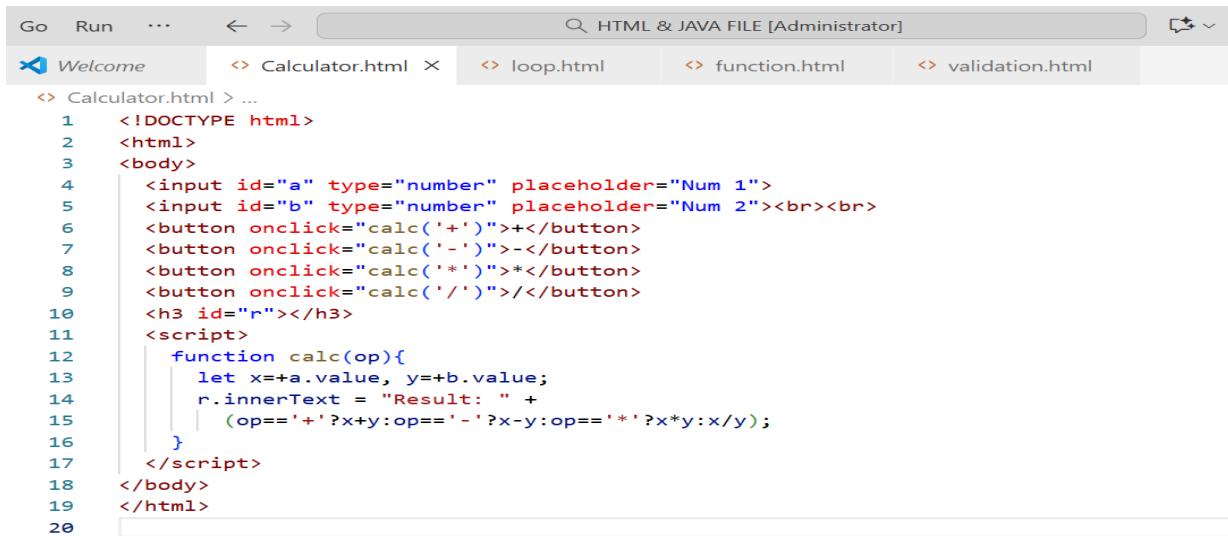


EXPERIMENT-11

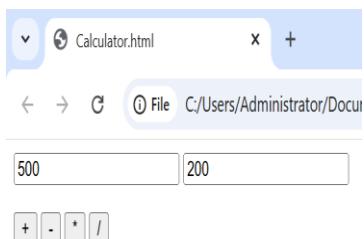
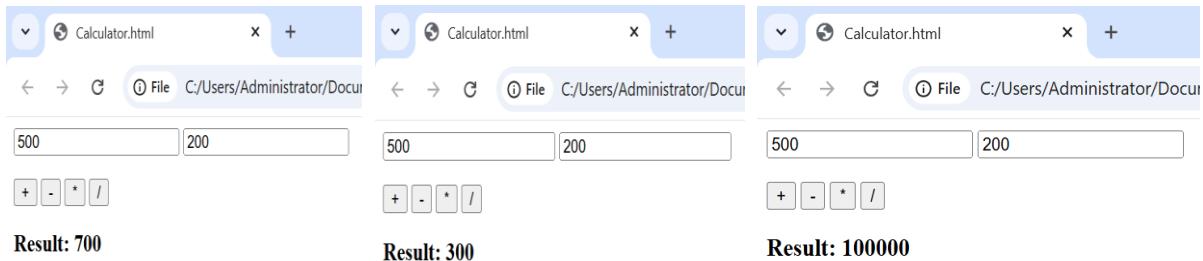
PROBLEM STATEMENT:- Create a JavaScript-enabled web page that performs basic arithmetic operations using input from users.

CODE:-



```
Go Run ... ← → ⌂ HTML & JAVA FILE [Administrator]
Welcome <> Calculator.html <> loop.html <> function.html <> validation.html
<> Calculator.html > ...
1  <!DOCTYPE html>
2  <html>
3  <body>
4      <input id="a" type="number" placeholder="Num 1">
5      <input id="b" type="number" placeholder="Num 2"><br><br>
6      <button onclick="calc('+')">+</button>
7      <button onclick="calc('-')">-</button>
8      <button onclick="calc('*')">*</button>
9      <button onclick="calc('/')">/</button>
10     <h3 id="r"></h3>
11     <script>
12         function calc(op){
13             let x=+a.value, y=+b.value;
14             r.innerText = "Result: " +
15             (op=='+'?x+y:op=='-'?x-y:op=='*'?x*y:x/y);
16         }
17     </script>
18     </body>
19     </html>
20
```

