Number Guessing Game: A Fun Web Development Project Using HTML,CSS and JavaScript

Introduction

The Random Number Game is a captivating and interactive web application designed to challenge and entertain users through a simple yet engaging guessing game. The game's primary objective is for the user to guess a randomly generated number between 1 and 10. Each correct guess increases the user's score, while incorrect guesses prompt the user to try again, making it a fun and addictive way to test one's guessing abilities.This project serves as a practical demonstration of how HTML, CSS, and JavaScript can be combined to create a functional and visually appealing web application:

HTML (HyperText Markup Language)

Provides the structural foundation of the web page. It includes elements such as paragraphs, input fields, buttons, and spans for displaying the score, allowing users to interact with the game seamlessly.

CSS (Cascading Style Sheets)

Enhances the visual presentation of the game. It is used to center the content, apply attractive fonts, add colors, and style elements to create a user-friendly and aesthetically pleasing interface.

JavaScript

Adds dynamic functionality to the game. It generates random numbers, validates user input, updates the score, and provides real-time feedback, making the game interactive and responsive. The Random Number Game not only serves as an entertaining pastime but also as an educational tool for those learning web development. By building and interacting with this game, users can gain a deeper understanding of essential web technologies and how they work together to create interactive applications. This project illustrates key concepts such as event handling, DOM manipulation, and responsive design, making it an excellent starting point for aspiring web developers.

HTML Structure

The HTML file provides the structure of the web page. It includes elements for displaying the score, an input field for user guesses, a button to submit the guess, and a text area to display results.

Key HTML Elements:

- Score Display: Displays the current score.
- Input Field: Allows the user to enter their guess.
- Button: Submits the guess for validation.
- Result Text: Provides feedback on whether the guess was correct or incorrect.

```
<!DOCTYPE html>
<html lang="en">
<head>
     <meta charset="UTF-8">
     <meta name="viewport" content="width=device-width, initial-scale=1.0">
     <title>Document</title>
     <link rel="stylesheet" href="RandomNumGame.css">
</head>
<body>
    <div class="game-container">
        <p>Your Score: <span id="score">0</span></p>
        <input id="userIn" placeholder="Enter a number">
        <button onclick="checkFunc()">Check</button>
        <p id="resText">Guess the Correct Number!</p>
    </div>
     <script src="RandomNumGame.js"></script>
</body>
</html>

```

CSS Styling

The CSS file adds visual styles to the HTML elements, making the game visually appealing and user-friendly.

Key CSS Styles:

- Body: Centers content vertically and horizontally, sets background color.
- Game Container: Adds border, padding, background color, and shadow to create a box around the game content.
- Paragraphs, Input, Button: Styles the text, input field, and button to be larger and more visually appealing.

```css
/* Style for the body to center everything */
body {
    display: flex;
    flex-direction: column;
    align-items: center;
    justify-content: center;
    height: 100vh;
    margin: 0;
    font-family: Arial, sans-serif;
    background-color: #e0f7fa; /* Light cyan background color */
}

/* Style for the score paragraph */
p {
    font-size: 1.5em; /* Larger font size */
    margin: 20px 0; /* More space around paragraphs */
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
}

/* Style for the input and button */
#userIn {
    padding: 15px;
    font-size: 1.2em; /* Larger font size */
    margin: 15px 0;
    border: 2px solid #ccc;
    border-radius: 5px;
    width: 250px; /* Wider input field */
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
}
```

