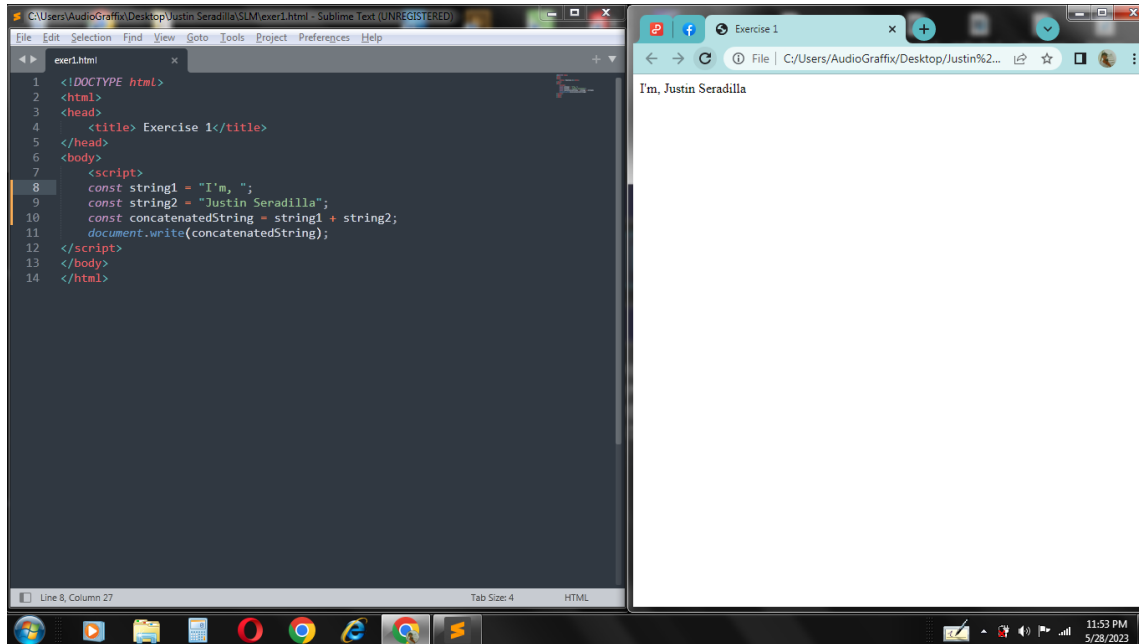


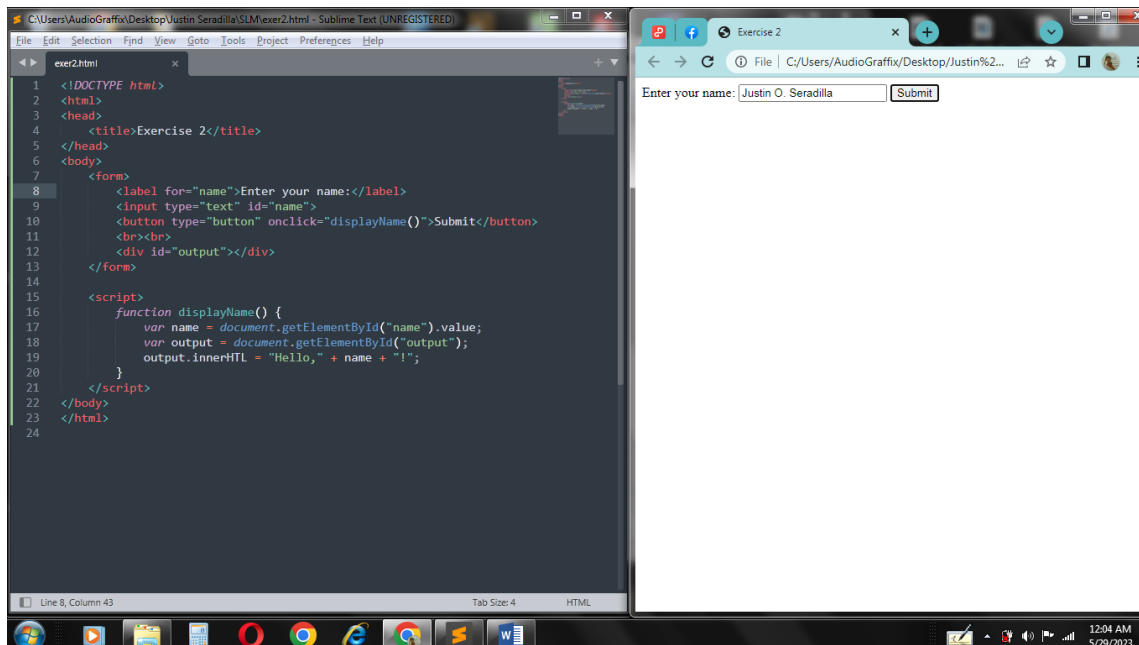
Seradilla Justin O.