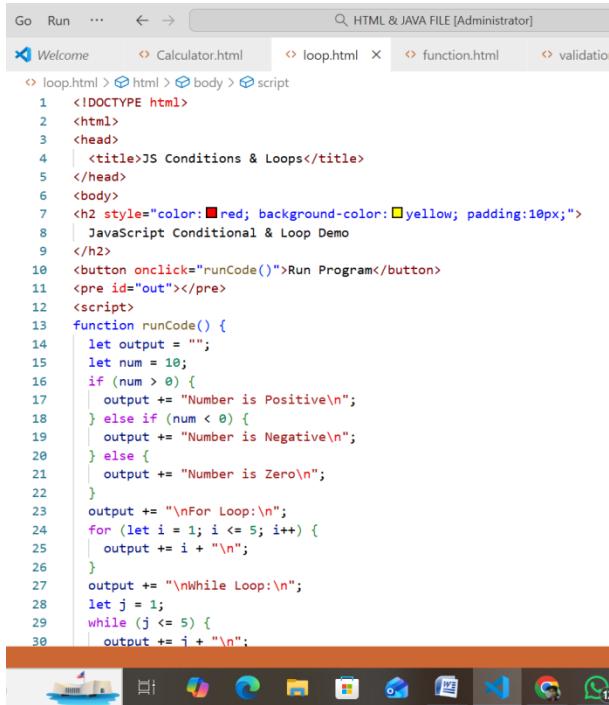
OUTPUT:-



EXPERIMENT-12

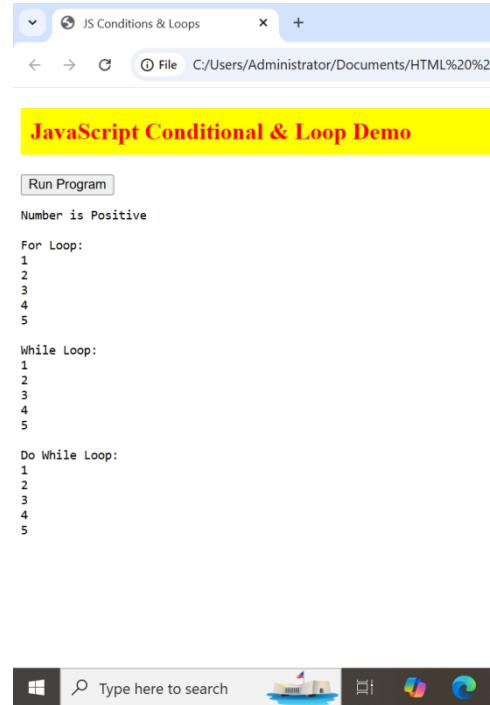
PROBLEM STATEMENT:- Write JavaScript to demonstrate conditional statements and looping (for, while, do...while).

CODE:-



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 | <title>JS Conditions & Loops</title>
5 </head>
6 <body>
7 <h2 style="color: red; background-color: yellow; padding:10px;">
8 | JavaScript Conditional & Loop Demo
9 </h2>
10 <button onclick="runCode()">Run Program</button>
11 <pre id="out"></pre>
12 <script>
13 function runCode() {
14     let output = "";
15     let num = 10;
16     if (num > 0) {
17         output += "Number is Positive\n";
18     } else if (num < 0) {
19         output += "Number is Negative\n";
20     } else {
21         output += "Number is Zero\n";
22     }
23     output += "\nFor Loop:\n";
24     for (let i = 1; i <= 5; i++) {
25         output += i + "\n";
26     }
27     output += "\nWhile Loop:\n";
28     let j = 1;
29     while (j <= 5) {
30         output += j + "\n";
31         j++;
32     }
33     output += "\nDo While Loop:\n";
34     let k = 1;
35     do {
36         output += k + "\n";
37         k++;
38     } while (k <= 5);
39     document.getElementById("out").textContent = output;
40 }
41 </script>
42 </body>
43 </html>
44
```

OUTPUT:-



JavaScript Conditional & Loop Demo

Run Program

Number is Positive

For Loop:

1
2
3
4
5

While Loop:

1
2
3
4
5

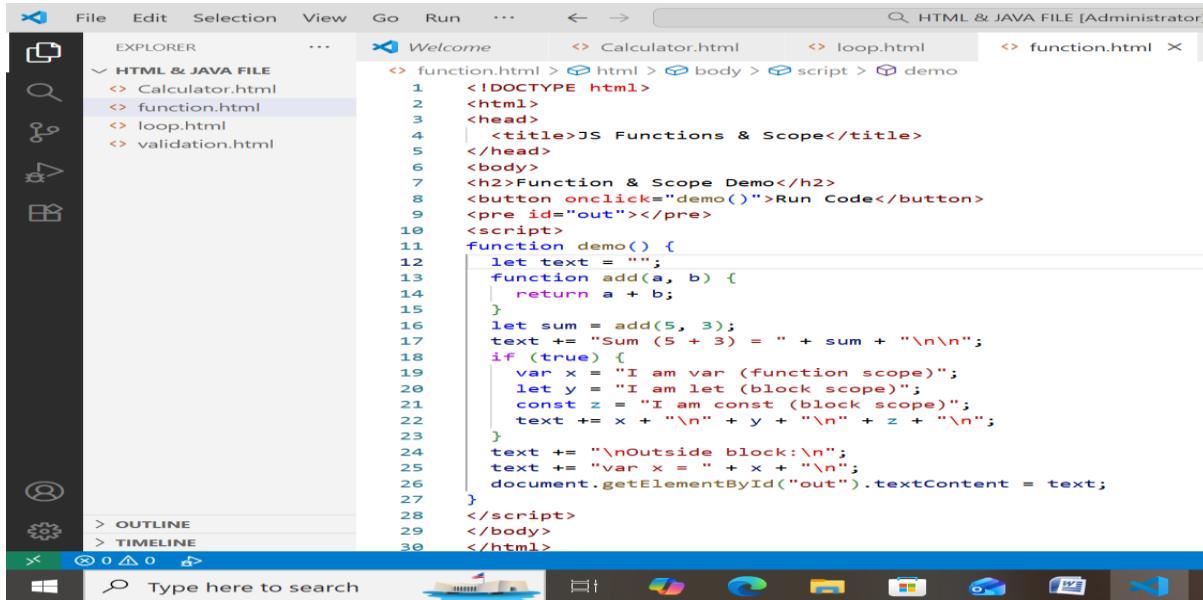
Do While Loop:

1
2
3
4
5

EXPERIMENT-13

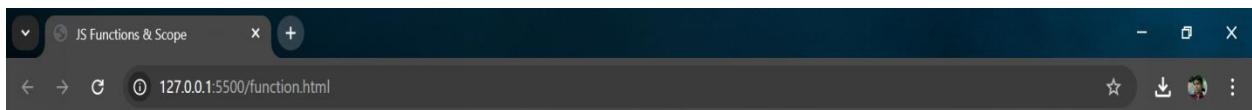
PROBLEM STATEMENT:- Create and invoke user-defined functions in JavaScript; use var, let, and const for scope demonstration.

CODE:-



