

Session 4

JavaScript HTML DOM

The **Document Object Model (DOM)** is a W3C-standard, language-independent interface that allows programs and scripts to dynamically access, modify, add, or remove the content, structure, and style of a web document. While the W3C DOM is categorized into Core DOM (all document types), XML DOM, and **HTML DOM**, this session focuses specifically on the HTML DOM, which is the standard model for HTML documents.

When a browser loads an HTML page, it automatically parses the markup and constructs a hierarchical **DOM tree**. This tree represents the document's structure through parent-child node relationships, where:

- The entire document is a `document` node
- Every HTML tag is an `element` node
- Text inside elements is a `text` node
- HTML comments are `comment` nodes

The HTML DOM bridges static HTML with dynamic JavaScript by treating every element, attribute, and piece of content as a **JavaScript object**. This object-oriented representation enables developers to:

-  Access or locate any element using IDs, tag names, or class names
-  Modify element content (innerHTML), attributes, and CSS styles on-the-fly
-  Dynamically create, append, or remove HTML elements
-  React to user interactions through DOM events (clicks, inputs, loads, etc.)

By leveraging the HTML DOM, you can transform static web pages into highly interactive, real-time applications without requiring page reloads. This session introduces the core DOM programming interface, node navigation techniques, and practical methods for manipulating HTML content and styles using JavaScript.

Objective

✗ Do not copy. Read for understanding and the viva

After completing this session one should be able to:

- Add, Modify or delete the existing content of HTML elements.
- Modify a CSS style in the HTML page.
- React to existing HTML DOM events in page.

DOM Tree (Node Types)

When a browser parses HTML, it builds a node tree:

Node Type	Description	Example
document	Root node of the entire page	document
element	HTML tags	<div> , <p> ,
text	Content inside elements	"Hello World" inside <p>
attribute	Element attributes (treated as nodes in legacy DOM)	id="btn" , src="img.jpg"
comment	HTML comments	<!-- hidden -->

Parent-Child-Sibling Relationships

- <html> → parent of <head> & <body>
- <body> → parent of <h1> & <p>
- <h1> & <p> → siblings

Accessing Elements (DOM Methods)

Method	Returns	Best For
<code>document.getElementById("id")</code>	Single element	Fastest & most reliable
<code>document.getElementsByClassName("class")</code>	HTMLCollection (Live)	Multiple elements sharing a class
<code>document.getElementsByTagName("tag")</code>	HTMLCollection (Live)	All elements of a specific tag
<code>document.querySelector("selector")</code>	First matching element	Modern, CSS-like selectors
<code>document.querySelectorAll("selector")</code>	NodeList (static)	All matching elements

Modifying Element

Content

```
1 element.innerHTML = "<b>New HTML</b>"; // Parses HTML (risk of XSS if user
2 element.textContent = "Plain text only"; // Safer, faster, ignores HTML tag
3 element.innerText = "Visible text only"; // Respects CSS styling (e.g., hid
```

Attributes

```
1 // Direct property (works for standard attributes)
2 img.src = "new.jpg";
3 link.href = "https://example.com";
4
5 // Universal method (works for custom/data attributes)
6 element.setAttribute("data-role", "admin");
7 let val = element.getAttribute("data-role");
```

Styles

```
1 // JS uses camelCase for multi-word CSS properties
2 element.style.backgroundColor = "lightblue";
3 element.style.fontSize = "18px";
4 element.style.marginLeft = "20px";
```

JS

Create & Append Element

```
1 let newP = document.createElement("p");           // <p></p>
2 let txt = document.createTextNode("Hello DOM");    // "Hello DOM"
3 newP.appendChild(txt);                             // <p>Hello DOM</p>
4 document.body.appendChild(newP);                  // Adds to end of <body>
```

JS

Insert at Specific Position

```
1 let parent = document.getElementById("container");
2 let reference = document.getElementById("refItem");
3 parent.insertBefore(newP, reference); // Inserts newP before reference
```

JS

Remove Elements

```
1 // Modern (recommended)
2 element.remove();
3
4 // Legacy (manual's syntax)
5 parent.removeChild(childElement);
```

JS

Suggestion

See Example 1 in Lab manual. It covers all the main methods mentioned above.

Question 1

Problem Statement

■ Write in lab record

Design an HTML page using JavaScript as shown below. When Try button is pressed, the Full Name should be shown below the Try button.

(Hint: Use concatenation to get Full Name).

