Understanding the HTML DOM

A Beginner to Intermediate Guide

Chirag Dhamija

Intro to Software Systems
Internationl Institute of Information Technology, Hyderabad

January 26, 2025

What is HTML?

- **HTML** stands for **H**yper**T**ext **M**arkup **L**anguage.
- It is the standard markup language for creating web pages and web applications.
- HTML describes the structure of a webpage semantically and originally included cues for the appearance of the document.
- It is a cornerstone technology of the World Wide Web, alongside CSS and JavaScript.

Basic Structure of an HTML File

- An HTML document has a nested structure defined by tags.
- The basic skeleton includes:
  - `<!DOCTYPE html>` declaration.
  - `<html>` element that wraps the entire content.
  - `<head>` section for metadata and links to scripts/styles.
  - `<body>` section for the visible content.

Common HTML Tags

- **Headings**: `<h1>` to `<h6>` define headings of different levels.
- **Paragraph**: `<p>` defines a paragraph.
- **Div**: `<div>` is a block-level container for grouping content.
- **Span**: `<span>` is an inline container for grouping content.
- **Anchor**: `<a>` defines a hyperlink.
- **Image**: `<img>` embeds an image.
- **Unordered List**: `<ul>` creates a bulleted list.
- **List Item**: `<li>` defines an item in a list.
- **Script**: `<script>` embeds JavaScript code.
- **Link**: `<link>` links external resources like CSS.

HTML Tags Examples

Anchor Tag Example

```
<a href="https://www.example.com">Visit Example.com</a>
```

- Creates a clickable link that navigates to the specified URL.

Image Tag Example

```
<img src="image.jpg" alt="Description␣of␣image">
```

- Embeds an image into the webpage.
- `src` specifies the image source.
- `alt` provides alternative text for accessibility.

What is the DOM?

- The Document Object Model (DOM) is a programming interface for web documents.
- It represents HTML or XML documents as a tree of objects.
- It defines:
  - HTML elements as objects.
  - Properties, methods, and events for all HTML elements.

Key Features

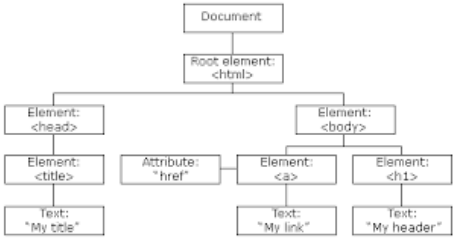
- Platform and language-independent.
- Enables dynamic manipulation of content, structure, and style.

The HTML DOM and JavaScript

- The HTML DOM is an API (Application Programming Interface) for JavaScript.
- With JavaScript, you can:
  - Add, change, or remove HTML elements and attributes.
  - Add, change, or remove CSS styles dynamically.
  - React to HTML events.
- When a web page is loaded, the browser creates a tree-like structure (DOM tree) to represent it.

DOM Tree Structure

- The DOM represents the document as a hierarchical tree structure.
- Nodes in the tree represent elements, attributes, and text.
- Node relationships include:
  - Parent, child, and sibling nodes.



Accessing the DOM

- To work with HTML elements, you need to find them first.
- Common methods for accessing elements:
  - `document.getElementById()`
  - `document.getElementsByTagName()`
  - `document.getElementsByClassName()`
  - `document.querySelector()`
  - `document.querySelectorAll()`

Code Example

```
var myElement = document.getElementById("intro");
console.log(myElement.textContent);
```

Manipulating the DOM

- You can dynamically modify content, attributes, and styles.
- Example:

Code Example

```
const newElement = document.createElement('div');
newElement.textContent = 'Hello, DOM!';
document.body.appendChild(newElement);
```

Events in the DOM

- Events are actions (e.g., clicks, hovers) occurring in the browser.
- Event listeners allow you to respond to these actions.

Code Example

```
const button = document.getElementById('myButton');
button.addEventListener('click', () => {
    alert('Button clicked!');
});
```

Finding HTML Elements

- Methods to locate elements in the DOM:
  - By ID: `document.getElementById("id")`
  - By tag name: `document.getElementsByTagName("tag")`
  - By class name: `document.getElementsByClassName("class")`
  - By CSS selector: `document.querySelector("selector")`
- Examples:
  - `var x = document.getElementsByTagName("p")`
  - `var x = document.querySelectorAll(".intro")`

Conclusion

- The DOM is crucial for building dynamic, interactive web pages.
- Mastering DOM manipulation empowers you to create engaging user experiences.
- Practice regularly to enhance your skills.

Thank You!