BSIT II D

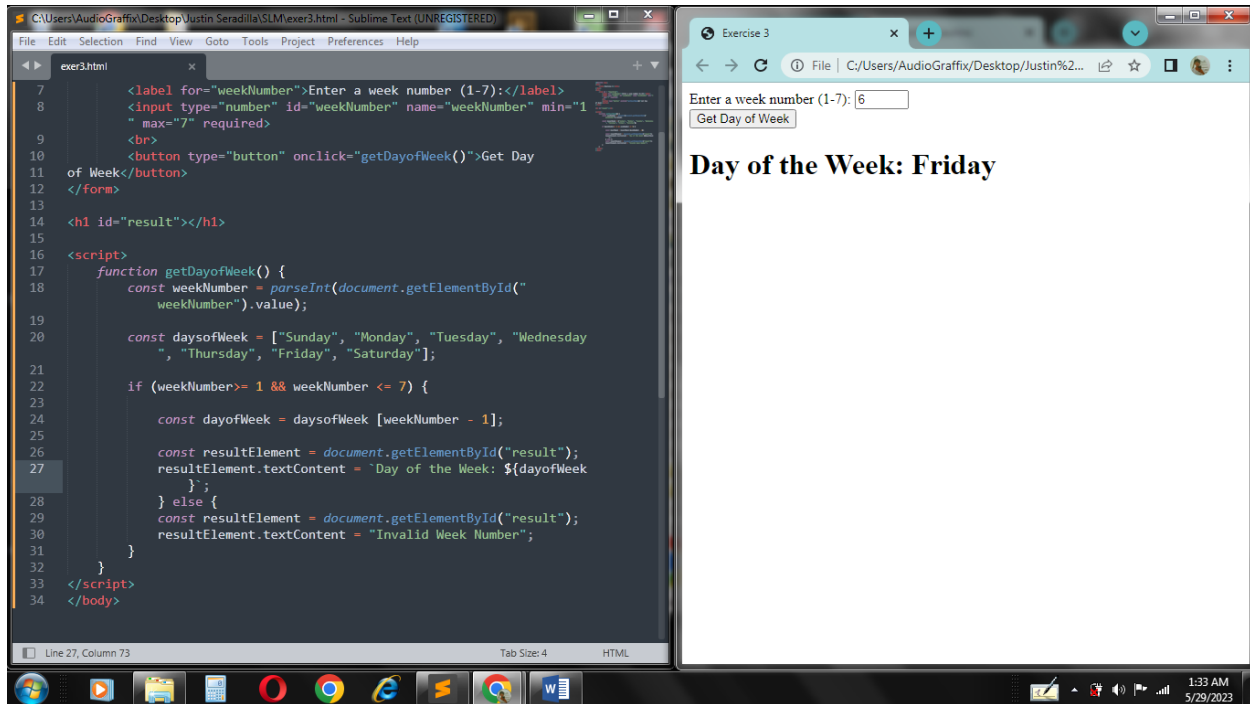
Exer1



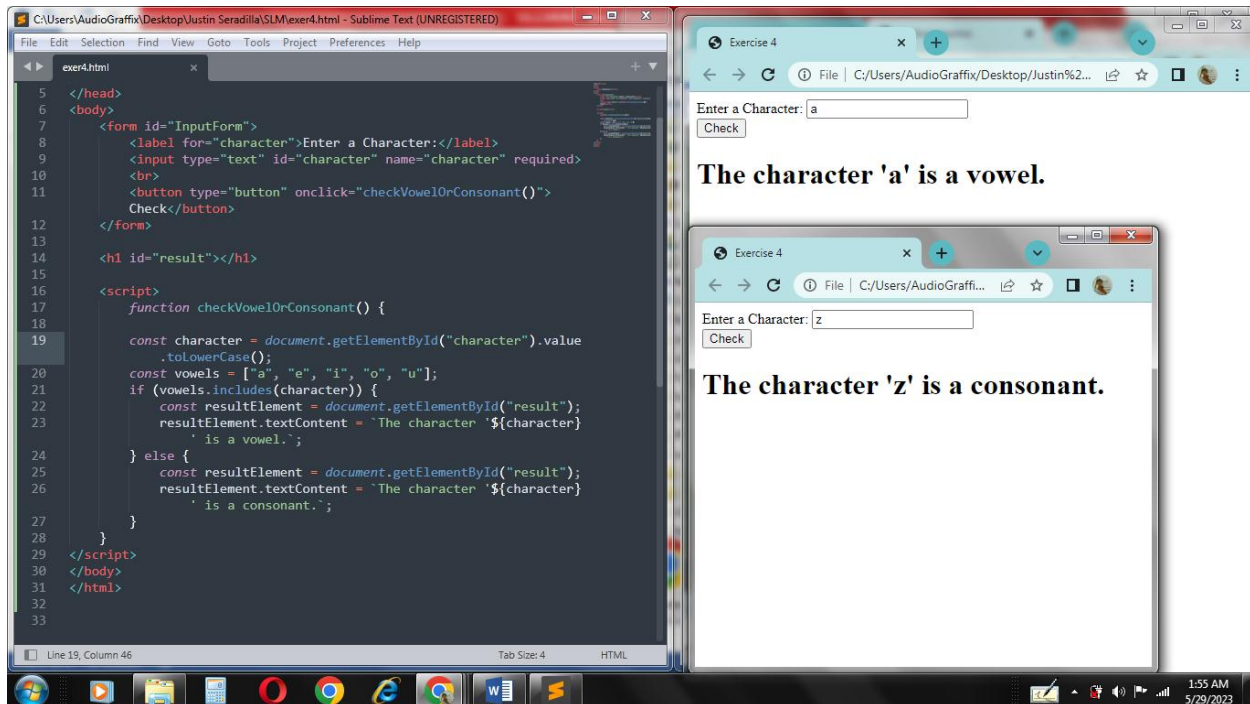
Exer2



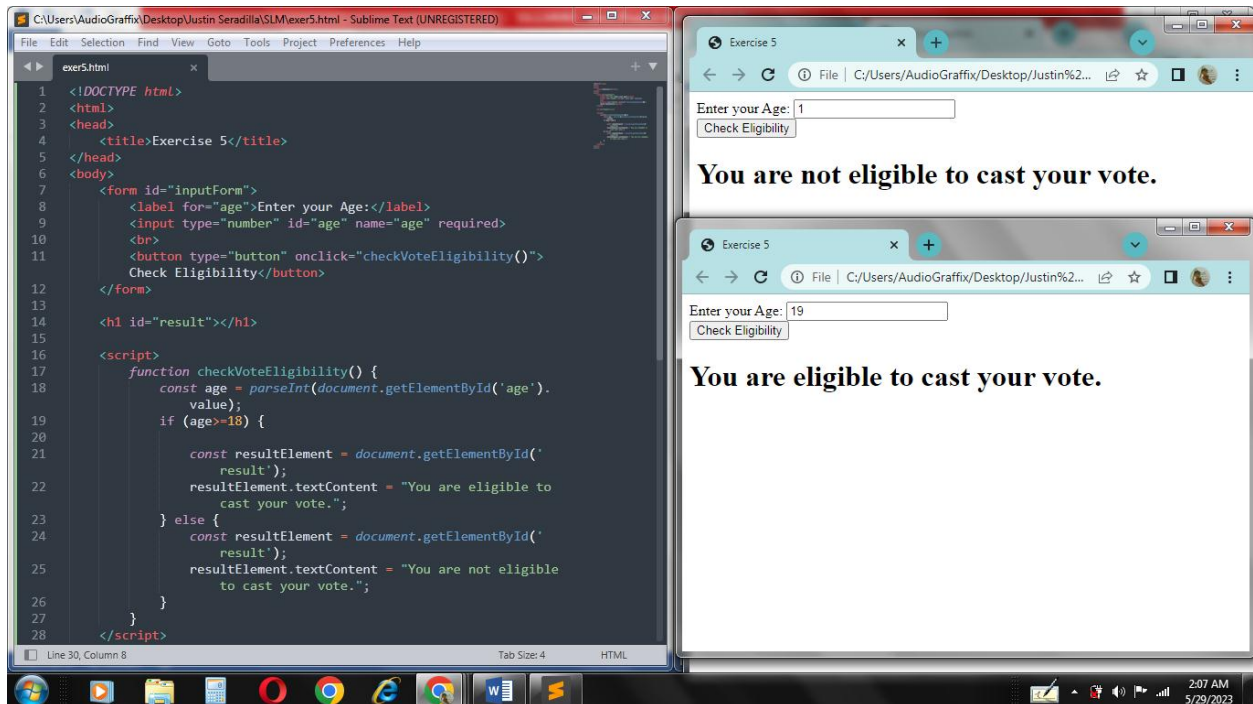
### Exer3



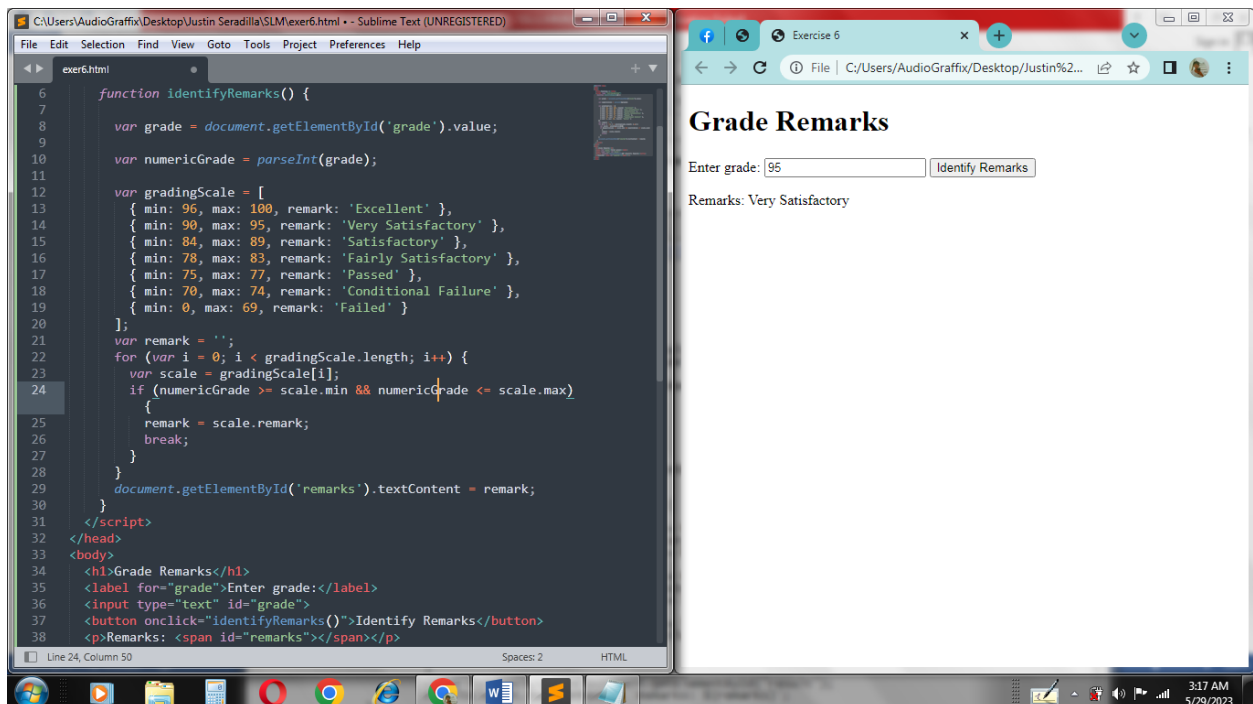
### Exer4



## Exer5



## Exer6



## Exer7

The screenshot displays the development environment for 'Exer7'. On the left, the Sublime Text editor shows the HTML and JavaScript code for a number classification form. The HTML includes a form with an input field for an integer and a 'Check' button. The JavaScript function 'checkNumberClassification' processes the input, determining its positivity (Positive, Negative, Zero) and parity (Even, Odd). On the right, two browser window snapshots show the form's output. The first snapshot shows the input '2' resulting in 'Number: 2 Positive: Even'. The second snapshot shows the input '-1' resulting in 'Number: -1 Negative: Odd'.

```
7 <form id="inputForm">
8   <label for="grade">Enter an Integer:</label>
9   <input type="number" id="number" name="number" required>
10  <br>
11  <button type="button" onclick="checkNumberClassification()">
