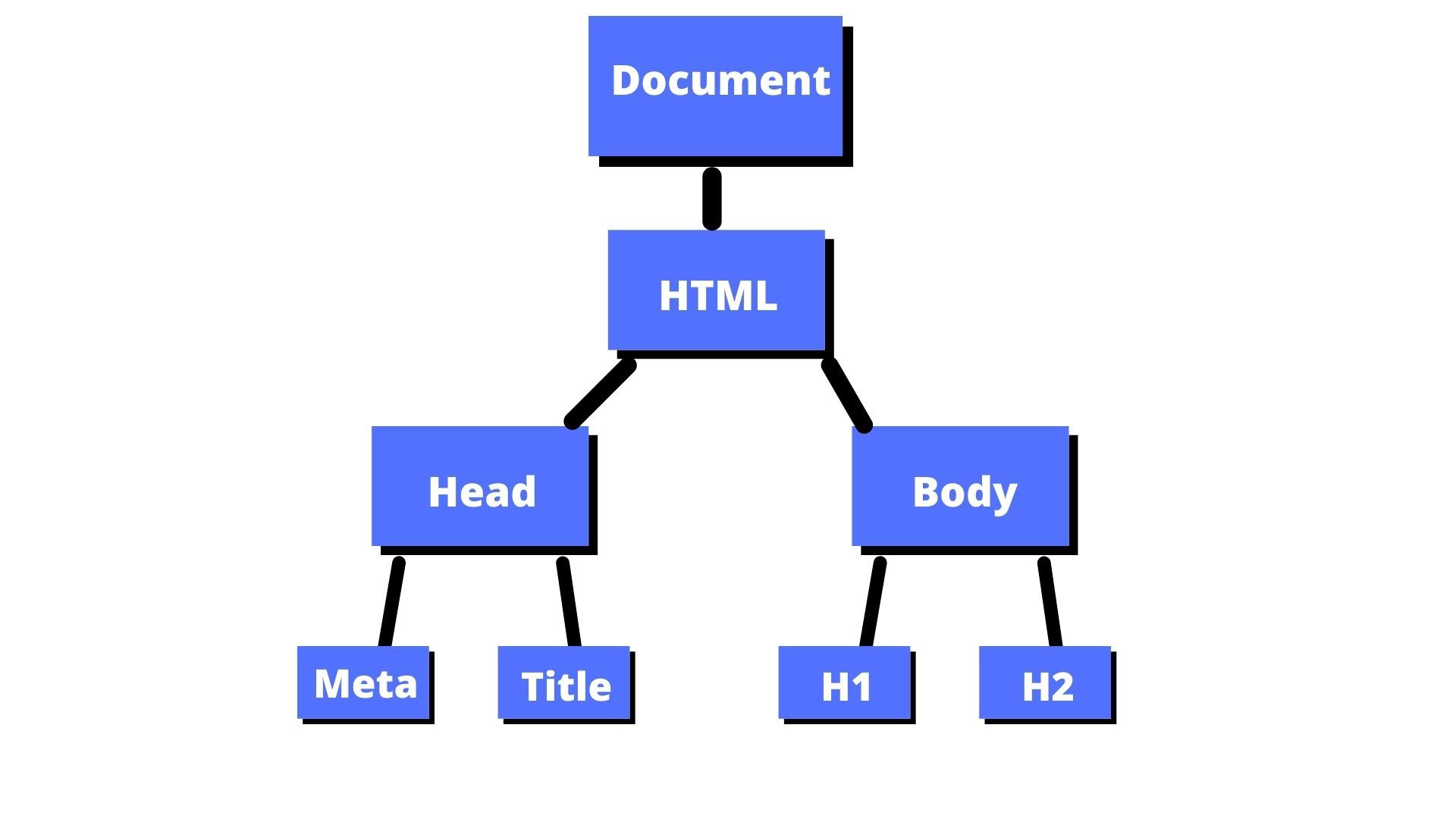
**Q1: What is DOM?**  
Ans: In the context of web development, "DOM" stands for Document Object Model. It's a programming interface for web documents. The DOM represents the structure of an HTML document as a tree-like model where each node represents a part of the document.



**Q2: What is element in HTML?**

**Ans:** In HTML (Hypertext Markup Language), an "element" refers to the individual components that make up the structure of a web page. Each element is enclosed within tags, which define its beginning and end. Elements can include things like text, images, links, forms, tables, and more.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <h1>"Hello world" </h1>

    <script>

    </script>

</body>

</html>

In this code:

<html> is the root element, which contains all other elements on the page.

**<head>** contains metadata about the document, such as the title in the <title> element.

**<body>** contains the main content of the document, such as headings <h1> paragraphs.

**Q3: What is DOM manipulation?**

**Ans:** The technique of dynamically changing a web page's structure, content, or style with JavaScript is known as DOM manipulation. It entails actions such as adding, deleting, and updating items as well as altering text content and stylistic elements. By adding event listeners to items and initiating actions in response, developers may react to user interactions. Interactive and responsive web apps can be made thanks to this dynamic DOM modification.

**Q4: What are different ways of manipulating DOM in javascript?**

**Ans:** It have serval method.

**1-QuerySelector and querySelectorAll:**

const element = document.querySelector('.className');

const elements = document.querySelectorAll('p');

**2- CreateElement:**

const newElement = document.createElement('div');

**3- AppendChlid and insertBefore:**

parentElement.appendChild(newElement);

parentElement.insertBefore(newElement, referenceElement);

**4- AddEventListener:**

const button = document.getElementById('myButton');

button.addEventListener('click', () => {

       console.log('Button clicked!');

});

**Q5: What are different ways of accessing DOM in javascript? Please learn about all different selectors of accessing the DOM?**

**Ans:** It have serval method.

1. **GetElementByid:**

To select an element by its unique ID attribute.

const element = document.getElementById('elementId');

1. **getElementByClassName:**

Collection of elements that have specific class name.

const elements = document.getElementsByClassName('className');

1. **getElementsByTagName:**

Select a element by specific tag.

const elements = document.getElementsByTagName('tagname');

1. **querySelector:**

Select the first element that matches a specified css selector

const element = document.querySelector('.className');

**Q6: How to change style of a DOM element using javascript?**

Ans: const element = document.getElementById('myElement');

          element.style.color = 'green';

         element.style.backgroundColor = 'black';

         element.style.fontSize = '30px';

**Q7: What are javascript DOM events?**

**Ans:** Here are a example of events handing in DOM.

<html>

<head>

    <title>Button Click Example</title>

</head>

<body>

<button id="myButton">Click Me</button>

<script>

    var button = document.getElementById('myButton');

    button.addEventListener('click', function() {

        alert('Button clicked!');

    });

</script>

</body>

</html>

**Q8: What are nodes in DOM?**

**Ans:** A node is a basic unit of the document structure in the Document Object Model (DOM) and it represents individual things in the document, including elements, attributes, and text. With the exception of the root node, which has no parents, nodes create a hierarchical tree-like structure. Each node can have zero or more child nodes and one parent node.