**Project Description/Summary**

This project involves using JavaScript to wire up a simple calculator.

**New Things Learned or Refreshed**

I spent most my time starring at the screen. I got my numbers to show up but I couldn't figure out how to do the math and make it show up. I thought the easiest way was to use the eval() function but I read in the MDN documentation to [**never use eval()**](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/eval#Do_not_ever_use_eval!). So, I got stuck and lost patience. I remember doing a JavaScript calculator project about 5 months ago but it was clear today that that I must have just copied the solution as a code along as Brian Holt typed through it. The code above in nothing like the code for [my previous Apple JavaScript Calculator Project](https://github.com/romeojeremiah/apple_calc). The Apple Calculator handles errors extremely well. It's more on the Intermediate side of the house, if you're up for a challenge. I ended up watching John's solution and implemented his since I had already been at this problem for more than an hour. **Word of caution**, the solution above works to the extent that the user uses the calculator flawlessly. If not, there are many use cases in which the code won't work and will throw errors. For example, if someone starts with an \* operator instead of a number, the eval() function won't work. You'll get an Uncaught SyntaxError: Unexpected token \* at HTMLButtonElement.<anonymous> error.

**Time to Code**

This took about **2 hours**, start to finish, including watching John's solution and then going back to finish my own solution.

**Lines of Code**

This took 30 lines of code, including comments and white space.

**Biggest Take Away(s)**

I couldn't for the life of me remember how I coded my last calculator project. This goes to show that continuous coding and going back to previous problems is just the name of the game.

**Your Turn!**