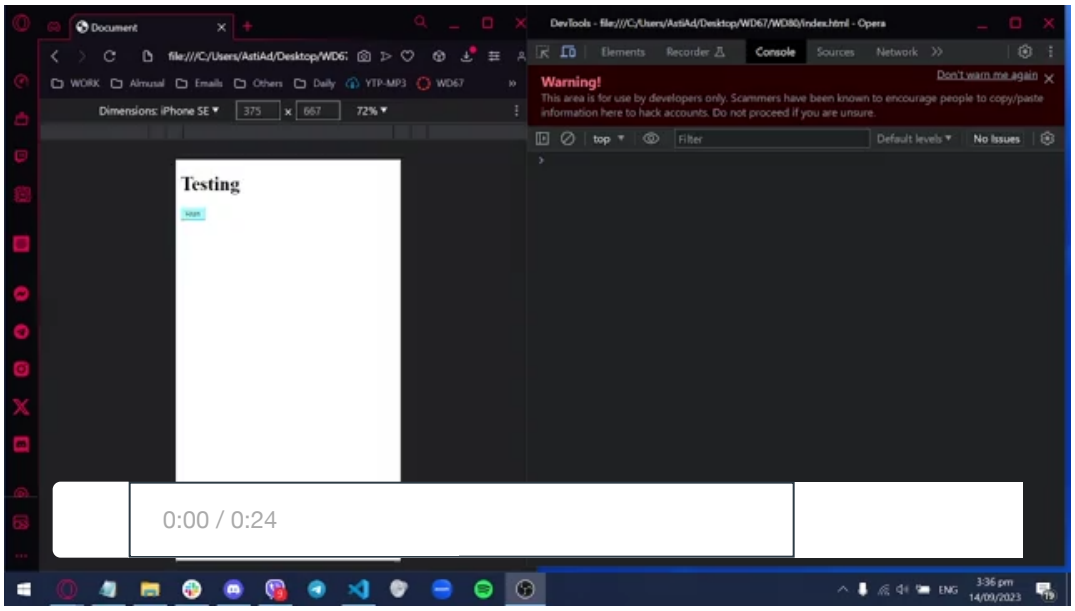
ACT #34 Number Evaluator

New Attempt

Due No Due Date **Points** 40 **Submitting** a file upload **File Types** js

SOFT DEADLINE: 12/1/2023 3:15 PM

HARD DEADLINE: 12/1/2023 3:20 PM



Instruction

Create a program that asks for a set of numbers, and keeps track of the number of even, odd, positive, negative, integer, and non-integer numbers.

Tools

JavaScript, HTML, Visual Studio Code

Description

- **Follow the Submit Your Work steps. (https://kodego.instructure.com/courses/379/pages/7-dot-1-essential-javascript-functions?module_item_id=17474)**
- Name your JavaScript program **counter.js**.
- Using a `for` statement, write a script that does the following:

1. **Prompt** (https://kodego.instructure.com/courses/379/pages/7-dot-1-essential-javascript-functions) the user for a number from 2 to 20. Then, prompt the user for a set of numbers based on the number they specified.
    - For example, the user answers 5 for the first prompt. The program would then ask the user for 5 numbers.
2. Determine whether each number is an even, odd, positive, negative, integer (non-decimal), and decimal number (don't output anything yet).
    - Special cases:
        - One is an odd number.
        - Zero is neither an even, odd, positive, nor negative number.
        - Zero is an integer.
        - A number like 3.00 is not a decimal number, but a number like 3.01 is.
            - You **don't** need a special function for this!
        - Decimal numbers can neither be even nor odd.
        - Negative numbers can still be even or odd.
    - Some number can be two classifications at once i.e. the number 50 is an even number, a positive number, and an integer.
    - Do **not** use any methods that we have not discussed yet (you don't need them). Stick with the fundamentals.
        - Specifically, methods such as `isInteger()` are not allowed to be used.
        - Check the Notes and Tips section below for an idea of how you can do your checks.
3. Once all numbers have been entered, output the total number of numbers inputted, even, odd, positive, negative, integer, and decimal numbers.
    - The final output would look something like this:

```
Number count: 10
Even numbers: 3
Odd numbers: 7
Positive numbers: 8
Negative numbers: 2
Integers: 9
Decimal numbers: 1
```

- Submit your JavaScript file **only** here.

Validation

- Aside from the requirements stated above, the program should also display an error message in the case of...
    - The user entering a number that isn't 2 to 20 at the first prompt.

Notes and Tips

- You can use modulo (%) to check whether a number divided by another number yields a remainder. For example, `7 % 2` yields a value of 1, because 7/2 equals 3 *remainder* 1.
- You can determine whether a number is even or odd by dividing the number by 2. If you get a remainder value, it's an odd number. If you don't and you get a remainder of 0, then it's an even number.
- You can determine whether a number is an integer or decimal by dividing the number by 1. If you get a remainder value, it's a decimal number. If you don't and you get a remainder of 0, then it's an integer.

Test Cases

- You can try these out to check if your program is working correctly.

Input	Result

Prompt #1: 2
Number prompts: -1, 100
	`Number count: 2`
	`Even numbers: 1`
	`Odd numbers: 1`
	`Positive numbers: 1`
	`Negative numbers: 1`
	`Integers: 2`
	`Decimal numbers: 0`

Prompt #1: 5
Number prompts: 4, 0, -5, -56, 3923
	`Number count: 5`
	`Even numbers: 2`
	`Odd numbers: 2`
	`Positive numbers: 2`
	`Negative numbers: 2`
	`Integers: 5`
	`Decimal numbers: 0`

Prompt #1: 10
Number prompts: -100, -23.23, 0, 62, 4.2, 5, 230.2, 12, 8.9999, -500.5
	`Number count: 10`
	`Even numbers: 3`
	`Odd numbers: 1`
	`Positive numbers: 6`
	`Negative numbers: 3`
	`Integers: 5`
	`Decimal numbers: 5`

Prompt #1: -2	`Error message`