```
<!DOCTYPE html>
<html>
<head>
    <title>JS Functions & Scope</title>
</head>
<body>
    <h2>Function & Scope Demo</h2>
    <button onclick="demo()">Run Code</button>
    <pre id="out"></pre>
<script>
function demo() {
    let text = "";
    function add(a, b) {
        return a + b;
    }
    let sum = add(5, 3);
    text += "Sum (5 + 3) = " + sum + "\n\n";
    if (true) {
        var x = "I am var (function scope)";
        let y = "I am let (block scope)";
        const z = "I am const (block scope)";
        text += x + "\n" + y + "\n" + z + "\n";
    }
    text += "\nOutside block:\n";
    text += "var x = I am var (function scope)";
    document.getElementById("out").textContent = text;
}
</script>
</body>
</html>
```

OUTPUT:-



Function & Scope Demo

Sum (5 + 3) = 8

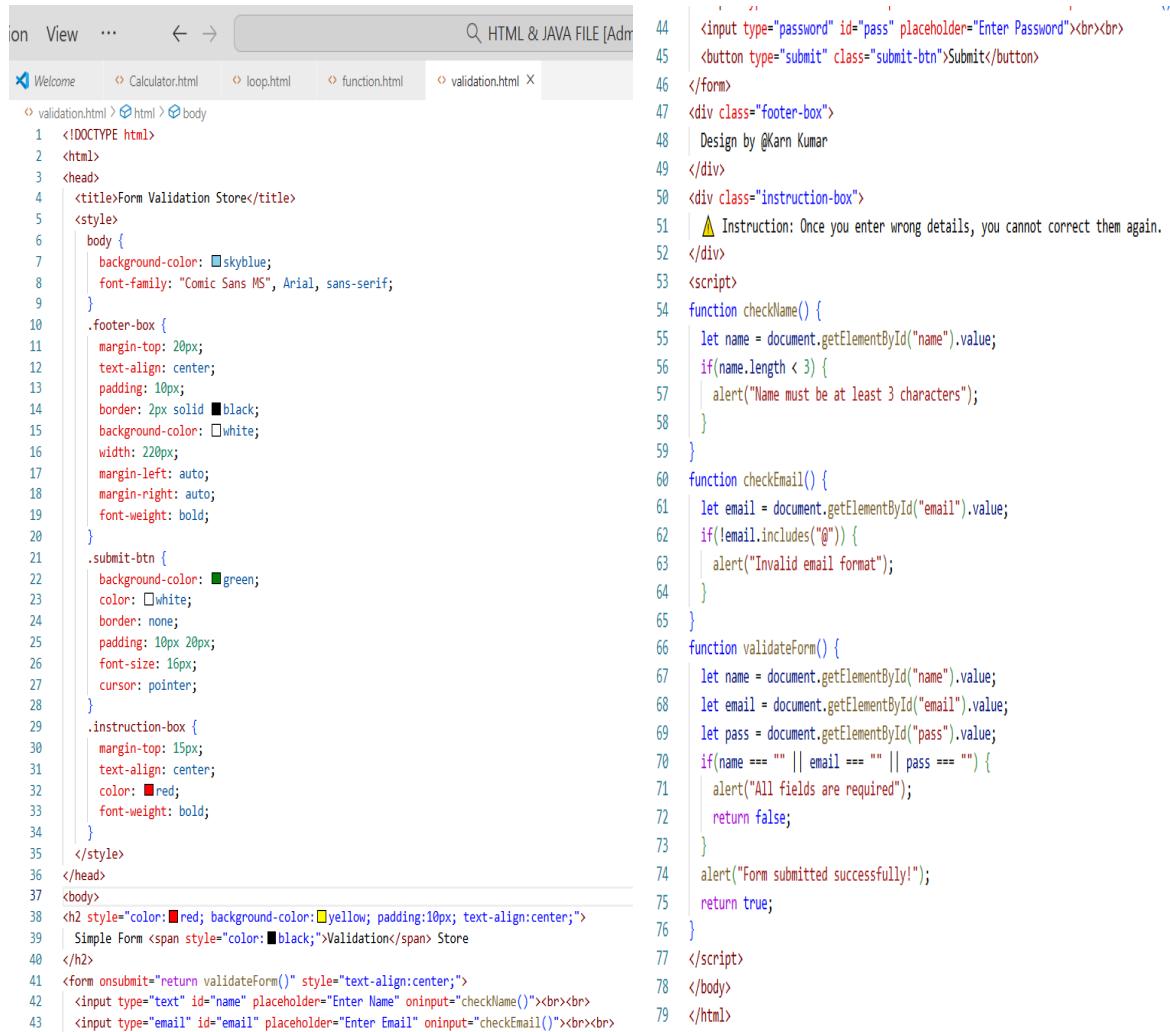
I am var (function scope)
I am let (block scope)
I am const (block scope)

Outside block:
var x = I am var (function scope)

EXPERIMENT-14

PROBLEM STATEMENT:- Handle HTML form validation using JavaScript events like on submit, on input, and use alert() for feedback.

CODE:-



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Form Validation Store</title>
5   <style>
6     body {
7       background-color: #skyblue;
8       font-family: "Comic Sans MS", Arial, sans-serif;
9     }
10    .footer-box {
11      margin-top: 20px;
12      text-align: center;
13      padding: 10px;
14      border: 2px solid #black;
15      background-color: #white;
16      width: 220px;
17      margin-left: auto;
18      margin-right: auto;
19      font-weight: bold;
20    }
21    .submit-btn {
22      background-color: #green;
23      color: #white;
24      border: none;
25      padding: 10px 20px;
26      font-size: 16px;
27      cursor: pointer;
28    }
29    .instruction-box {
30      margin-top: 15px;
31      text-align: center;
32      color: #red;
33      font-weight: bold;
34    }
35  </style>
36 </head>
37 <body>
38 <h2 style="color: #red; background-color: #yellow; padding: 10px; text-align: center;">
39 | Simple Form <span style="color: #black;">Validation</span> Store
40 </h2>
41 <form onsubmit="return validateForm()" style="text-align: center;">
42   <input type="text" id="name" placeholder="Enter Name" oninput="checkName()"><br><br>
43   <input type="email" id="email" placeholder="Enter Email" oninput="checkEmail()"><br><br>
44   <input type="password" id="pass" placeholder="Enter Password"><br><br>
45   <button type="submit" class="submit-btn">Submit</button>
46 </form>
47 <div class="footer-box">
48 | Design by @Karn Kumar
49 </div>
50 <div class="instruction-box">
51 | ▲ Instruction: Once you enter wrong details, you cannot correct them again.
52 </div>
53 <script>
54   function checkName() {
55     let name = document.getElementById("name").value;
56     if(name.length < 3) {
57       alert("Name must be at least 3 characters");
58     }
59   }
60   function checkEmail() {
61     let email = document.getElementById("email").value;
62     if(!email.includes("@")) {
63       alert("Invalid email format");
64     }
65   }
66   function validateForm() {
67     let name = document.getElementById("name").value;
68     let email = document.getElementById("email").value;
69     let pass = document.getElementById("pass").value;
70     if(name === "" || email === "" || pass === "") {
71       alert("All fields are required");
72       return false;
73     }
74     alert("Form submitted successfully!");
75     return true;
76   }
77 </script>
78 </body>
79 </html>
```