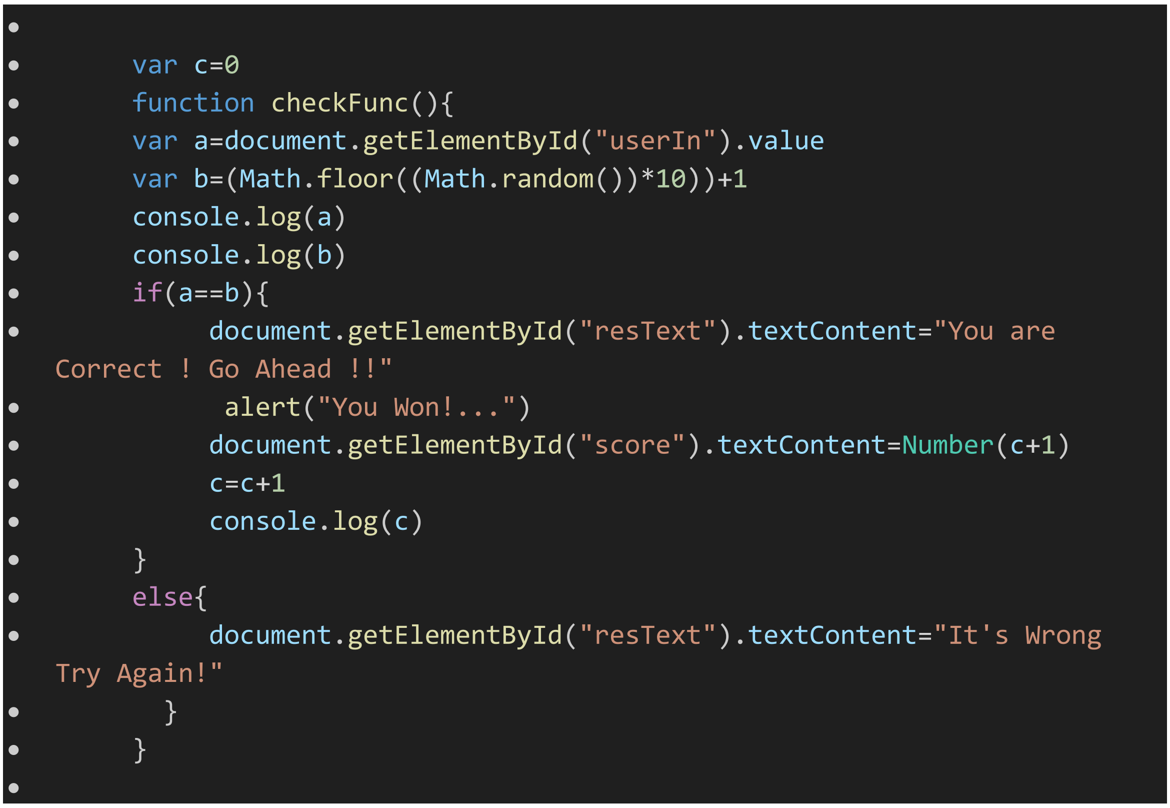
```css
.game-container {
    border: 2px solid #007BFF; /* Border color */
    border-radius: 10px; /* Rounded corners */
    padding: 100px; /* Space inside the box */
    background-color: #ffffff; /* White background for the box */
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); /* Slight shadow for depth */
    text-align: center; /* Center text inside the container */
}

button {
    padding: 15px 30px;
    font-size: 1.2em; /* Larger font size */
    margin: 15px 0;
    border: none;
    border-radius: 5px;
    background-color: #007BFF; /* Blue background color */
    color: white;
    cursor: pointer;
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
}

button:hover {
    background-color:forestgreen ;/* Darker blue on hover */
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
}

/* Style for the result text */
#resText {
    font-size: 1.5em; /* Larger font size */
    margin: 20px 0;
    color: #333;
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
}

```

JavaScript Functionality

The JavaScript file provides the game’s functionality, including generating random numbers, checking the user’s guess, and updating the score and result text.

Key JavaScript Functions:

