SE Boot Camp Project 3 Outline – Web Game App

Instruction:

The purpose of project 3 is to use the knowledge of HTML, CSS, JavaScript and DOM to create a simple game application.

Game Rules:
Guess a number between 1 and 100.

The program will generate a secret number which will be hidden to the audience. Players need to guess the number by typing the number into a box (input element) then clicking on check button.

Initial current score is 10. History High Score is 0.

```
1. When there is no input or invalid value, and user press check button,
there should be a message showing no input or invalid value.

2. When user put a wrong number, and user press check button, message should
show it is too low or too high. The current score is set to 10 but it will be
decreasing by 1 every time when user guesses a wrong number. If this number
becomes 0, the player loses the game, and a message such as "Sorry, game is
over" should show.

3. When user put a correct number before current score value becomes 0,
background color should of the block/page should change, the secret number
shows, the background color/dimensions of the box of secret number changes,
and high score will be updated with new high score(if the current play win
the game and gets a higher score) or remain the same.
```

Implement game rest functionality, so that the player can make a new guess!
```
1. Attach a click event to the 'Play' button
2. In the handler function, restore initial values of the 'score' and
'secretNumber'
3. Restore the initial conditions of the message, secret number, current
score and guess input fields
4. Also restore the original background color and number's background/width
and color.
```
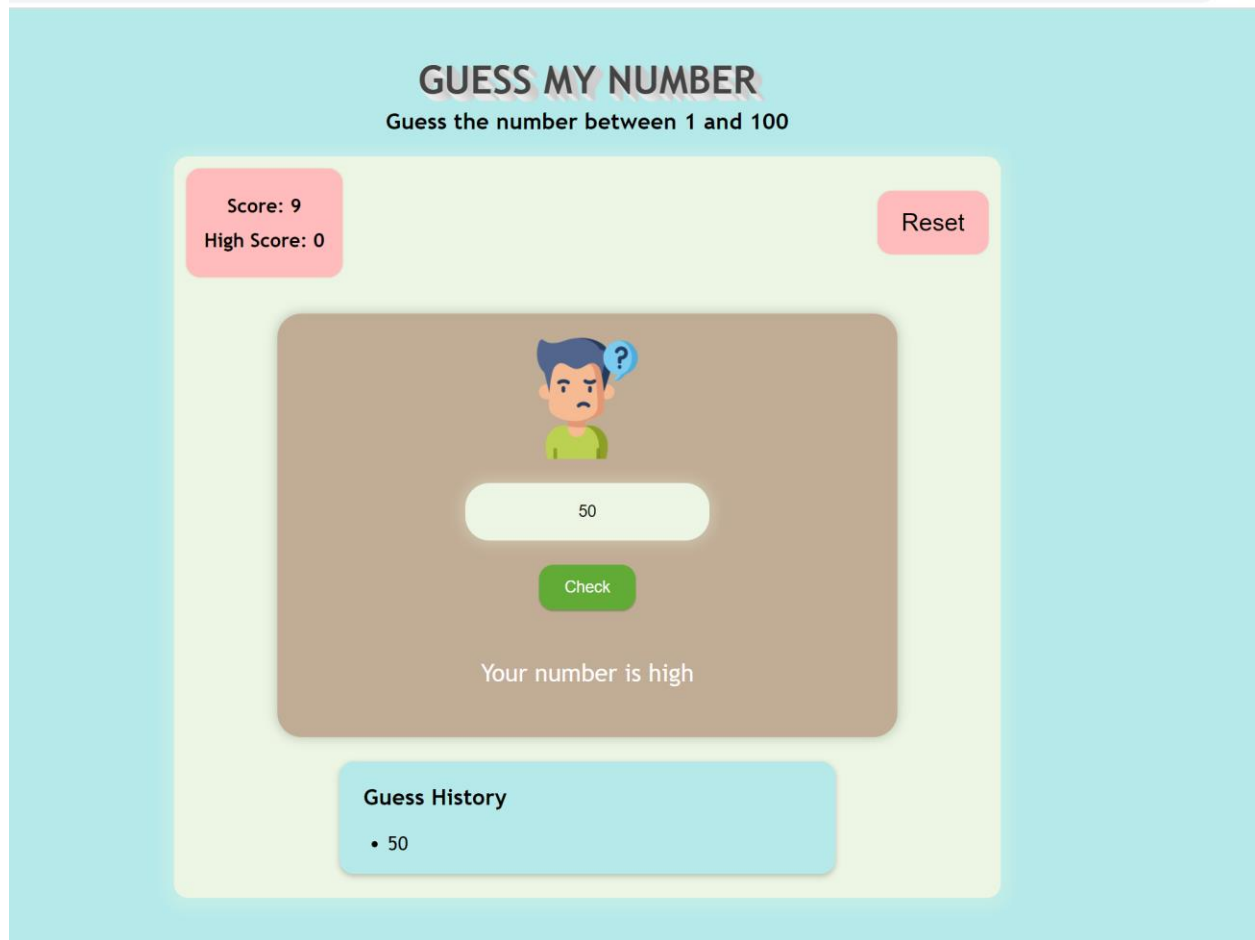
**Final Project will be graded as the following:**