    Check</button>
12 </form>
13
14 <h1 id="result"></h1>
15
16 <script>
17   function checkNumberClassification() {
18     const number = parseInt(document.getElementById("number")
19       .value);
20
21     let positivity = "";
22     if (number > 0){
23       positivity = "Positive";
24     } else if (number < 0) {
25       positivity = "Negative";
26     } else {
27       positivity = "Zero";
28     }
29
30     let parity = "";
31     if (number % 2 === 0) {
32       parity = "Even";
33     } else {
34       parity = "Odd";
35     }
36
37     const resultElement = document.getElementById("result");
38     resultElement.textContent = `Number: ${number} ${positivity}: ${parity}`;
39   }
40 </script>
```

Browser Output 1: Enter an Integer: 2  
Check  
Number: 2 Positive: Even

Browser Output 2: Enter an Integer: -1  
Check  
Number: -1 Negative: Odd

## Exer8

The screenshot displays the development environment for 'Exer8'. On the left, the Sublime Text editor shows the JavaScript code for three different loops: While, Do-While, and For. Each loop is designed to generate a sequence of numbers from 0 to 15. On the right, three browser window snapshots show the output of each loop, all displaying the sequence '0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15'.

```
9 <h2>While Loop</h2>
10 <p id="whileLoopResult"></p>
11
12 <h2>Do-While Loop</h2>
13 <p id="doWhileLoopResult"></p>
14
15
16 <h2>For Loop</h2>
17 <p id="forLoopResult"></p>
18 <script>
19   let whileLoopResult = "";
20   let whileLoopCount = 0;
21   while (whileLoopCount <= 15) {
22     whileLoopResult += whileLoopCount + " ";
23     whileLoopCount++;
24   }
25   document.getElementById("whileLoopResult").textContent =
    whileLoopResult;
26
27   let doWhileLoopResult = "";
28   let doWhileLoopCount = 0;
29   do {
30     doWhileLoopResult += doWhileLoopCount + " ";
31     doWhileLoopCount++;
32   } while (doWhileLoopCount <= 15);
33   document.getElementById("doWhileLoopResult").textContent =
    doWhileLoopResult;
34
35   let forLoopResult = "";
36   for (let forLoopCount = 0; forLoopCount <= 15;
37     forLoopCount++) {
38     forLoopResult += forLoopCount + " ";
39   }
40   document.getElementById("forLoopResult").textContent =
    forLoopResult;
```

Browser Output 1: Display Number  
While Loop  
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Browser Output 2: Display Number  
Do-While Loop  
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Browser Output 3: Display Number  
For Loop  
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

## Exer9

The screenshot shows a web browser window titled "Exercise 9" and a code editor window titled "exer9.html". The browser displays a form with the text "Enter a number:" followed by an input field containing the value "2" and a "Submit" button. Below the input field, the number "8" is displayed. The code editor shows the HTML and JavaScript code for the exercise. The HTML includes a title "Exercise 9", a form with a label "Enter a number:", an input field with id="num1", and a submit button with id="submit". The JavaScript code defines a function "cubeNum()" that takes the value from the input field, calculates its cube, and displays the result in a span with id="result".