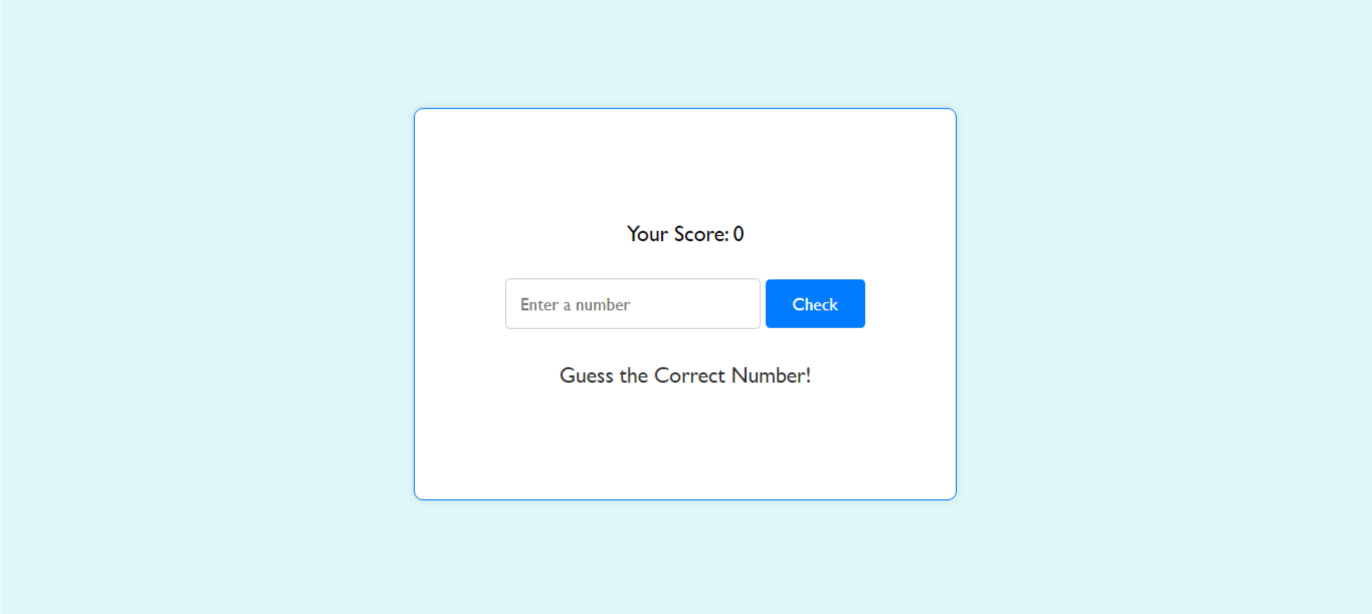
- checkFunc(): Validates the user's guess, updates the score if the guess is correct, and provides feedback.



```
var c=0
function checkFunc(){
var a=document.getElementById("userIn").value
var b=(Math.floor((Math.random())*10))+1
console.log(a)
console.log(b)
if(a==b){
     document.getElementById("resText").textContent="You are Correct ! Go Ahead !!"
      alert("You Won!...")
     document.getElementById("score").textContent=Number(c+1)
     c=c+1
     console.log(c)
}
else{
     document.getElementById("resText").textContent="It's Wrong Try Again!"
  }
}
```

