

CSI 3140 - Summer 2025

LAB 4: Math Practice Tool

Due Date: Monday Jun 16th at 11:59PM EST

Compress all your files and upload the Zip file to the Brightspace. Please name the file using the following format: `CSI3140_Lab_<lab_number>_<student_id_1>_<student_id_2>.zip`

You will build a "Guess the Number" game. The script will first run an interactive guessing session using prompt and alert. After the game session concludes (either by winning, losing, or canceling), the script will replace the initial page content with a polished, styled "Game Over" screen that displays the results and the user's guess history.

Instructions:

Build a number guessing game that interacts with the user via browser prompts and then generates a styled HTML results page summarizing the game.

1. Generate a secret random number between 1 and 100. Set a guess limit (e.g., 7 tries).

2. Create `playGame()` **function**:

- **Game Loop:** Use a `while` loop that continues as long as the player has guesses remaining.
- **User Input:** Inside the loop, use `prompt()` to ask the user for their guess. Also, display the number of remaining guesses in the prompt.
- **Core Logic:** Use a conditional statement to display a message based on the user's guess:
 - If the guess is more than 50 numbers away from the secret number, the feedback should be "Too high!" or "Too low!".
 - If the guess is within 50 numbers (but not correct), the feedback should be "A little high" or "A little low".
 - When the guess is correct, display a success message.
- **Guess Count:** The page must display the number of guesses the user made during the game.
- **End Conditions:** The loop should terminate if the player wins, runs out of guesses, or clicks "Cancel" on the prompt.

3. Create `buildResultsPage()` **function**:

- After the game loop ends, use `document.write()` to generate a new HTML page content.
- **Show Guess Count:** The page must display the number of guesses the user made during the game.
- **Incorporate a switch block** to customize the final feedback only when the user wins,

based on the total number of guesses used.

- 1 guess → Exceptional performance
 - 2–3 guesses → Very good
 - 4–5 guesses → Decent effort
 - 6–7 guesses → Last-minute success
- **Display Result:** The page must display a clear "Game Over" title and a final message indicating whether the player won, lost, or cancelled.

4. The generated results page must be styled using CSS for a clean, professional look (you can define these styles in your `<head>`).

Example:

The example shows three screenshots of a game interface with a dark background and a central grey dialog box.

Screenshot 1: The dialog box has the title "This page says" and the text "I have picked a secret number between 1 and 100. You have 7 guesses." with a single blue "OK" button at the bottom right.

Screenshot 2: The dialog box has the title "This page says" and the text "Enter your guess (1-100). Guesses left: 7". It features a text input field containing "50" and two buttons at the bottom right: a blue "OK" button and a grey "Cancel" button.

Screenshot 3: The dialog box has the title "This page says" and the text "Too low! Try again." with a single blue "OK" button at the bottom right.

• • •

You Won!

Congratulations! You guessed the number!

You used 4 guess(es).

Decent effort