**Question 1: What is the DOM (Document Object Model) in JavaScript? How does JavaScript interact with the DOM?**

The DOM (Document Object Model) is an API that represents and interacts with any HTML or XML -based markup language document. The DOM is a document model loaded in the browser and representing the document as a node tree, or DOM tree, where each node represents part of the document

**Key Features of the DOM:**

1. **Tree Structure** → The DOM represents the HTML document as a tree of nodes.
2. **Interactivity** → JavaScript can manipulate elements, attributes, and styles dynamically.
3. **Platform-Independent** → The DOM is a standard API supported by all modern browsers.

**How JavaScript Interacts with the DOM:**

JavaScript uses the DOM API to access, modify, add, or delete elements on a webpage.

**1. Selecting Elements**

JavaScript provides methods to select elements from the DOM:

// Select an element by ID

let heading = document.getElementById("main-title");

// Select elements by class

let items = document.getElementsByClassName("list-item");

// Select elements using querySelector

let firstItem = document.querySelector(".list-item");

let allItems = document.querySelectorAll(".list-item");

**2. Modifying Elements**

JavaScript can change content, attributes, and styles of elements.

// Change text content

heading.textContent = "New Heading";

// Change an attribute

heading.setAttribute("class", "highlight");

// Modify styles

heading.style.color = "blue";

heading.style.fontSize = "24px";

**3. Adding and Removing Elements**

JavaScript can dynamically create or remove elements.

// Create a new paragraph element

let newParagraph = document.createElement("p");

newParagraph.textContent = "This is a new paragraph!";

document.body.appendChild(newParagraph); // Add it to the body

// Remove an existing element

let oldElement = document.getElementById("old-div");

oldElement.remove();

**4. Handling Events**

JavaScript can respond to user interactions through event listeners.

document.getElementById("myButton").addEventListener("click", function() {

alert("Button Clicked!");

});

**Question 2: Explain the methods getElementById(), getElementsByClassName(),and querySelector() used to select elements from the DOM.**

**Methods to Select Elements from the DOM in JavaScript**

JavaScript provides several methods to select elements from the Document Object Model (DOM). Here, we'll discuss three commonly used methods:

1. **getElementById()**

This method selects a single element based on its **ID**. Since IDs are unique, it always returns a **single element (or null if not found).**

**Syntax:**

document.getElementById("id");

**Example:**

let heading = document.getElementById("main-title");

heading.textContent = "New Title"; // Modifies the text

1. **getElementsByClassName()**

This method selects all elements that have a specified class name. It returns an HTMLCollection, which is similar to an array but doesn't support all array methods.

**Syntax:**

document.getElementsByClassName("className");

**Example:**

let items = document.getElementsByClassName("list-item");

// Loop through the elements and modify them

for (let i = 0; i < items.length; i++) {

items[i].style.color = "blue";

}

1. **querySelector()**

This method selects the first matching element based on a CSS selector (e.g., #id, .class, tag).

**Example:**

let firstItem = document.querySelector(".list-item");

firstItem.style.fontWeight = "bold";

**Comparison Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Returns** | **Selection Type** | **Example** |
| **getElementById()** | **Single element** | **By ID** | **document.getElementById("header")** |
| **getElementsByClassName()** | **HTMLCollection (like an array)** | **By Class Name** | **document.getElementsByClassName("items")** |
| **querySelector()** | **First matching element** | **By CSS Selector** | **document.querySelector(".container div")** |

Question 2: Explain the methods getElementById(), getElementsByClassName(),and querySelector() used to select elements from the DOM.