OUTPUT:-

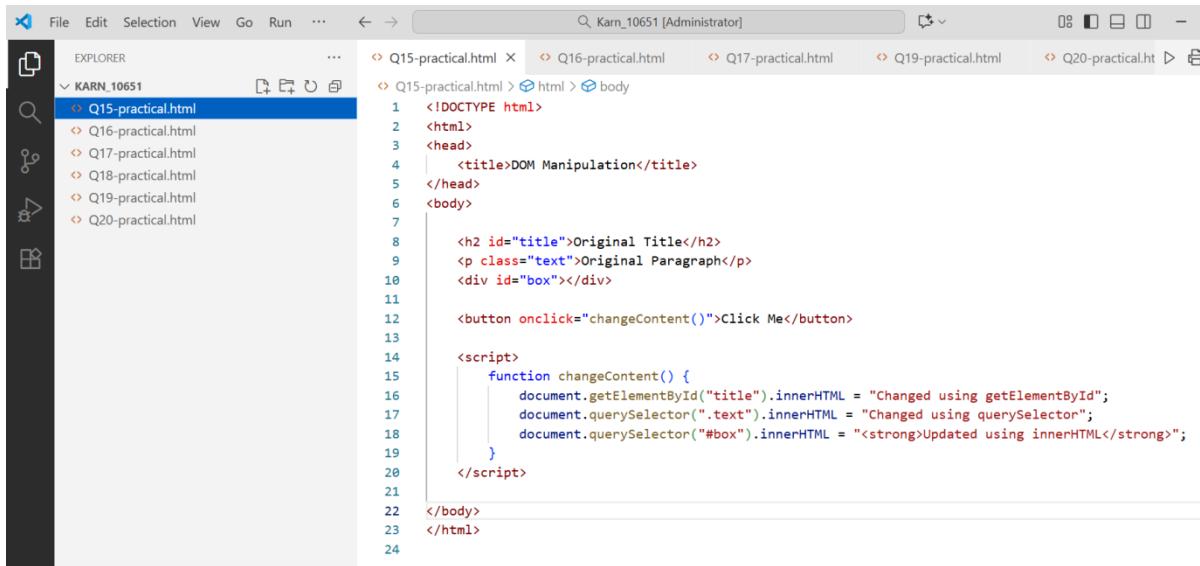
The screenshot shows a web browser window with two tabs: "JS Functions & Scope" and "Form Validation Store". The active tab is "Form Validation Store" at the URL "127.0.0.1:5500/validation.html". The page title is "Simple Form Validation Store". It contains three input fields: "KARN KUMAR" (text), "karnkumarpoddar0987@gm" (text), and "*****" (password). A green "Submit" button is below the inputs. A message box says "Design by @Karn Kumar". At the bottom, a warning message reads "⚠ Instruction: Once you enter wrong details, you cannot correct them again." The browser's taskbar at the bottom shows various pinned icons and the system tray indicates it's 8:29 AM on 12/11/2025.