| Items | Grade |
|---|---|
| Application and manipulation of JavaScript skills learned in this class | 30 |
| Complete all website and game's features as explained in class | 20 |
| Complete all HTML CSS and JavaScript Files, publish it through Github | 15 |
| Presentation and organization of the web application (the appearance of the website) | 10 |
| Files organization | 5 |
| Oral Presentation of the work. The presentation will be hold on May 26/27, 2022 at 5:10pm | 10 |
| Submission on time through GitHub: Due date is May 25, 2022 before 5:00pm. | 10 |

**Expected Project Result:**

**Initial Status (same as after clicking on Play! button. History High score value may vary)**
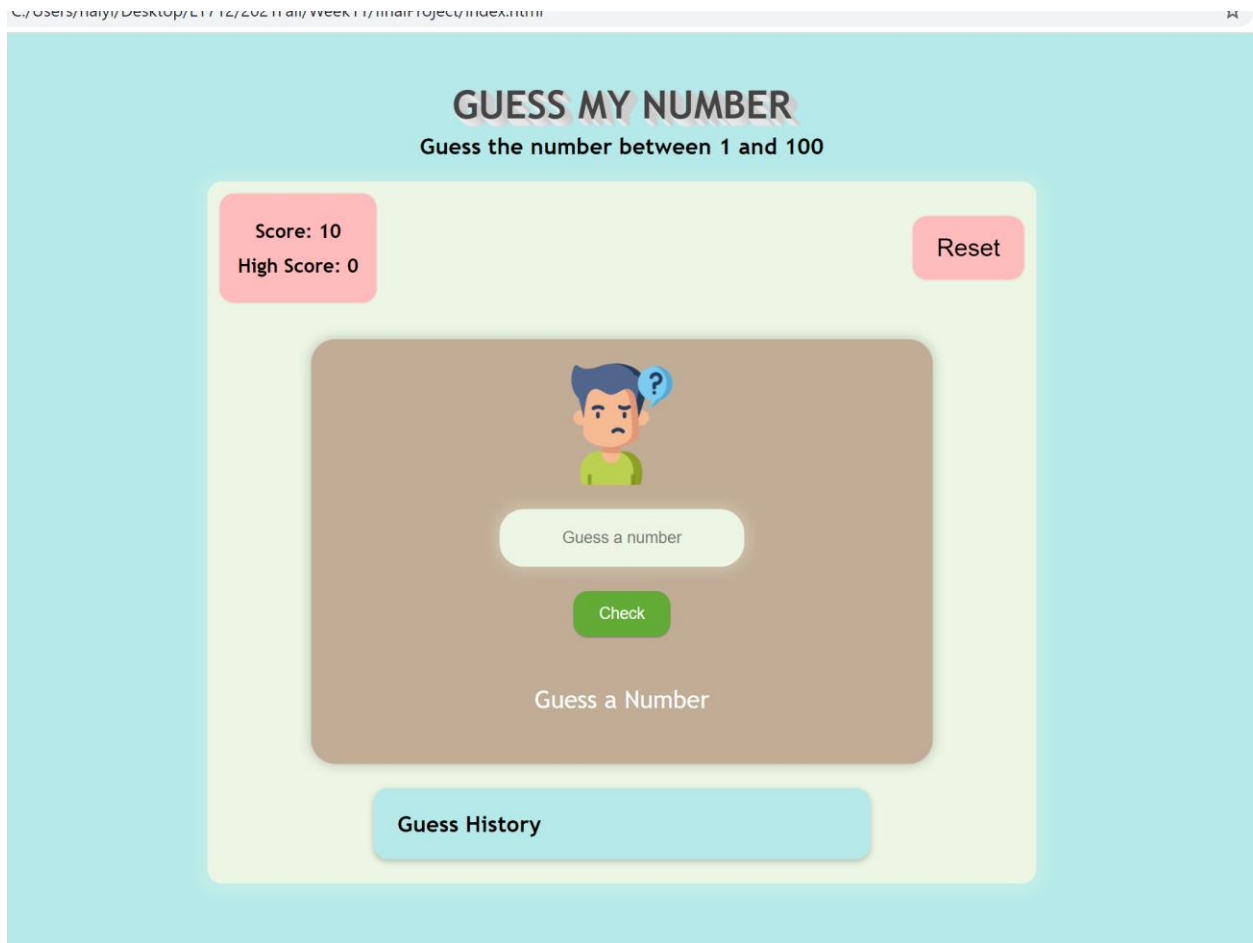
If user puts a number greater than secret number, it shows as below:

# GUESS MY NUMBER

**Guess the number between 1 and 100**

Score: 9
High Score: 0

Reset

50

Check

Your number is high

**Guess History**

- 50

If user puts a number less than the secret number, it shows as below:
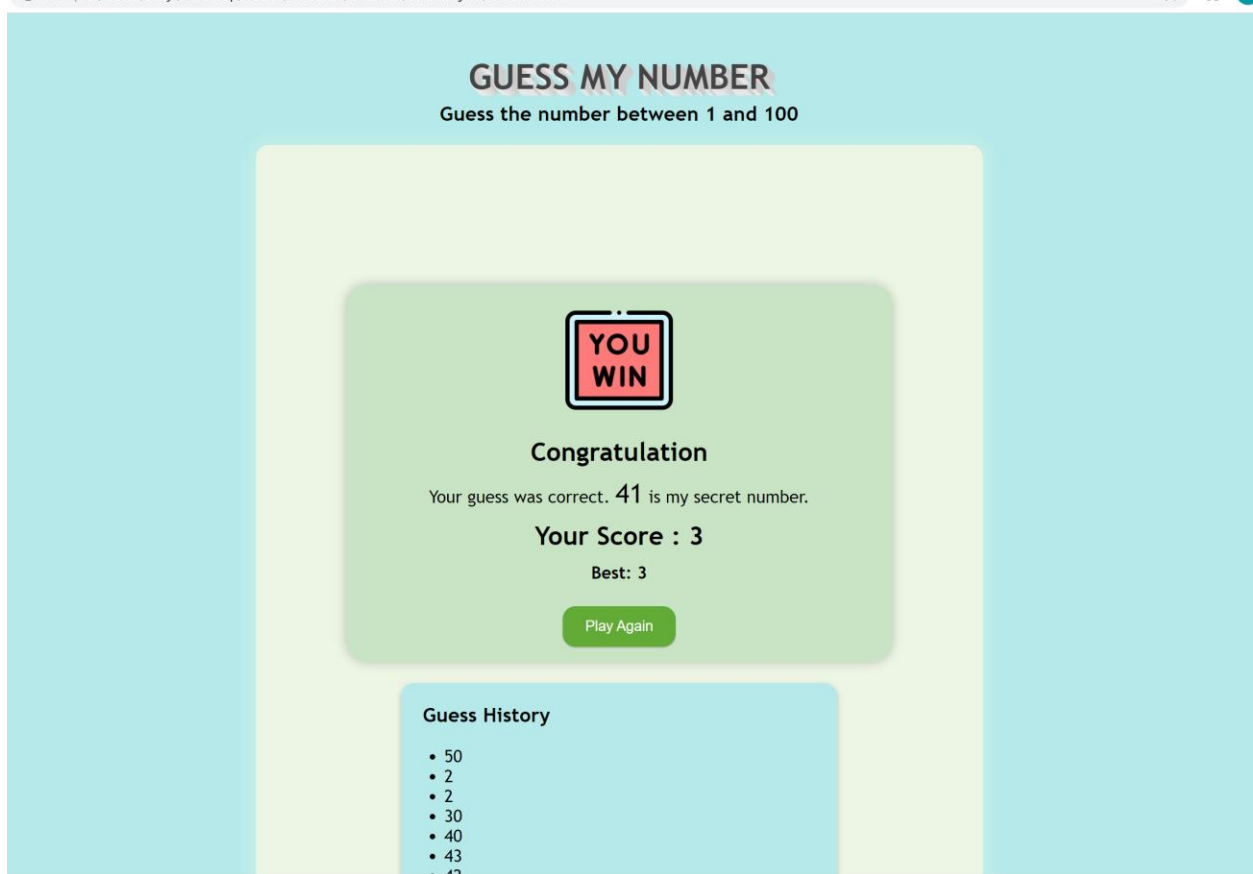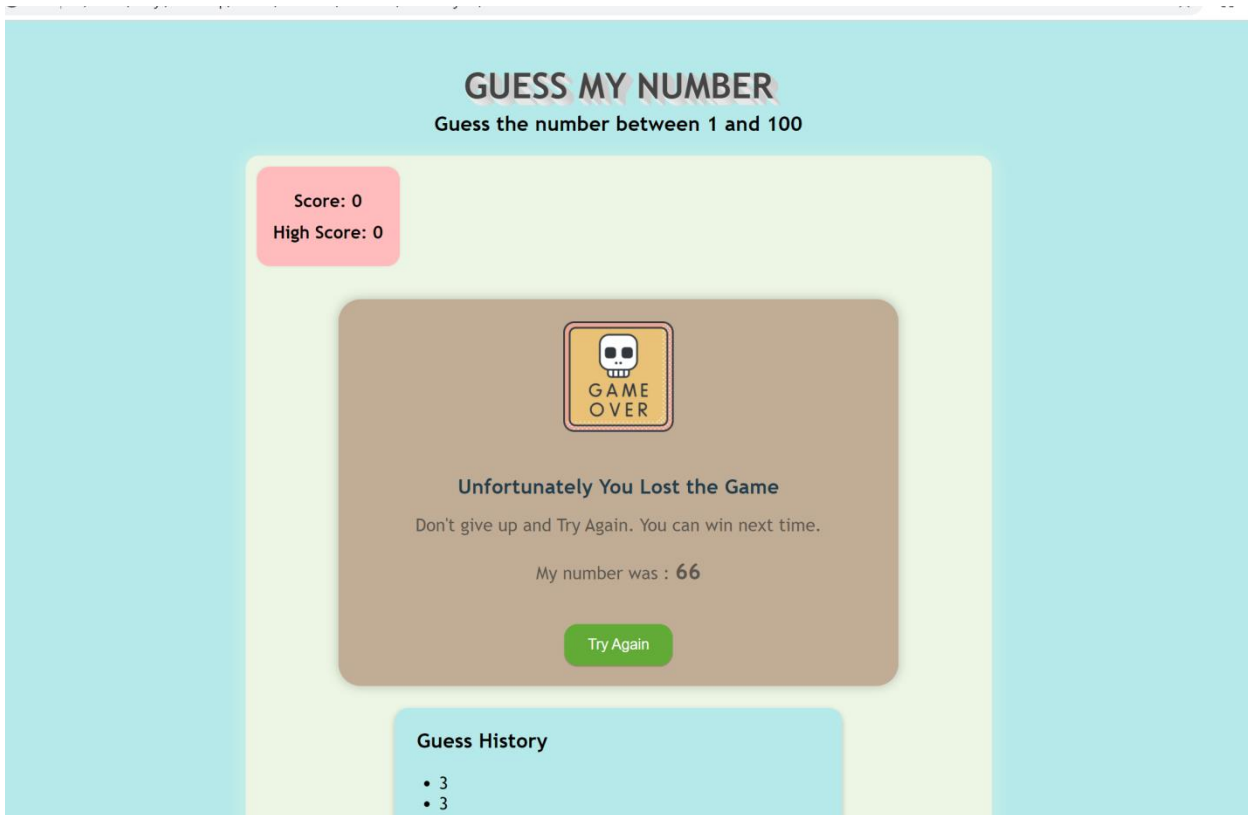
If user doesn't put a number (invalid input such as string "Hello") and click on the Check button, it should show:

# GUESS MY NUMBER

**Guess the number between 1 and 100**

Score: 7
High Score: 0

Reset

Guess a number

Check

You can only enter numbers between 1 and 100

**Guess History**

- 50
- 2
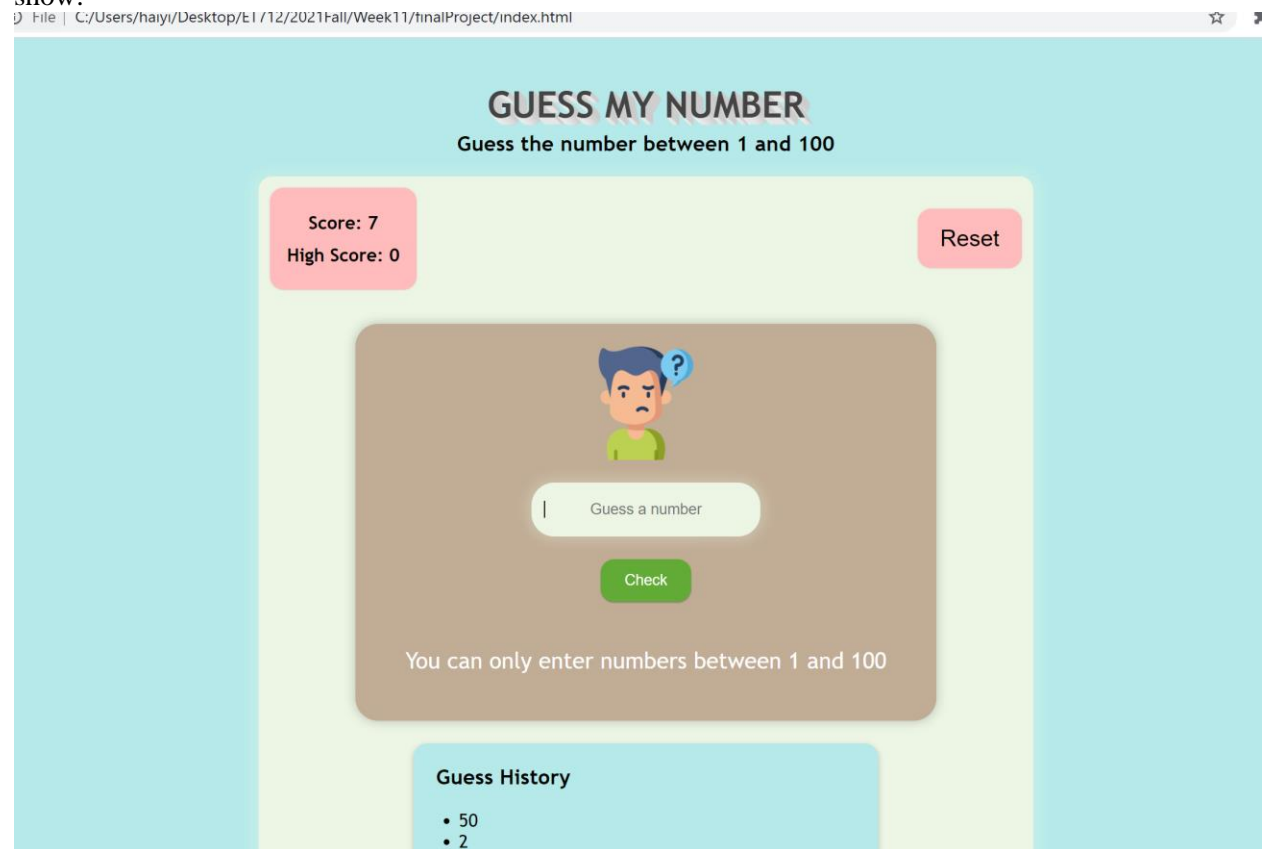
When user types the correct secret number, input should be disabled or hidden, demo result is shown as below:

If user lost the game, it shows:



# GUESS MY NUMBER
**Guess the number between 1 and 100**

Score: 0
High Score: 0

GAME OVER

**Unfortunately You Lost the Game**

Don't give up and Try Again. You can win next time.