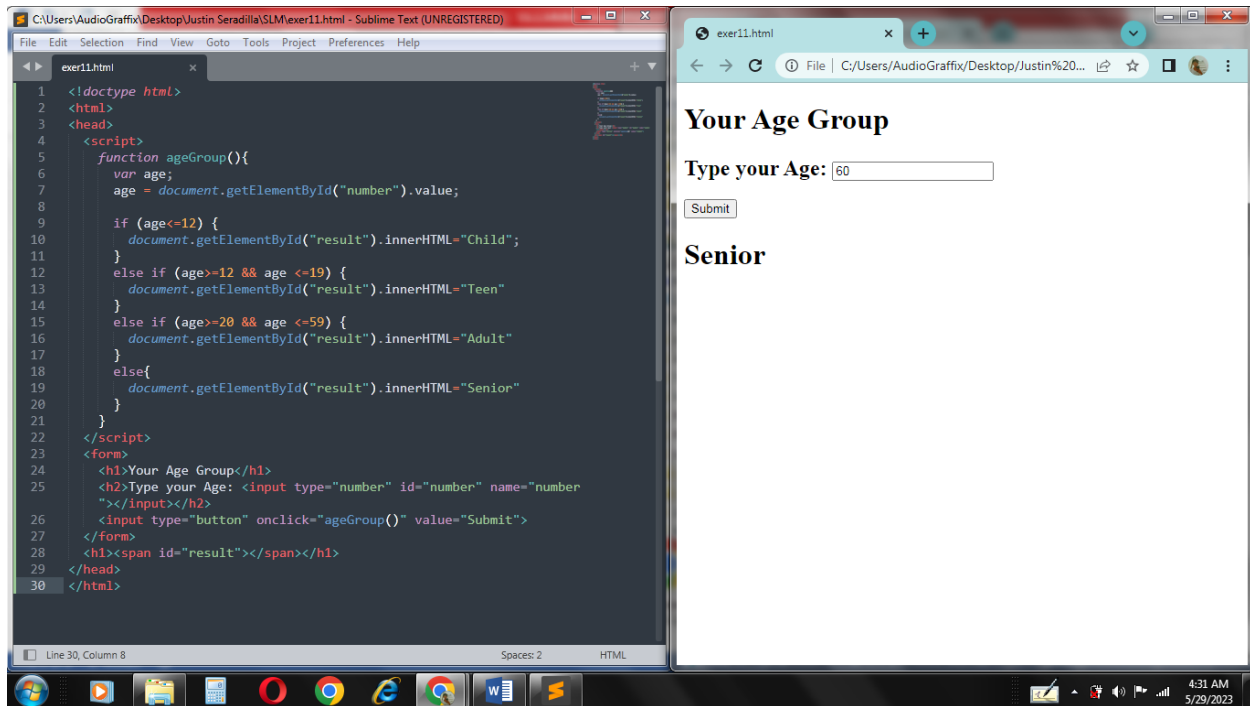
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Exercise 9</title>
5 </head>
6 <body>
7   <form>
8     <div>
9       <label for="cube">Enter a number: </label>
10      <input type="text" placeholder="Enter a number" name="
11      cube" id="num1" required>
12      <input type="button" name="" value="Submit" onclick="
13      cubeNum()" id="submit">
14    </div>
15  </form>
16  <h1><span id="result"></span></h1>
17  <script>
18    function cubeNum() {
19      number = document.getElementById("num1").value;
20      document.getElementById("result").innerHTML = number*number*
21      number;
22    }
23  </script>
24 </body>
25 </html>
```

## Exer10

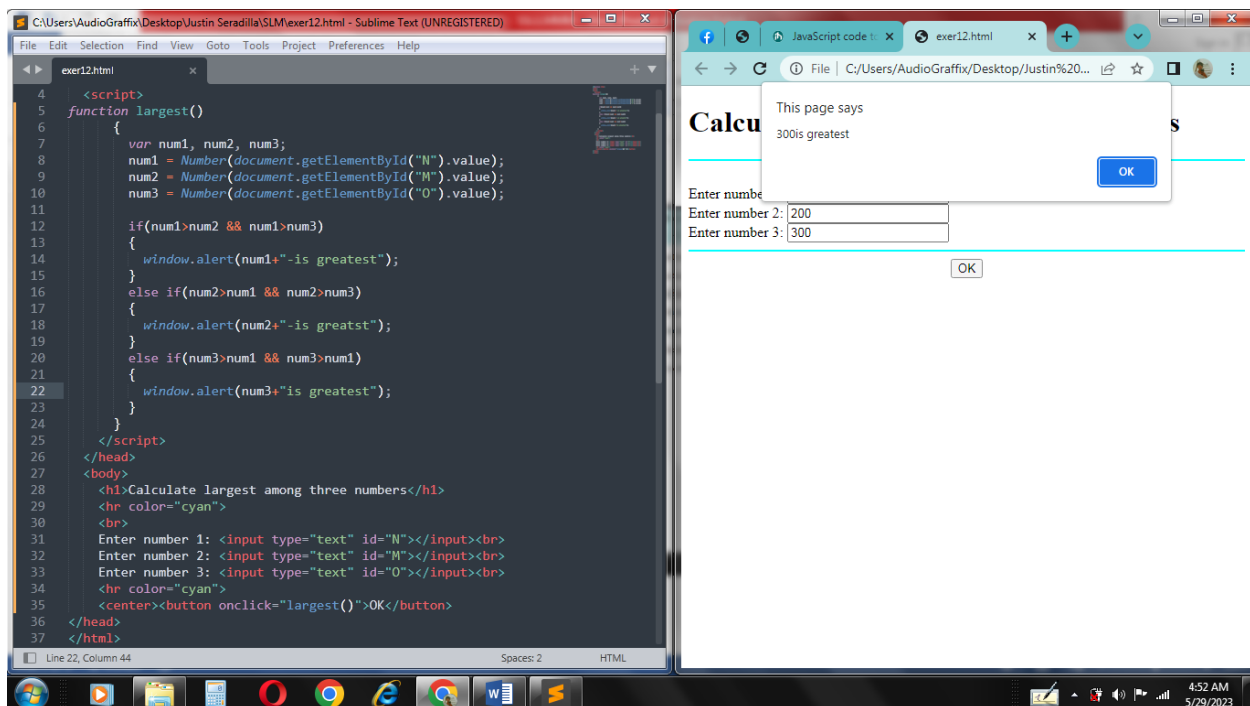
The screenshot shows a web browser window titled "exer10.html" and a code editor window titled "exer10.html". The browser displays a form with two input fields labeled "Enter First Number:" and "Enter Second Number:". Below the input fields, there is a button labeled "Add, Subtract, Multiply, Divide". Below the button, there are four output fields labeled "Addition Result =", "Subtraction Result =", "Multiplication Result =", and "Division Result =", each followed by an input field. The code editor shows the HTML and JavaScript code for the exercise. The HTML includes the input fields, the button, and the output fields. The JavaScript code defines a function "fun()" that takes the values from the input fields, performs the requested operation (Addition, Subtraction, Multiplication, or Division), and displays the result in the corresponding output field.

