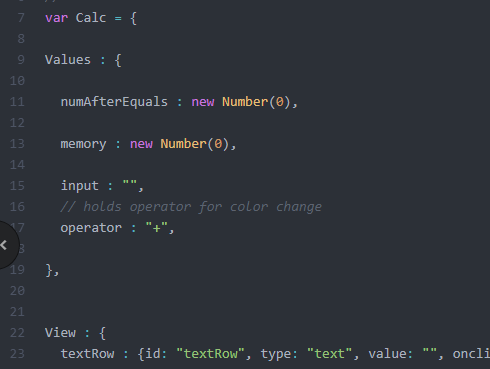
Hw3 Report ---- Com S 319 ---- Richard Smith

**TASK 1:** Calculator.js and Calculator.html

So at first it was easy to get it going based off the given “match.js” file that had code I based my calculator off of. It helped a lot with getting the buttons in the correct rows and assigning all the buttons their functions.

but getting the equal button to work correctly was the hardest part. I used the “eval()” function that JavaScript always has available for simple math. It takes an input, evaluates the numbers and operant, and returns just the answer. Even if negative numbers or numbers with decimals are involved.

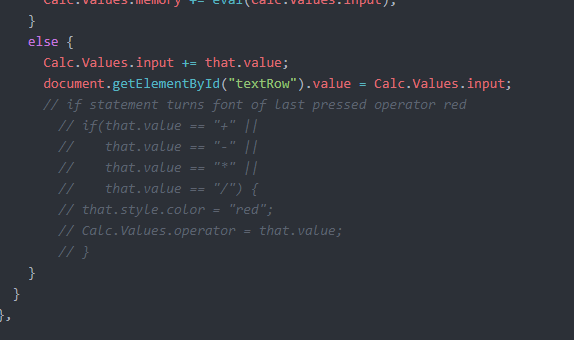
basically, all my code is in calculator.js because the instructor mainly wanted us to use JavaScript. He didn’t want any global variables, so I made a class called “Values” and have objects that hold values and are named based off exactly what they do or are used for.



I spent a ton of time getting the if statements in my main “Controller” to work with all the button handling

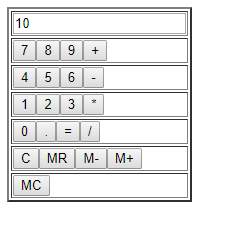
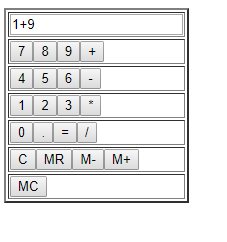


eventually with enough debugging and reloading the calculator page, I got it to work correctly. But unfortunately, I could not get the color to work with the operators. I put info about that in the README.txt for the calculator.



I commented out the code for color change, It worked but I couldn’t find out how to make it go back to back once the equal operant was pressed!

Overall, it was pretty cool coding a real working calculator.



**TASK 2:** snake.html/js

This one didn’t have much skeleton code so I just had to mess around with random code I made until it worked. All my code for javascript and html is in the snake.html.

For snake I have a bunch of global variables used because the instructor said it didn’t matter for this one. So, I used the timer as a while loop and set up a bunch of if statements with this variable I called “state”. the “state” variable decides what direction the snake is going based off of 1, 2, 3, and 4. The “turn left” and “turn right” buttons subtract or add to the state variable thus, changing the direction of the snake I put a good amount of comments in the snake code if you want to take a look. Also, sorry my snake is a little slow and sorry this is late! It was my birthday weekend.

The README.txt explains about how it all runs. Thanks.

