

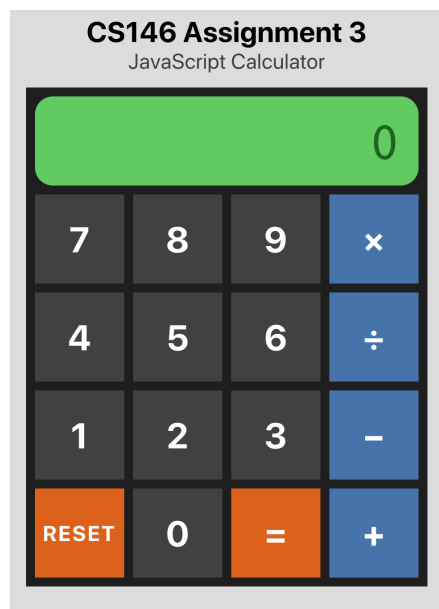
CS-146 Homework 3

JavaScript and the Document Object Model

1 Assignment Objectives

The goal of this assignment is to help you understand how JavaScript and HTML connect to achieve webpage functionality. You will be making a calculator that functions based on the clicking of different DOM elements.

This is what the calculator looks like:



Here is a the gif with the calculator functionality: [Gfycat](#)

2 Setup

For this assignment, you will be editing one file, `solution.js`, which is linked to `index.html`. **DO NOT** edit the HTML in any way, you only need to write the JavaScript functions. In the `solution.js`, you will find skeleton code for the different functions you have to create. You will also be utilizing global variables to manage the state of your calculator. It is recommended to have a global `result` value to hold the current value of the calculator, a global `currentOp` value to represent the operation being used, and a global `buffer` value to hold the previous value to compute with the `result` for each operation.

3 Functions

You are responsible for the following functions:

3.1 `resetCalc()`

This function will reset all values of the calculator, including the current result, the buffer value and the current operation.

3.2 `setDisplay(str)`

This function will find the element that has the `id="output"` property and change the inner text of that element to reflect the current result of the calculator.

3.3 `getResult()`

This function will return the current result of your calculator. (Make sure it's a number, not a string!)

3.4 `pressNum(num)`

This function is attached to every number button element like so:

```
<button onclick="pressNum(1)">1</button>
```

This function will append the current number to the current result value. Make sure to restrict this value to be between the minimum (-999999999) and maximum (999999999).

3.5 `pressOp(op)`

This function is attached to every operation element like so:

```
<button class="operation" onclick="pressOp('-')">&minus;</button>
```

This function will move the current result value to a temporary buffer and set the current result to zero. If a number has not been entered between operations, this function should only change the operation, and keep the same buffer value (not move 0 to the buffer).

3.6 `pressEquals()`

Should compute the operation between the buffer value and current result value based on the current operation. If there is no current operation, do nothing. If a user attempts an invalid operation (such as $0 \div 0$) the calculator should display "ERROR". Also, be sure to once again check that your result value is restricted between the minimum (-999999999) and maximum (999999999).

4 Submission

Upload your `solution.js` file to Canvas as `LastName_FirstName_Assignment3.js`.

Good Luck!