

QUES - in properties of DOM which object is at top of hierarchy

ANS - In the **DOM (Document Object Model)** hierarchy, the **window object** is at the top of the hierarchy.

Hierarchical Structure:

1. **window Object:**

- Represents the browser window or a frame in which the document is displayed.
- It is the global object in the browser environment, meaning all global variables and functions are properties of this object.

```
console.log(window.alert === alert); // true
```

2. **document Object:**

- A property of the **window** object.
- Represents the web page (HTML document) loaded into the browser.
- Allows access to and manipulation of the content and structure of the document.

```
console.log(window.document === document); // true
```

3. **HTML Elements:**

- Nested under the **document** object.
- Represent individual elements in the HTML structure.
- Examples:
 - `document.body` for the `<body>` element.
 - `document.getElementById('id')` to access an element by its ID.

Simplified DOM Hierarchy:

php

window

└ document

└ <html>

└┬ <head>

└┬ <body>

└┬ <div>

└┬ <p>

The **window** object serves as the global scope, while the **document object** and other DOM nodes are accessible through it.

QUES - when an html document is loaded into a window it become a

Ans - When an **HTML document** is loaded into a browser window, it becomes a **Document Object Model (DOM)** document.

The DOM represents the structure of the HTML document as a tree of objects, which allows JavaScript and other scripts to dynamically access, manipulate, and interact with the content, structure, and style of the page.

HTML to DOM Transformation:

- The browser parses the HTML document.
- Converts it into a structured tree-like representation called the DOM.

DOM as a Programming Interface:

- The DOM provides a programming interface to interact with the HTML and CSS.
- Through the DOM, scripts can:
 - Modify the document structure (e.g., add or remove elements).
 - Update the content or style of elements.
 - React to user interactions (e.g., button clicks, form submissions).

QUES - what is difference b/w html and html5

Feature	HTML	HTML5
Version	Basic version of HTML without modern features.	Latest version with extended features.
Doctype Declaration	Complex and lengthy.	Simplified and easier to write.
	<pre><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"></pre>	<pre><!DOCTYPE html></pre>
Multimedia Support	Requires third-party plugins (e.g., Flash) for audio and video.	Native support for <code><audio></code> and <code><video></code> tags.

Semantics	Limited set of semantic elements like <code><div></code> .	Includes semantic tags like <code><article></code> , <code><section></code> , <code><header></code> , and <code><footer></code> .
Forms	Basic form elements with limited validation.	New input types (<code>email</code> , <code>date</code> , etc.), attributes (<code>required</code> , <code>placeholder</code>), and validation.
Graphics and Media	Graphics handled via plugins (e.g., Flash).	Built-in <code><canvas></code> and support for SVG (Scalable Vector Graphics).
APIs	Limited or no API support.	New APIs such as Geolocation, Web Storage, WebSockets, and Drag-and-Drop.
Mobile Compatibility	No specific features for mobile.	Improved mobile support with responsive design capabilities.

Advantages of HTML5 Over HTML:

1. **Simplified Syntax:** Easier to write and read.
2. **Enhanced Features:** Includes modern capabilities like video/audio, graphics, and APIs.
3. **Improved Semantics:** Semantic tags make the structure more meaningful and accessible.
4. **Faster Development:** Features like form validation and offline capabilities reduce dependency on external scripts.
5. **Better for Modern Web Apps:** Optimized for responsive and interactive web applications.

QUES - WHAT are <!DOCTYPE> and different types of it

The **<!DOCTYPE> declaration**, short for "Document Type Declaration," is used in HTML documents to specify the version of HTML that the browser should use to render the page. It must be the very first line in an HTML document, and it is not an HTML tag—it's an instruction for the web browser.

Why <!DOCTYPE> is Important:

- Without it, browsers may render the page in **quirks mode**, leading to inconsistent behavior across browsers.
- Ensures that modern web standards are applied.

Purpose of <!DOCTYPE>:

1. **Mode Selection:** Helps the browser decide whether to render the page in:
 - **Standards Mode:** Adheres to modern web standards.
 - **Quirks Mode:** Emulates older, less strict behavior for backward compatibility.

2. **Validation:** Indicates the rules and syntax to validate the HTML document.