Demonstration of the project

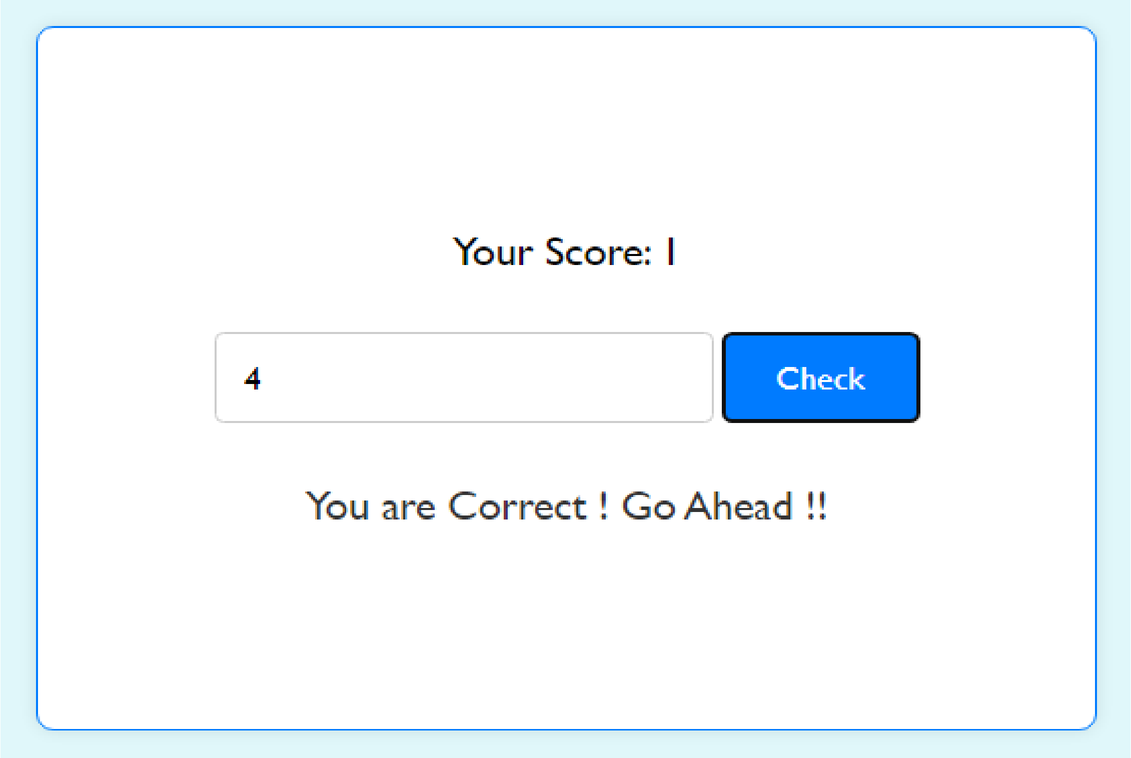
User Interface



Your Score: 0

Guess the Correct Number!

