mul() {  
  const operand1 = currentResult;  
  const operand2 = getUserInput();  
  currentResult = operand1 * operand2;  
  console.log(`${operand1} * ${operand2} = ${currentResult}`);  
  const calcText = `${operand1} * ${operand2}`;  
  outputResult(currentResult, calcText);  
}
```

Browser Console Log:

```
0 - 5 = -5  
-5 - 2 = -7  
-7 * 5 = -35  
-35 * 10 = -350
```

W01-P5: 能做除法計算 $\text{operand1} / \text{operand2}$

The screenshot shows the development environment for 'The Unconventional Calculator'. In Visual Studio Code, the `div()` function is highlighted in red, showing the logic for dividing two operands. The browser window displays the calculator interface with the number 2 entered. The division button is highlighted in red, and the result 0 / 2 = 0 is shown. The console log shows the calculation steps.

```
function div() {  
  const operand1 = currentResult;  
  const operand2 = getUserInput();  
  currentResult = operand1 / operand2;  
  console.log(`${operand1} / ${operand2} = ${currentResult}`);  
  const calcText = `${operand1} / ${operand2}`;  
  outputResult(currentResult, calcText);  
}
```

Browser Console Log:

```
0 / 10 = 0  
0 / 2 = 0
```

W01-P6: 能做四則計算，加減乘除都要執行一遍，可任意順序，結果要正確

The screenshot displays a web browser window with the URL `http://127.0.0.1:5500/demo/w01/index.html`. The page title is "The Unconventional Calculator". The calculator interface includes a display showing "0" and buttons for "+", "-", "*", and "/". Below the display, it shows "5 * 0" and "Result: 0". The background shows the source code in a code editor, with the following functions highlighted:

```
function add() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 + operand2;
  console.log(`${operand1} + ${operand2} = ${currentResult}`);
  const calcText = `${operand1} + ${operand2}`;
  outputResult(currentResult, calcText);
}

function sub() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 - operand2;
  console.log(`${operand1} - ${operand2} = ${currentResult}`);
  const calcText = `${operand1} - ${operand2}`;
  outputResult(currentResult, calcText);
}

function mul() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 * operand2;
  console.log(`${operand1} * ${operand2} = ${currentResult}`);
  const calcText = `${operand1} * ${operand2}`;
  outputResult(currentResult, calcText);
}

function div() {
  const operand1 = currentResult;
  const operand2 = getUserInput();
  currentResult = operand1 / operand2;
  console.log(`${operand1} / ${operand2} = ${currentResult}`);
  const calcText = `${operand1} / ${operand2}`;
  outputResult(currentResult, calcText);
}
```

The console output shows the following log messages:

```
0 / 10 = 0
0 / 2 = 0
0 + 5 = 5
5 + 10 = 15
15 - 10 = 5
5 * 0 = 0
```

logs

```
$ git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2023-02-15"
e88661f yuwen1213 Thu Feb 16 22:59:08 2023 +0800 add github url in w01_40.md
7b78b18 yuwen1213 Thu Feb 16 22:48:06 2023 +0800 W01-P6: 能做四則計算，加減乘除都要執行一遍，可任意順序，結果要正確
abde568 yuwen1213 Thu Feb 16 20:29:13 2023 +0800 Initial commit
```