The DOM API

HyperionDev

Muhammad Zahir Junejo

Lecture – Housekeeping

- ❑ The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.
    - ❑ Please review Code of Conduct (in Student Undertaking Agreement) if unsure
- ❑ No question is daft or silly - **ask them!**
- ❑ Q&A session at the end of the lesson, should you wish to ask any follow-up questions.
- ❑ Should you have any questions after the lecture, please schedule a mentor session.
- ❑ For all non-academic questions, please submit a query: www.hyperiondev.com/support

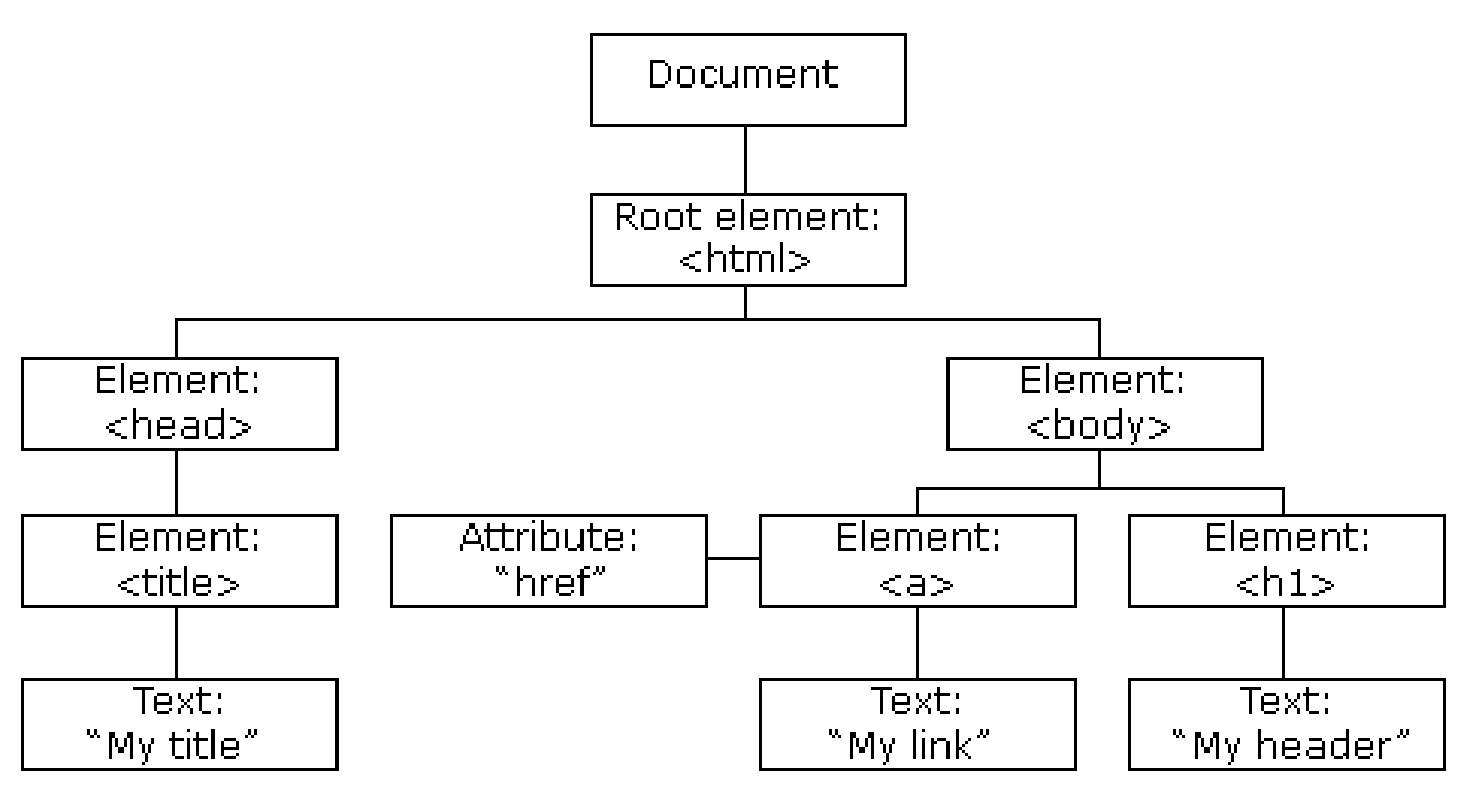
Lecture Objectives

1. What is the DOM?
2. The Structure of the DOM
3. Accessing Elements
4. Modifying Elements
5. Traversing the DOM

What is the DOM?

- ❑ Definition: The Document Object Model (DOM) is a programming interface for web documents. It represents the page so programs can change the document structure, style, and content dynamically.
- ❑ Importance: The DOM is a fundamental concept in web development, enabling interaction with web pages.
- ❑ With the HTML DOM, JavaScript can access and change all the elements of an HTML document. JavaScript gets all the power it needs to create dynamic HTML.
- ❑ When a web page is loaded, the browser creates a Document Object Model of the page.
- ❑ The DOM tree represents the structure of an HTML document. Each element in the document is a node in the tree, forming a parent-child relationship. Understanding this structure is essential for manipulating web page content using the DOM API.
- ❑ The document object represents your web page.

The Structure of the DOM



Accessing Elements

```
// Access elements by ID
const container = document.getElementById("container");

// Access elements by class name
const paragraphs = document.getElementsByClassName("paragraph");

// Access elements by tag name
const headings = document.getElementsByTagName("h2");

// Access elements using querySelector
const firstParagraph = document.querySelector("p");

// Access multiple elements using querySelectorAll
const allParagraphs = document.querySelectorAll("p");
```

Modifying Elements

```
// Create a new element
const newParagraph = document.createElement("p");
newParagraph.textContent = "This is a new paragraph.";

// Append the new element to the DOM
containerDiv.appendChild(newParagraph);

// Read and update content
const existingHeading = document.querySelector("h1");
console.log(existingHeading.textContent); // Read
existingHeading.textContent = "Updated Heading"; // Update

// Remove an element
const paragraphToRemove = document.querySelector("p");
container.removeChild(paragraphToRemove);
```

Traversing the DOM

```
// Traversing up the DOM
const parentDiv = containerDiv.parentNode;

// Traversing down the DOM
const childNodes = containerDiv.childNodes;
const firstChild = containerDiv.firstChild;
const lastChild = containerDiv.lastChild;

// Traversing siblings
const nextElement = containerDiv.nextElementSibling;
const previousElement = containerDiv.previousElementSibling;
```

References

- https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction
- https://www.w3.org/TR/WD-DOM/introduction.html
- https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model

Questions and Answers

Thank You!