My number was : **66**

Try Again

**Guess History**
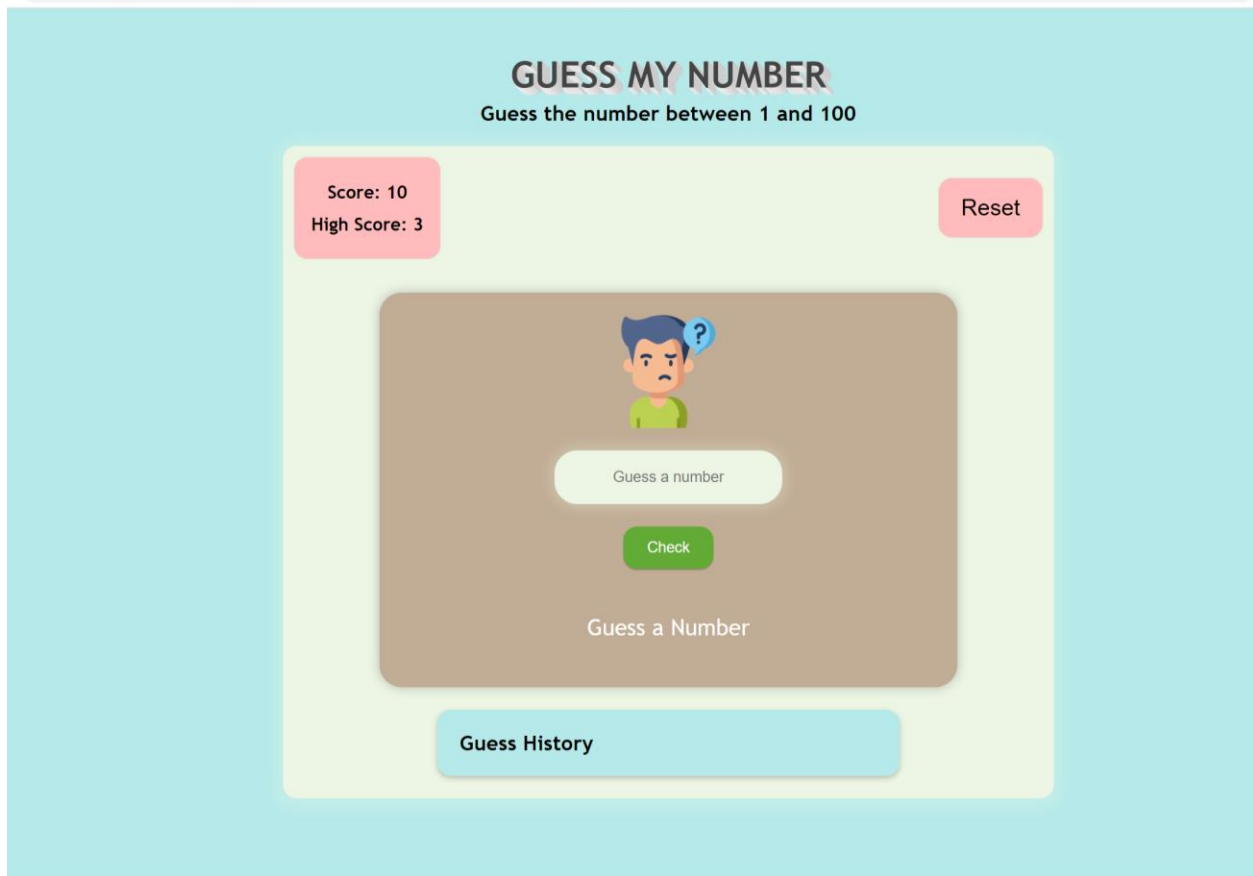
- 3
- 3

Click on Play button to restart the game.
Note: the **highest score** of previous user should always remain in a new game unless the page is refreshed.



Additional Fetures (bonus):

1.Add gussing history so players can track their previous input.
2. Let the players know if the same wrong number is checked again. No point should be detucted in this case.