CMPS 2680 Lab 6

Spring 2025

Part 1 - Setup

1. For this assignment you will have multiple files, so I would highly recommend creating a lab6 folder.
2. Inside of the lab6 folder you will run the following commands:

```
wget https://www.cs.csub.edu/~paul/cs2680/labs/lab6.zip
unzip lab6
```

   This will provide you with a starting website you will add javascript functionality to.
3. Review **`guess.html`** and become familiar with how it is set up.
   Notice the id attributes that have been added to the various elements.
   YOU WILL NOT MAKE ANY CHANGES TO **`guess.html`** FOR THIS LAB!
4. Review script.js and make sure you understand the existing code.
   Notice there are two sections marked **`TODO`**, this is where you will write your code.

Part 2 - Modify script.js

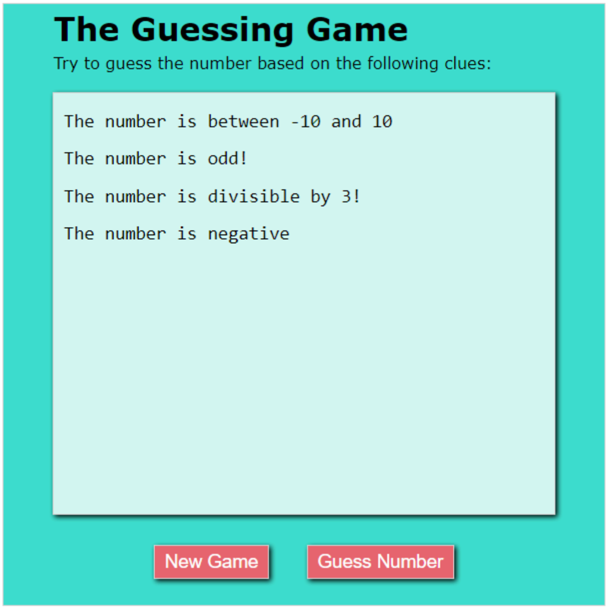
TODO #1:

1. Use getElementById and addEventListener to add a "click" event to the newGame button.
   This click event will call the **`startGame`** function you will define later.
2. Use getElementById and addEventListener to add a "click" event to the guessNumber button.
   This click event will call the **`playGame`** function you will define later.

TODO #2:

1. Define a function called **`startGame()`** that will do the following (in this order):
   a. Use the provided **`genRandom`** function to get a random number between -10 and 10
      Assign the returned value to the provided global variable **`number`**
   b. Use **`if/else`** blocks to analyze the number and create at least 3 clues (as strings)
   c. Assign these clues to the **`innerHTML`** of the "clues" div
      (You may want to create a variable to build your output using concatenation so you only have to update the innerHTML once at the end
2. Define a function called **`playGame()`** that will do the following (in this order):
   a. Use an **`if`** block to check if **`number`** is **`null`**
      If it is, use **`window.alert`** to notify the player that they must click "New Game" and end the function early using the **`return;`** statement.
   b. Use **`window.prompt`** and have the user enter their guess.
      Assign the returned value from the prompt to a variable called **`guess`**
   c. Use an **`if/else`** block to check if **`guess`** equals **`number`**
      - If they guess the right number, use **`window.alert`** to notify the user they won
      - If they guess the wrong number, use **`window.alert`** to notify the user they lost
      - Whether they win or lose, you should also include the value of **`number`** in the prompt
   d. Assign **`number`** to **`null`**