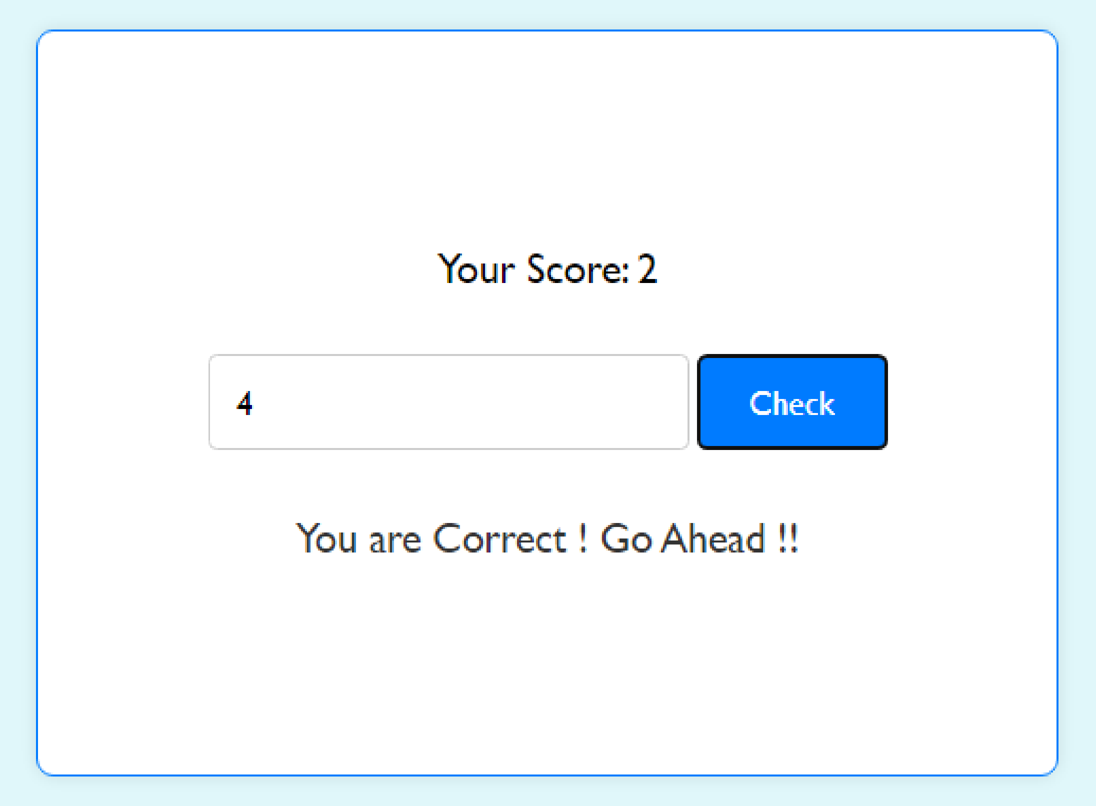
For the correct Guess



Your Score: 1

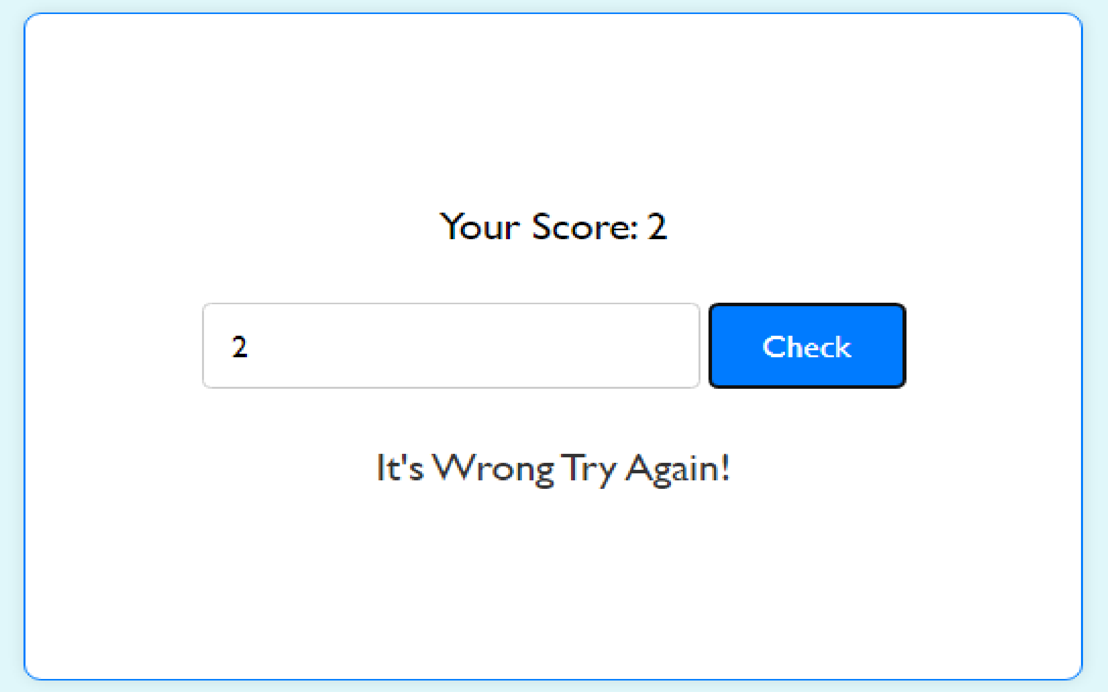
4

You are Correct ! Go Ahead !!