The screenshot shows the same browser window after a successful form submission. A dark gray modal dialog box appears with the text "127.0.0.1:5500 says" and "Form submitted successfully!". An "OK" button is at the bottom right of the dialog. The rest of the page remains the same, including the input fields, the "Submit" button, the "Design by @Karn Kumar" message, and the warning instruction at the bottom. The browser's taskbar and system tray are visible at the bottom.

EXPERIMENT-15

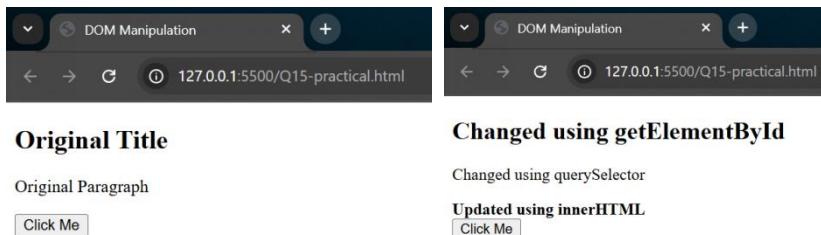
PROBLEM STATEMENT:- Demonstrate DOM manipulation using getElementById(), querySelector(), and innerHTML.

CODE:-



```
File Edit Selection View Go Run ... ← → Q_KARN_10651 [Administrator] EXPLORER KARN_10651 Q15-practical.html Q16-practical.html Q17-practical.html Q18-practical.html Q19-practical.html Q20-practical.html Q15-practical.html > html > body
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>DOM Manipulation</title>
5  </head>
6  <body>
7
8      <h2 id="title">Original Title</h2>
9      <p class="text">Original Paragraph</p>
10     <div id="box"></div>
11
12     <button onclick="changeContent()">Click Me</button>
13
14     <script>
15         function changeContent() {
16             document.getElementById("title").innerHTML = "Changed using getElementById";
17             document.querySelector(".text").innerHTML = "Changed using querySelector";
18             document.querySelector("#box").innerHTML = "<strong>Updated using innerHTML</strong>";
19         }
20     </script>
21
22 </body>
23 </html>
24
```