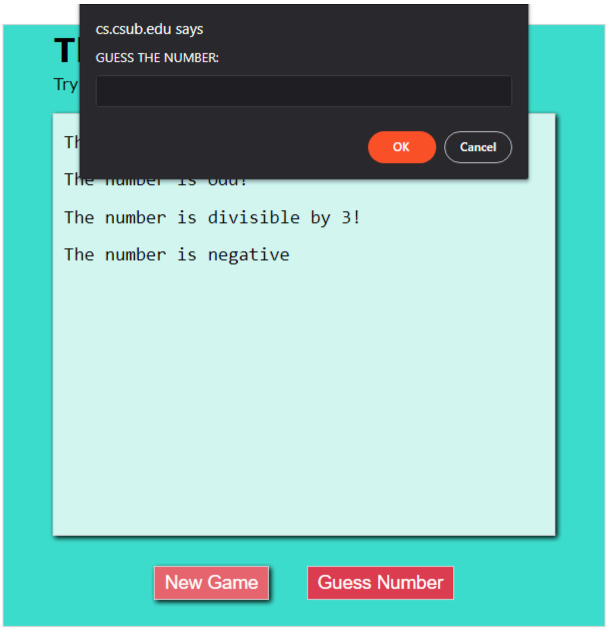
Completed Example



The Guessing Game

Try to guess the number based on the following clues:

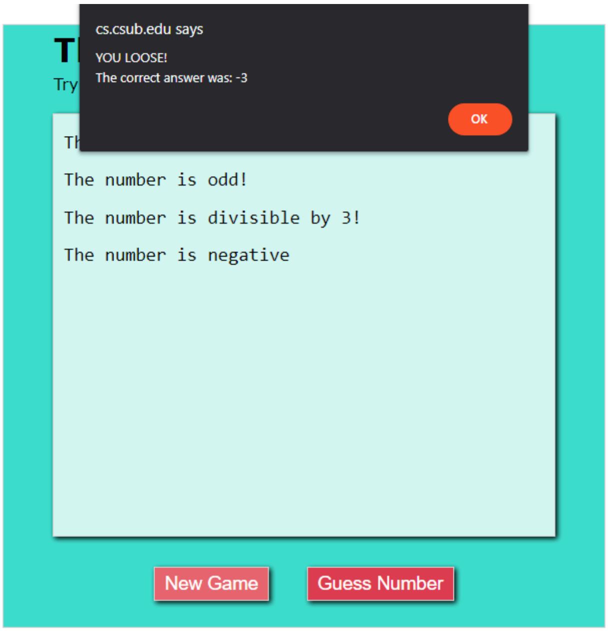
```
The number is between -10 and 10
The number is odd!
The number is divisible by 3!
The number is negative
```



cs.csub.edu says

GUESS THE NUMBER:

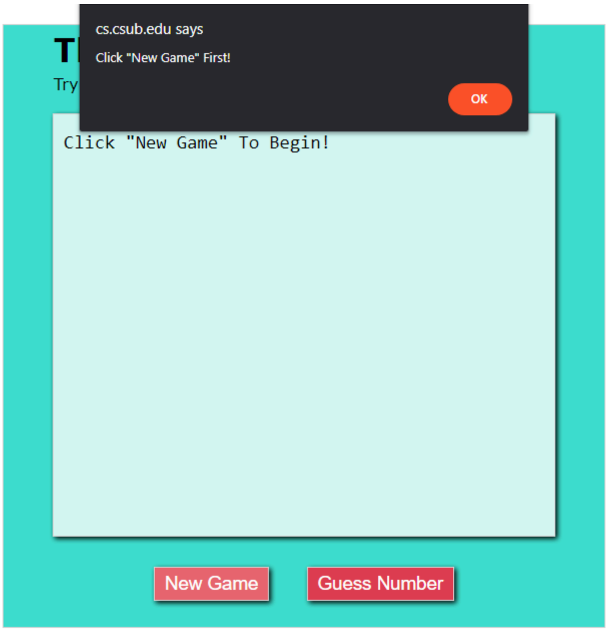
```
The number is odd!
The number is divisible by 3!
The number is negative
```



cs.csub.edu says

YOU LOOSE!
The correct answer was: -3

```
The number is odd!
The number is divisible by 3!
The number is negative
```



cs.csub.edu says

Click "New Game" First!

```
Click "New Game" To Begin!
```