***DOM***

A web page is a document which can be displayed in the web browsers or as the HTML source but ultimately both are the same documents. **Document Object Model** (DOM) represents the same document to be manipulated.

* An object-oriented representation of any web page is called DOM. This web page can be modified with JavaScript.
* Document Object Model (DOM) is also an interface for HTML and XML documents for programming.
* DOM represents the page in a way where programs can change the document structure, style, and content.
* DOM also represents the document as nodes and objects which allow programming languages to connect to the pages.

**DOM tree**

The backbone of an HTML document is tags.

According to the Document Object Model (DOM), every HTML tag is an object. Nested tags are “children” of the enclosing one. The text inside a tag is an object as well.

All these objects are accessible using JavaScript, and we can use them to modify the page.

The DOM represents HTML as a tree structure of tags. Here’s how it looks:

<!DOCTYPE HTML>

<html>

    <head>

        <title>About DOM</title>

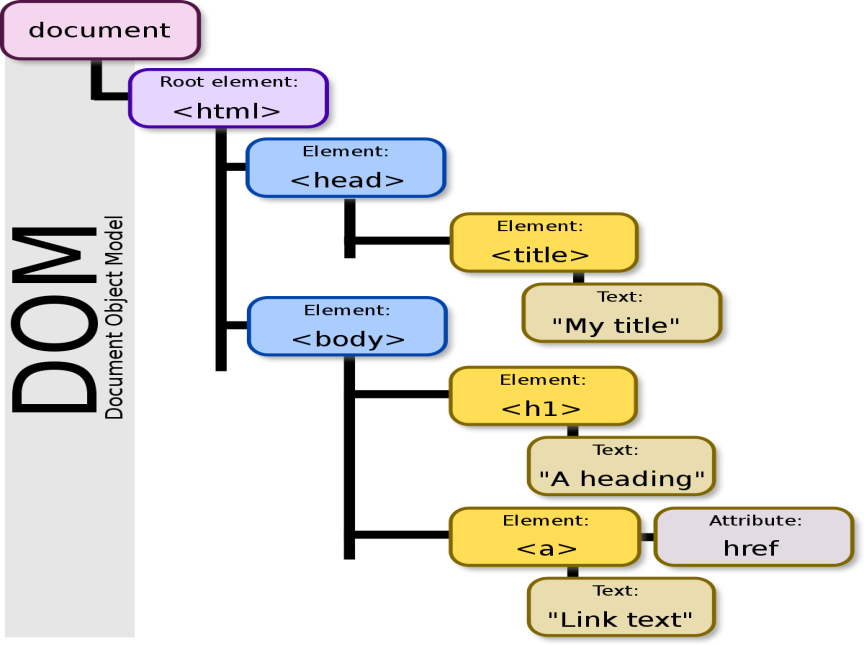
    </head>

    <body>

        An object-oriented representation of any web page is called DOM.

    </body>

</html>



Every tree node is an object.

Tags are element nodes (or just elements) and form the tree structure: <html> is at the root, then <head> and <body> are its children, etc.

The text inside elements forms text nodes, labelled as #text. A text node contains only a string. It may not have children and is always a leaf of the tree.

Every element in a document (the head, tables within the document, table headers, text within the table cells) is part of DOM. They can all be accessed and manipulated using a scripting language like JavaScript for the purpose of DOM.

JavaScript and the DOM were tightly intertwined at the starting, but they evolved into separate entities afterward. The page content is stored in the DOM which may be accessed and manipulated via JavaScript, which may lead us to the following equation:

**API (HTML or XML page) = DOM + JS (scripting language)**

DOM was designed to be totally independent of any particular programming language. Though our focus is exclusively on JavaScript, implementations of the DOM can be built for any language.

The common APIs in web and XML page scripting using the DOM are listed below:

1. **document.getElementById(id)**
2. **document.getElementsByTagName(name)**
3. **document.createElement(name)**
4. **parentNode.appendChild(node)**
5. **element.innerHTML**
6. **element.setAttribute()**
7. **element.getAttribute()**
8. **element.addEventListener()**
9. **window.content**
10. **window.onload**
11. **console.log()**
12. **window.scrollTo()**

## [Summary](https://javascript.info/dom-nodes" \l "summary)

* An HTML/XML document is represented inside the browser as the DOM tree.
* Tags become element nodes and form the structure.
* Text becomes text nodes.
* Everything in HTML has its place in DOM, even comments.
* We can use developer tools to inspect DOM and modify it manually.