```
1 <script>
2 var numOne, numTwo, res, temp;
3 function fun()
4 {
5   numOne = parseInt(document.getElementById("one").value);
6   numTwo = parseInt(document.getElementById("two").value);
7   if(numOne && numTwo)
8   {
9     temp = document.getElementById("res");
10    temp.style.display = "block";
11    res = numOne + numTwo;
12    document.getElementById("add").value = res;
13    res = numOne - numTwo;
14    document.getElementById("subtract").value = res;
15    res = numOne * numTwo;
16    document.getElementById("multiply").value = res;
17    res = numOne / numTwo;
18    document.getElementById("divide").value = res;
19  }
20 }
21 </script>
22 </head>
23 <body>
24 <p id="input">Enter First Number: <input id="one">
25 <br/><br/>
26 Enter Second Number: <input id="two"></p>
27 <p><button onclick="fun()">Add, Subtract, Multiply, Divide</button>
28 </p>
29 <p id="res" style="display:none">
30 Addition Result = <input id="add"><br/><br/>
31 Subtraction Result = <input id="subtract"><br/><br/>
32 Multiplication Result = <input id="multiply"><br/><br/>
33 Division Result = <input id="divide"></p>
34 </body>
```

## Exer11



## Exer12





## Exer13

The screenshot shows a Sublime Text editor window titled "exer13.html" containing the following code:

