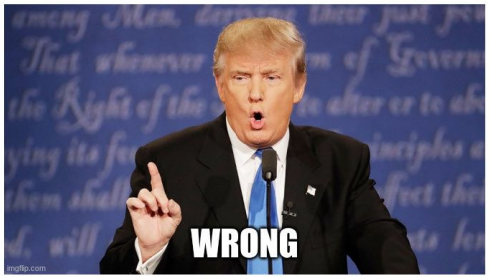
# Today's Lab (Due by DEC 06, 11:59pm)

## Number Guessing Game




111306086_Number Guessing Game



- Game Rule:https://ppt.cc/fKICPx(Confirm that we share the same understanding of the game rules.)

- The HTML content must begin with an `<h1>` element whose text is “StudentID_Number Guessing Game”.

- Use `window.prompt()` to let the user enter a guessed number.

- The guessing range is 1 to 100, and the maximum number of attempts is seven.

- Each prompt must inform the user of the current valid range and the remaining number of attempts.

- Create an empty `<div>`. Insert `correct.jpg` into it when the user guesses correctly, or insert `wrong.jpg` when the attempts reach zero, and then end the game. When the guess is correct, use `alert()` to notify the user. When attempts reach zero, notify the user of the correct answer.

- Each input must be validated for empty values and checked whether it falls within the valid range. If invalid, use `alert()` to remind the user of the incorrect input.

# Tips

- Math.random() : 0~1

```
var answer = Math.floor(Math.random() * 100) + 1;
```

- var chances = 7;

```
while (chances > 0) {

….

}
```

- input.trim()
- if (input === "" || isNaN(input))
- var guess = Number(input);

# Submission Rule

1. File names: **StudentID_JS1_Lab3.html**
2. Deadline: **DEC. 6 (SAT), 11:59pm**
3. Submission method:
   - Submit through the GitHub repo created in Week 3.
   - In your repo, create a folder named **Lab3_GuessNum**, place all files inside (including "images" folder with all images placed in it), and push to GitHub.
   - Submit the **Deployed GitHub Page URL** to your **StudentID_JS1_Lab3.html** file to Moodle.
   - **Put images in the folder and push to GitHub.**

# Today’s Homework

## 111306086_1A2B

Enter 4 unique numbers [ ] Guess!

Game Record：
第1次: 1234 → 0A2B
第2次: 5678 → 0A2B
第3次: 0912 → 1A0B
第4次: 9012 → 1A0B
第5次: 0913 → 0A1B
第6次: 0932 → 1A1B
第7次: 3092 → 1A1B
第8次: 0392 → 2A0B
第9次: 4362 → 2A1B
第10次: 5342 → 2A1B
第11次: 0352 → 3A0B
第12次: 0354 → 2A0B
第13次: 7352 → 3A0B
第14次: 8352 → 3A0B
第15次: 9352 → 3A0B
第16次: 1352 → 3A0B
第17次: 4352 → 3A0B
第18次: 6352 → 4A0B

**Game Rules:** https://en.wikipedia.org/wiki/Bulls_and_Cows

- **Requirements**
  - The top of the HTML content must use an `<h1>` tag with the text "**StudentID_HW6_1A2B**".
  - The numbers (digits) must not be repeated.
  - The result of each guess and the **XAYB** outcome must be recorded on the page.
  - Use HTML `<input>` and `<button>` elements to enter and submit answers.
  - Count the number of attempts. When the user guesses correctly, use `alert()` to notify them of the success and display the total number of attempts.
  - Validate each submission. If the input is invalid, use `alert()` to warn the user of the error:
    - Check if the input consists of exactly **4 digits**.
    - Check if there are any **duplicate digits**.
- **Optional:** If you have extra time/energy, feel free to style/beautify the page!

**USEFUL TIP!**

- Use `<button onclick="myFunction()">` to call the function that checks the answer when the submit button is pressed.
- JavaScript Functions: https://www.w3schools.com/js/js_functions.asp
- You can clear the input field by setting `inputField.value = ""`.
- Convert both the answer (guess) and the secret code into Arrays to make comparing digit positions easier.
- **Logic for generating random numbers:**
  1. Create an Array containing numbers 0~9.
  2. Randomly generate a number between 0~9 to select an index from the above Array.
  3. Add the selected number to a new Array and remove that number from the original Array.
  4. Repeat steps 2 and 3 until you have four numbers.

- function generateAnswer()

  Generates the answer for the game.

  It randomly selects four non-repeating digits from 0 to 9 and builds a string until its length becomes four. The final four-digit string is returned as the game's answer.

- function play()

  Handles the game flow each time the player clicks "Guess."

  It reads the user's input, checks whether it is valid, then updates the guess count, calculates the A/B result, and displays the attempt in the record area. Finally, it clears the input box for the next guess.

- function isValid(str)

  Validates the player's input.

  The input must be exactly four characters long, must be all digits, and all four digits must be unique. If all conditions are met, it returns true; otherwise, it returns false.

- function checkAB(guess, ans)

  Calculates how many A's and B's the guess gets.

  It compares each position one by one: if the digit and position match, A increases; if the digit exists in the answer but is in a different position, B increases. It then returns a string such as "2A1B."

# Submission Rule

1. File names: **StudentID_JS1_HW7.html**
2. Deadline: **DEC. 11 (THU), 11:59pm**
3. Submission method:
   - Submit through the GitHub repo created in Week 3.
   - In your repo, create a folder named **HW7_JS1**, place all files inside and push to GitHub.
   - Submit the **Deployed GitHub Page URL** to your **StudentID_JS1_HW7.html** file to Moodle.