OUTPUT:-



DOM Manipulation 127.0.0.1:5500/Q15-practical.html

Original Title

Original Paragraph

Click Me

DOM Manipulation 127.0.0.1:5500/Q15-practical.html

Changed using getElementById

Changed using querySelector

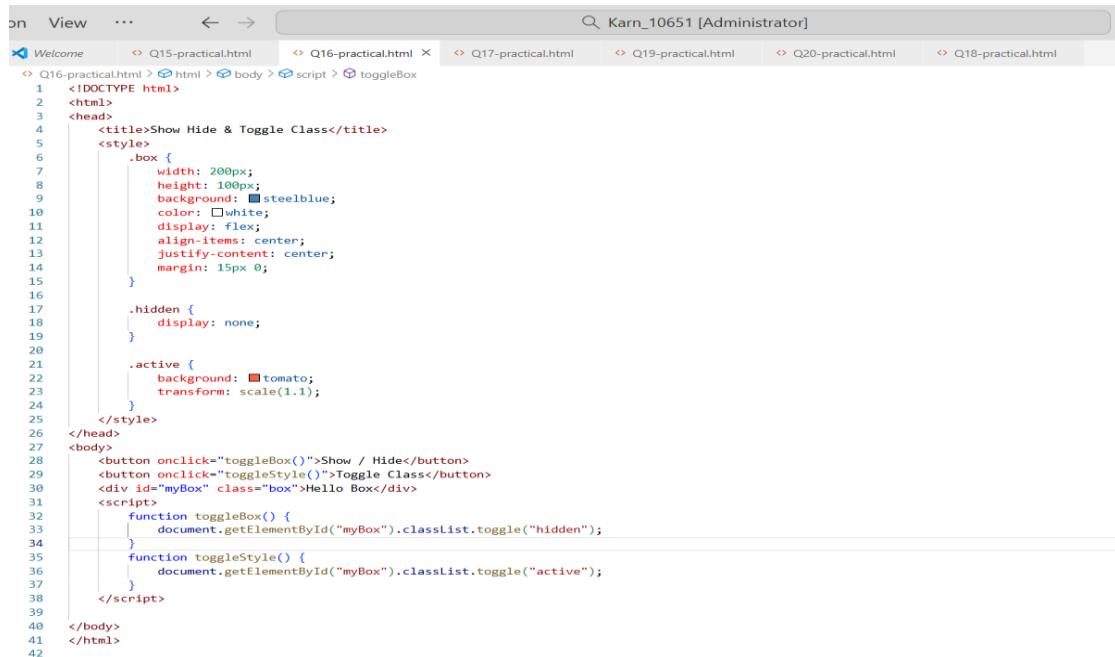
Updated using innerHTML

Click Me

EXPERIMENT-16

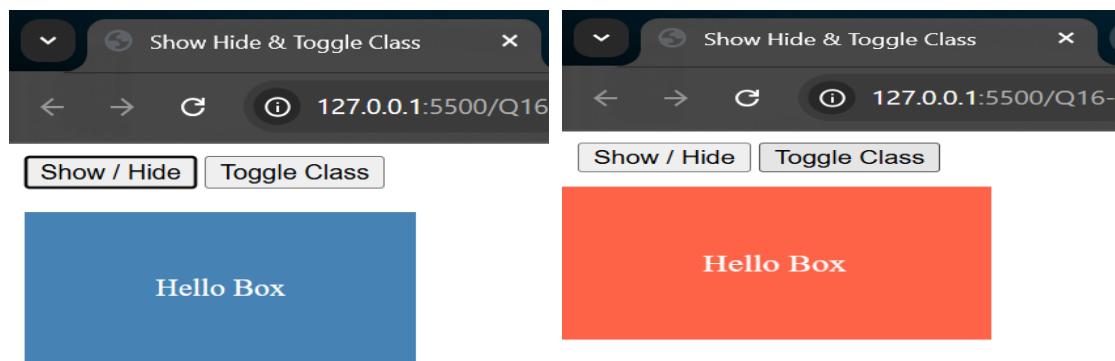
PROBLEM STATEMENT:- Show or hide HTML elements using JavaScript and toggle CSS classes dynamically.

CODE:-



```
on View ... ← → 🔍 Karn_10651 [Administrator]
Welcome Q15-practical.html Q16-practical.html Q17-practical.html Q19-practical.html Q20-practical.html Q18-practical.html
<!DOCTYPE html>
<html>
  <head>
    <title>Show Hide & Toggle Class</title>
    <style>
      .box {
        width: 200px;
        height: 100px;
        background: #steelblue;
        color: #white;
        display: flex;
        align-items: center;
        justify-content: center;
        margin: 15px 0;
      }
      .hidden {
        display: none;
      }
      .active {
        background: #tomato;
        transform: scale(1.1);
      }
    </style>
  </head>
  <body>
    <button onclick="toggleBox()">Show / Hide</button>
    <button onclick="toggleStyle()">Toggle Class</button>
    <div id="myBox" class="box">Hello Box</div>
    <script>
      function toggleBox() {
        document.getElementById("myBox").classList.toggle("hidden");
      }
      function toggleStyle() {
        document.getElementById("myBox").classList.toggle("active");
      }
    </script>
  </body>
</html>
```