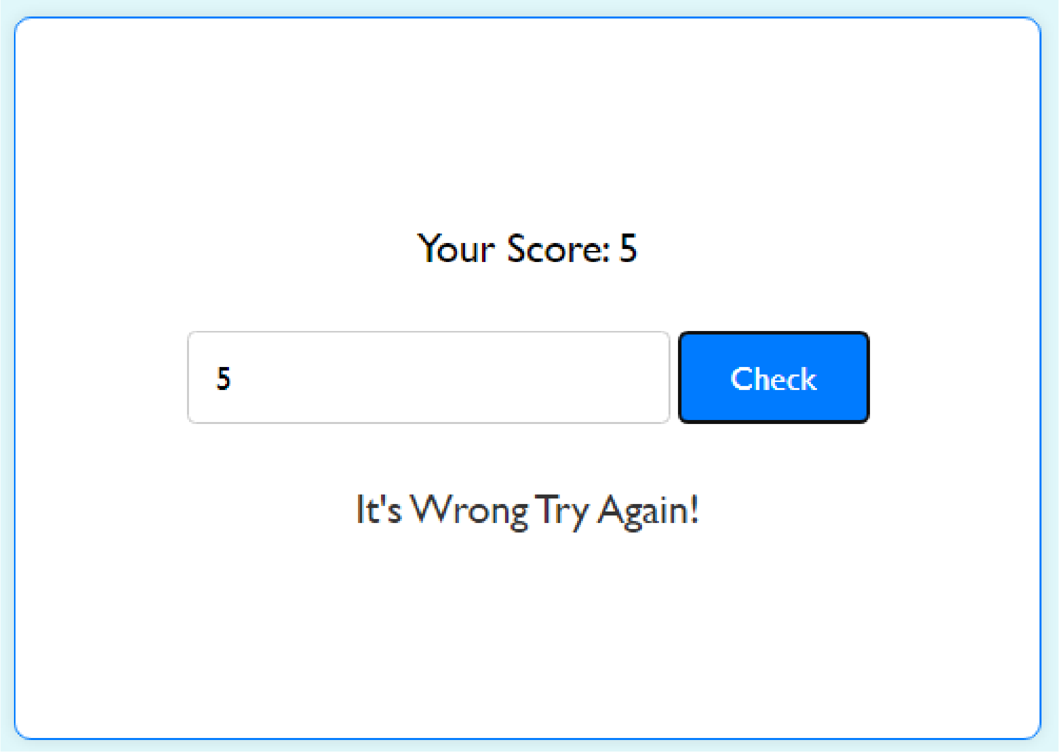

Your Score: 2

You are Correct ! Go Ahead !!

For the wrong Guess


Your Score: 2

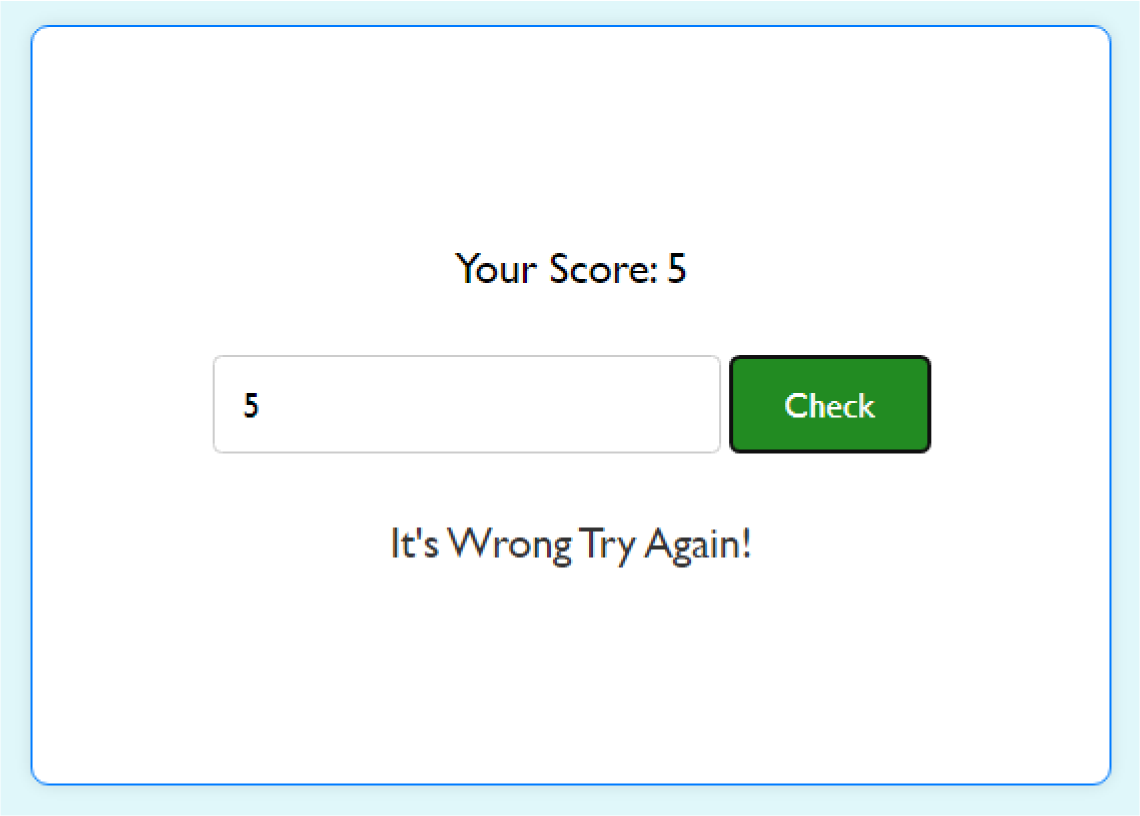
It's Wrong Try Again!