(Please refer to images in lab manual)

Preview (Working)

✕ Do not copy. Read for understanding and the viva

Ex 1: Concatenate Full Name

Code

■ Write in lab record

button-on-click.html

```
1 <!doctype html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <title>Session 4: JavaScript DOM Exercises</title>
6     <style>
7       body {
8         font-family: "Segoe UI", sans-serif;
9         margin: 20px;
10        background: #f4f4f4;
11      }
12      .exercise {
13        background: #fff;
14        padding: 20px;
15        margin-bottom: 20px;
16        border-radius: 8px;
17        box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
18      }
19      input,
20      button {
21        padding: 8px 12px;
22        margin: 5px;
23        border: 1px solid #ccc;
24        border-radius: 4px;
25      }
26      button {
27        cursor: pointer;
28        background: #0056b3;
29        color: #fff;
30        border: none;
31      }
32      button:hover {
33        background: #003d82;
34      }
35      img {
36        border: 2px solid #333;
37        border-radius: 4px;
38      }
39    </style>
40  </head>
41  <body>
42    <h1 style="text-align: center">
43      Session 4: JavaScript HTML DOM Exercises
```

```
44     </h1>
45     <div class="exercise">
46         <h2>Ex 1: Concatenate Full Name</h2>
47         <input type="text" id="fname" placeholder="First Name" />
48         <input type="text" id="lname" placeholder="Last Name" />
49         <button id="btnTry">Try</button>
50         <p
51             id="fullNameOutput"
52             style="font-weight: bold; margin-top: 10px"
53         ></p>
54     </div>
55
56     <script>
57         // DOM Access + innerHTML + Concatenation
58         document.getElementById("btnTry").onclick = function () {
59             const fn = document.getElementById("fname").value;
60             const ln = document.getElementById("lname").value;
61             document.getElementById("fullNameOutput").innerHTML =
62                 `Full Name: ${fn} ${ln}`;
63         };
64     </script>
65 </body>
66 </html>
```

Question 2

Problem Statement

■ Write in lab record

Modify the below snippet of the code to change the src attribute of the image element to "pqr.jpg" using HTML DOM.

```
1 
2 <script>
3 //Write your code here.
4 </script>
```

HTML

Preview (Working)

✕ Do not copy. Read for understanding and the viva

Ex 2: Change Image `src`



Change to opr.jpg

Code

■ Write in lab record

image-change.html

```
1 <!doctype html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <title>Session 4: JavaScript DOM Exercises</title>
6     <style>
7       body {
8         font-family: "Segoe UI", sans-serif;
9         margin: 20px;
10        background: #f4f4f4;
11      }
12      .exercise {
13        background: #fff;
14        padding: 20px;
15        margin-bottom: 20px;
16        border-radius: 8px;
17        box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
18      }
19      input,
20      button {
21        padding: 8px 12px;
22        margin: 5px;
23        border: 1px solid #ccc;
24        border-radius: 4px;
25      }
26      button {
27        cursor: pointer;
28        background: #0056b3;
29        color: #fff;
30        border: none;
31      }
32      button:hover {
33        background: #003d82;
34      }
35      img {
36        border: 2px solid #333;
37        border-radius: 4px;
38      }
39    </style>
40  </head>
41  <body>
42    <h1 style="text-align: center">
43      Session 4: JavaScript HTML DOM Exercises
```

```
44     </h1>
45
46     <div class="exercise">
47         <h2>Ex 2: Change Image `src` Attribute</h2>
48         <br />
53         <button id="btnChangeImg">Change to opr.jpg</button>
54     </div>
55
56     <script>
57         document.getElementById("btnChangeImg").onclick = function () {
58             document.getElementById("pic").src =
59                 "https://placeholder.co/150x100/FF0000/FFFFFF?text=pqr.jpg"
60         };
61     </script>
62 </body>
63 </html>
```

Question 3

Problem Statement

■ Write in lab record

Write the JavaScript code to change the color of the HTML element h2 to green, on click the change color button.

Preview (Working)

✕ Do not copy. Read for understanding and the viva

Ex 3: Change H2 Color

Click button to change my color

Change Color

Code

■ Write in lab record

color-change.html

```
1 <!doctype html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <title>Session 4: JavaScript DOM Exercises</title>
6     <style>
7       body {
8         font-family: "Segoe UI", sans-serif;
9         margin: 20px;
10        background: #f4f4f4;
11      }
12      .exercise {
13        background: #fff;
14        padding: 20px;
15        margin-bottom: 20px;
16        border-radius: 8px;
17        box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
18      }
19      input,
20      button {
21        padding: 8px 12px;
22        margin: 5px;
23        border: 1px solid #ccc;
24        border-radius: 4px;
25      }
26      button {
27        cursor: pointer;
28        background: #0056b3;
29        color: #fff;
30        border: none;
31      }
32      button:hover {
33        background: #003d82;
34      }
35      img {
36        border: 2px solid #333;
37        border-radius: 4px;
38      }
39    </style>
40  </head>
41  <body>
42    <h1 style="text-align: center">
43      Session 4: JavaScript HTML DOM Exercises
```

```
44     </h1>
45     <div class="exercise">
46         <h2>Ex 3: Change H2 Color</h2>
47         <h2 id="heading1" style="color: red">
48             This heading will turn green
49         </h2>
50         <button id="btnChangeColor">Change Color</button>
51     </div>
52
53     <script>
54         document.getElementById("btnChangeColor").onclick = function ()
55             document.getElementById("heading1").style.color = "green";
56         };
57     </script>
58 </body>
59 </html>
```

[✎ Edit this page](#)

[< Previous](#)
Session 3

Session 5 [Next >](#)