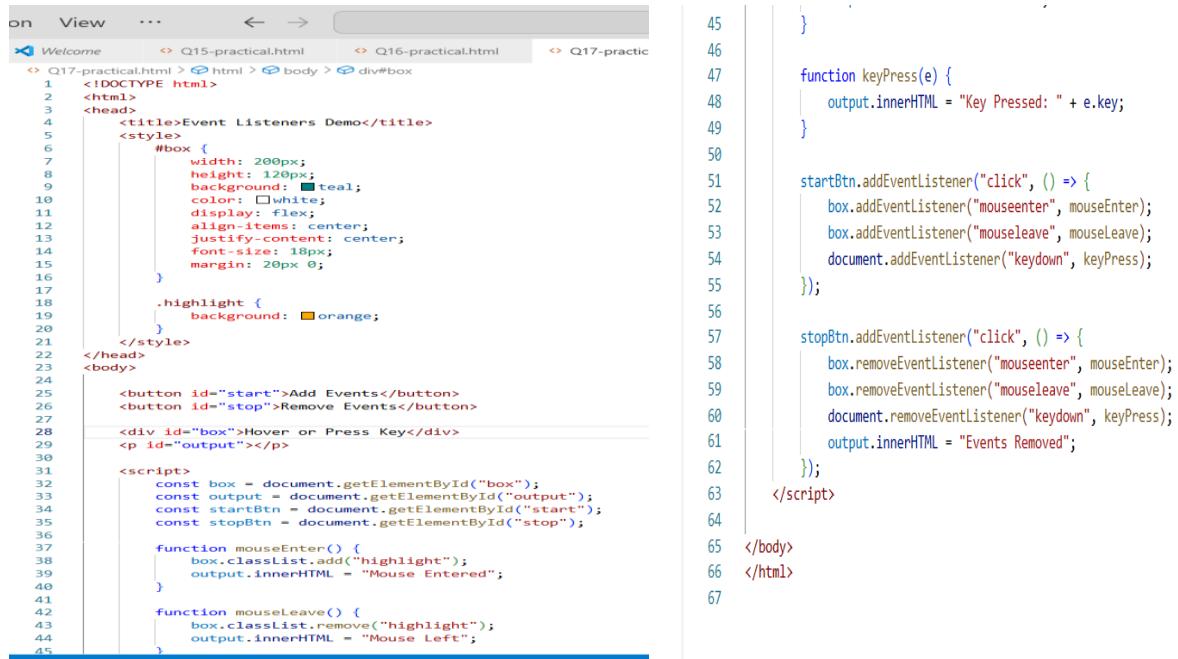
OUTPUT:-



EXPERIMENT-17

PROBLEM STATEMENT:- Add interactivity using event listeners (addEventListener, removeEventListener) for mouse/keyboard Events.

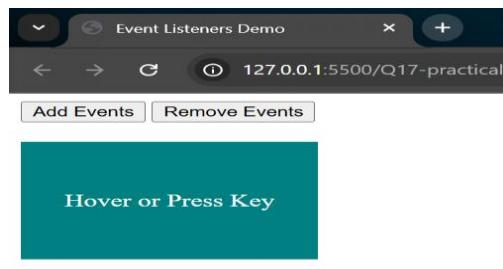
CODE:-



The screenshot shows a code editor with two panes. The left pane displays the HTML code for a web page titled "Event Listeners Demo". It includes a CSS style block for a "#box" element, which has a width of 200px, height of 120px, background color of teal, and white text. The HTML contains a button to add events, a button to remove events, and a "box" div containing a "p" tag with the text "Hover or Press Key". The right pane displays the corresponding JavaScript code. It defines functions for "mouseEnter" and "mouseLeave" events on the "#box" element, and "keyPress" and "stop" events on the "start" and "stop" buttons respectively. It also includes code to add and remove event listeners for these specific events.

```
on View ... ← →
Welcome Q15-practical.html Q16-practical.html Q17-practical.html
Q17-practical.html > html > body > div#box
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Event Listeners Demo</title>
5   <style>
6     #box {
7       width: 200px;
8       height: 120px;
9       background: teal;
10      color: white;
11      display: flex;
12      align-items: center;
13      justify-content: center;
14      font-size: 18px;
15      margin: 20px 0;
16    }
17
18    .highlight {
19      background: orange;
20    }
21  </style>
22 </head>
23 <body>
24
25   <button id="start">Add Events</button>
26   <button id="stop">Remove Events</button>
27
28   <div id="box">Hover or Press Key</div>
29   <p id="output"></p>
30
31 <script>
32   const box = document.getElementById("box");
33   const output = document.getElementById("output");
34   const startBtn = document.getElementById("start");
35   const stopBtn = document.getElementById("stop");
36
37   function mouseEnter() {
38     box.classList.add("highlight");
39     output.innerHTML = "Mouse Entered";
40   }
41
42   function mouseLeave() {
43     box.classList.remove("highlight");
44     output.innerHTML = "Mouse Left";
45
46
47   function keyPress(e) {
48     output.innerHTML = "Key Pressed: " + e.key;
49   }
50
51   startBtn.addEventListener("click", () => {
52     box.addEventListener("mouseenter", mouseEnter);
53     box.addEventListener("mouseleave", mouseLeave);
54     document.addEventListener("keydown", keyPress);
55   });
56
57   stopBtn.addEventListener("click", () => {
58     box.removeEventListener("mouseenter", mouseEnter);
59     box.removeEventListener("mouseleave", mouseLeave);
60     document.removeEventListener("keydown", keyPress);
61     output.innerHTML = "Events Removed";
62   });
63 </script>
64
65 </body>
66 </html>
67
```