Your Score: 5

It's Wrong Try Again!

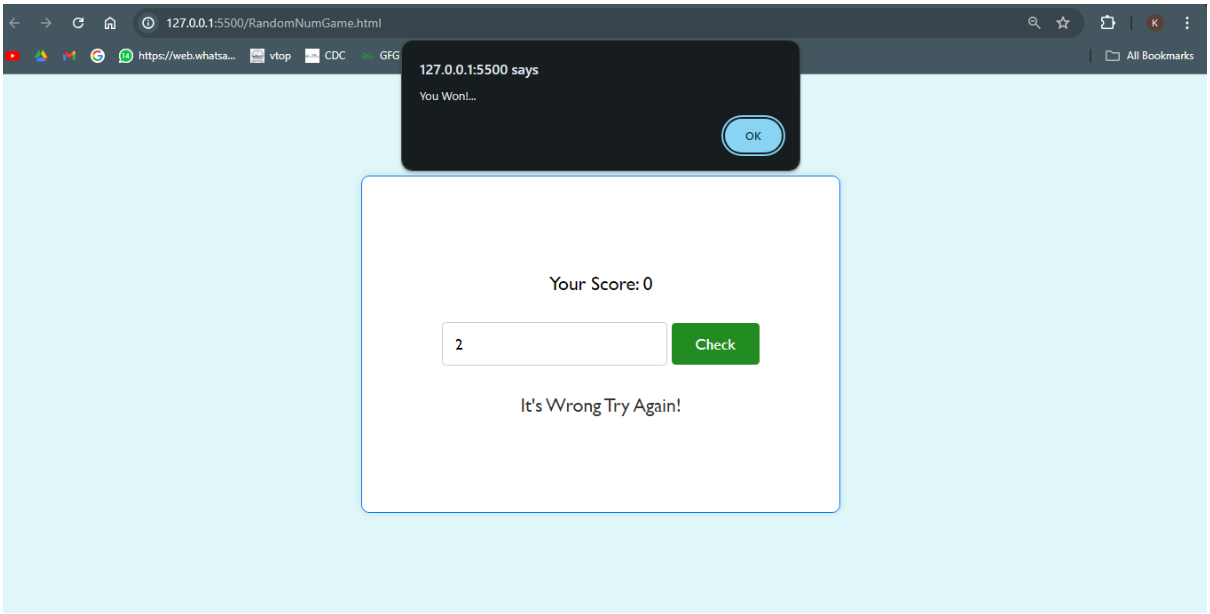
Touching the Check button it will change the color



Your Score: 5

It's Wrong Try Again!

Getting alert message if we Guess Correct Number



127.0.0.1:5500 says

You Won!...

Your Score: 0

It's Wrong Try Again!

Conclusion

The Random Number Game project demonstrates how HTML, CSS, and JavaScript can be combined to create a simple interactive web application. HTML provides the structure, CSS enhances the appearance, and JavaScript adds dynamic functionality. This project is an excellent example of basic web development practices, integrating user input, event handling, and dynamic content updates.