```
3 <script>
4   function largest()
5   {
6     var num1, num2, num3;
7     num1 = Number(document.getElementById("N").value);
8     num2 = Number(document.getElementById("M").value);
9     num3 = Number(document.getElementById("O").value);
10
11     if(num2>num1 && num3>num1)
12     {
13       window.alert(num1+"smallest");
14     }
15     else if(num1>num2 && num3>num2)
16     {
17       window.alert(num2+"is smallest");
18     }
19     else if(num1>num3 && num1>num3)
20     {
21       window.alert(num3+"is smallest");
22     }
23   }
24 </script>
25 </head>
26 <body>
27   <h1>Calculate smallest among three numbers</h1>
28   <hr color="cyan">
29   <br>
30   Enter number 1: <input type="text" id="N"></input><br>
31   Enter number 2: <input type="text" id="M"></input><br>
32   Enter number 3: <input type="text" id="O"></input><br>
33   <hr color="cyan">
34   <center><button onclick="largest()">OK</button>
35 </body>
36 </html>
```

The browser window shows the page with the title "Calculators". An alert box displays "1smallest". The input fields contain "2", "4", and "6".

## Exer14

The screenshot shows a Sublime Text editor window titled "exer14.html" containing the following code:

```
3 <body>
4   <form id="inputForm">
5     <label for="num1">Enter the Firstnumber:</label>
6     <input type="number" id="num1" name="num1" required>
7     <br>
8     <label for="num2">Enter the Secondnumber:</label>
9     <input type="number" id="num2" name="num2" required>
10    <br>
11    <label for="num3">Enter the Thirdnumber:</label>
12    <input type="number" id="num3" name="num3" required>
13    <br>
14    <button type="button" onclick="calculateAverage()">Calculate
15    Average</button>
16  </form>
17
18  <h1 id="result"></h1>
19
20  <script>
21    function calculateAverage() {
22      const num1 = parseInt(document.getElementById('num1').
23      value);
24      const num2 = parseInt(document.getElementById('num2').
25      value);
26      const num3 = parseInt(document.getElementById('num3').
27      value);
28
29      const average = (num1 + num2 + num3) / 3;
30      const resultElement = document.getElementById("result");
31      resultElement.textContent = "The Average is:" + average.
32      toFixed(2);
33    }
34  </script>
35 </body>
36 </html>
```

The browser window shows the page with the title "exer14.html". The input fields contain "2", "4", and "6". The "Calculate Average" button is visible. The result displays "The Average is:4.00".

## Exer15

The image shows a web browser window displaying the output of a JavaScript function. The browser address bar shows the file path: `C:/Users/AudioGrafix/Desktop/Justin%20...`. The page content includes a form with a label "Enter the number:", an input field containing the value "10", and a "Generate Table" button. Below the form, the output of the function is displayed as a list of multiplication facts for the number 10, ranging from 10 x 1 to 10 x 10.

The code editor window (Sublime Text) shows the HTML and JavaScript code for the page. The HTML code includes a form with a label, an input field, and a button. The JavaScript code defines a function `generateMultiplicationTable()` that takes the input value, calculates the multiplication facts, and updates the content of the `result` element.

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <form id="inputForm">
5 <label for="number">Enter the number:</label>
6 <input type="number" id="number" name="number" required>
7 <br>
8 <button type="button" onclick="generateMultiplicationTable()">
  Generate Table</button>
9 </form>
10
11 <h1 id="result"></h1>
12
13 <script>
14   function generateMultiplicationTable() {
15     const number = parseInt(document.getElementById("number")
16       .value);
17
18     let table = "";
19     for (let i = 1; i <= 10; i++) {
20       const result = number * i;
21       table += `${number} x ${i} = ${result}<br>`;
22     }
23     const resultElement = document.getElementById("result");
24     resultElement.innerHTML = table;
25   }
26 </script>
27 </body>
28 </html>
```

Line 18, Column 35      Tab Size: 4      HTML

6:06 AM  
5/29/2023