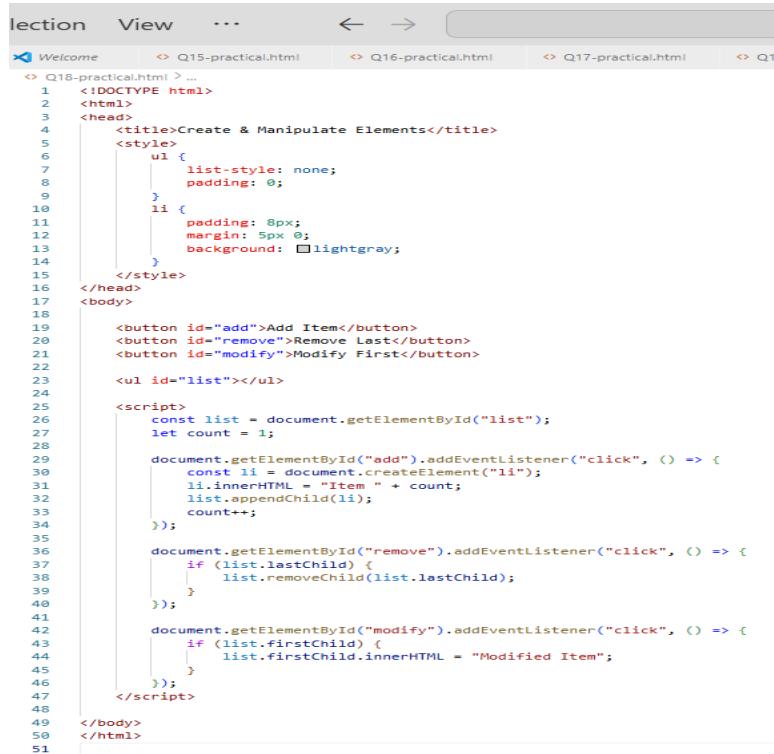
OUTPUT:-



EXPERIMENT-18

PROBLEM STATEMENT:- Create and manipulate elements using JavaScript: append, remove, or modify child nodes.

CODE:-



```
lection View ... ← → 
Q18-practical.html < Q15-practical.html > Q16-practical.html < Q17-practical.html > Q18-practical.html < Q19-practical.html > 
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Create & Manipulate Elements</title>
5   <style>
6     ul {
7       list-style: none;
8       padding: 0;
9     }
10    li {
11      padding: 8px;
12      margin: 5px 0;
13      background: #f0f0f0;
14    }
15  </style>
16 </head>
17 <body>
18
19   <button id="add">Add Item</button>
20   <button id="remove">Remove Last</button>
21   <button id="modify">Modify First</button>
22
23   <ul id="list"></ul>
24
25   <script>
26     const list = document.getElementById("list");
27     let count = 1;
28
29     document.getElementById("add").addEventListener("click", () => {
30       const li = document.createElement("li");
31       li.innerHTML = "Item " + count;
32       list.appendChild(li);
33       count++;
34     });
35
36     document.getElementById("remove").addEventListener("click", () => {
37       if (list.lastChild) {
38         list.removeChild(list.lastChild);
39       }
40     });
41
42     document.getElementById("modify").addEventListener("click", () => {
43       if (list.firstChild) {
44         list.firstChild.innerHTML = "Modified Item";
45       }
46     });
47   </script>
48
49 </body>
50 </html>
```

OUTPUT:-



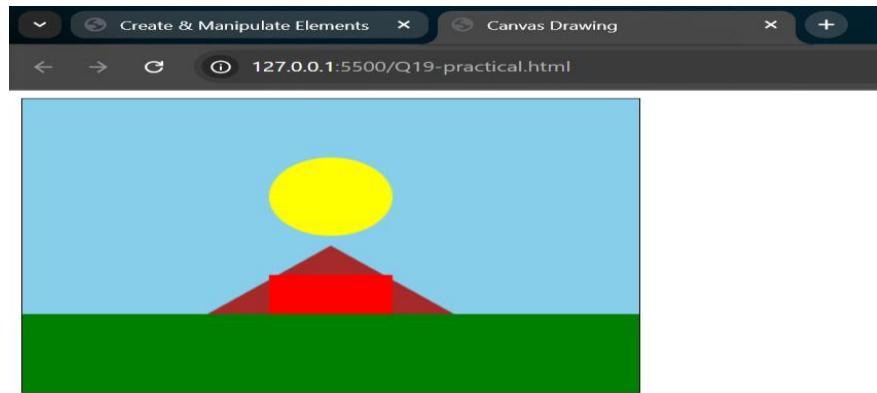
EXPERIMENT-19

PROBLEM STATEMENT:- Create a simple drawing using the <canvas> element: draw shapes and fill colors.

CODE:-

```
File View ... ← → ⌂ Karn_10651 [Admin]
Welcome Q15-practical.html Q16-practical.html Q17-practical.html Q19-practical.html ×
Q19-practical.html > ...
1  <!DOCTYPE html>
2  <html>
3  |   <head>
4  |       <title>Canvas Drawing</title>
5  |   </head>
6  |   <body>
7
8      <canvas id="myCanvas" width="400" height="300" style="border:1px solid black;"></canvas>
9
10     <script>
11         const canvas = document.getElementById("myCanvas");
12         const ctx = canvas.getContext("2d");
13
14         ctx.fillStyle = "skyblue";
15         ctx.fillRect(0, 0, 400, 300);
16
17         ctx.fillStyle = "green";
18         ctx.fillRect(0, 220, 400, 80);
19
20         ctx.beginPath();
21         ctx.arc(200, 100, 40, 0, Math.PI * 2);
22         ctx.fillStyle = "yellow";
23         ctx.fill();
24
25         ctx.beginPath();
26         ctx.moveTo(120, 220);
27         ctx.lineTo(200, 150);
28         ctx.lineTo(280, 220);
29         ctx.closePath();
30         ctx.fillStyle = "brown";
31         ctx.fill();
32
33         ctx.fillStyle = "red";
34         ctx.fillRect(160, 180, 80, 40);
35
36     </script>
37
38  </body>
39  </html>
```

OUTPUT:-



EXPERIMENT-20

PROBLEM STATEMENT:- Initialize a Git repository, commit changes, and push to GitHub; Deploy a personal portfolio site on GitHub Pages or Netlify.

OUTPUT:-

Already post in the github website and netlify.

