

# DOM

JAVASCRIPT

JS

# What Is The DOM?

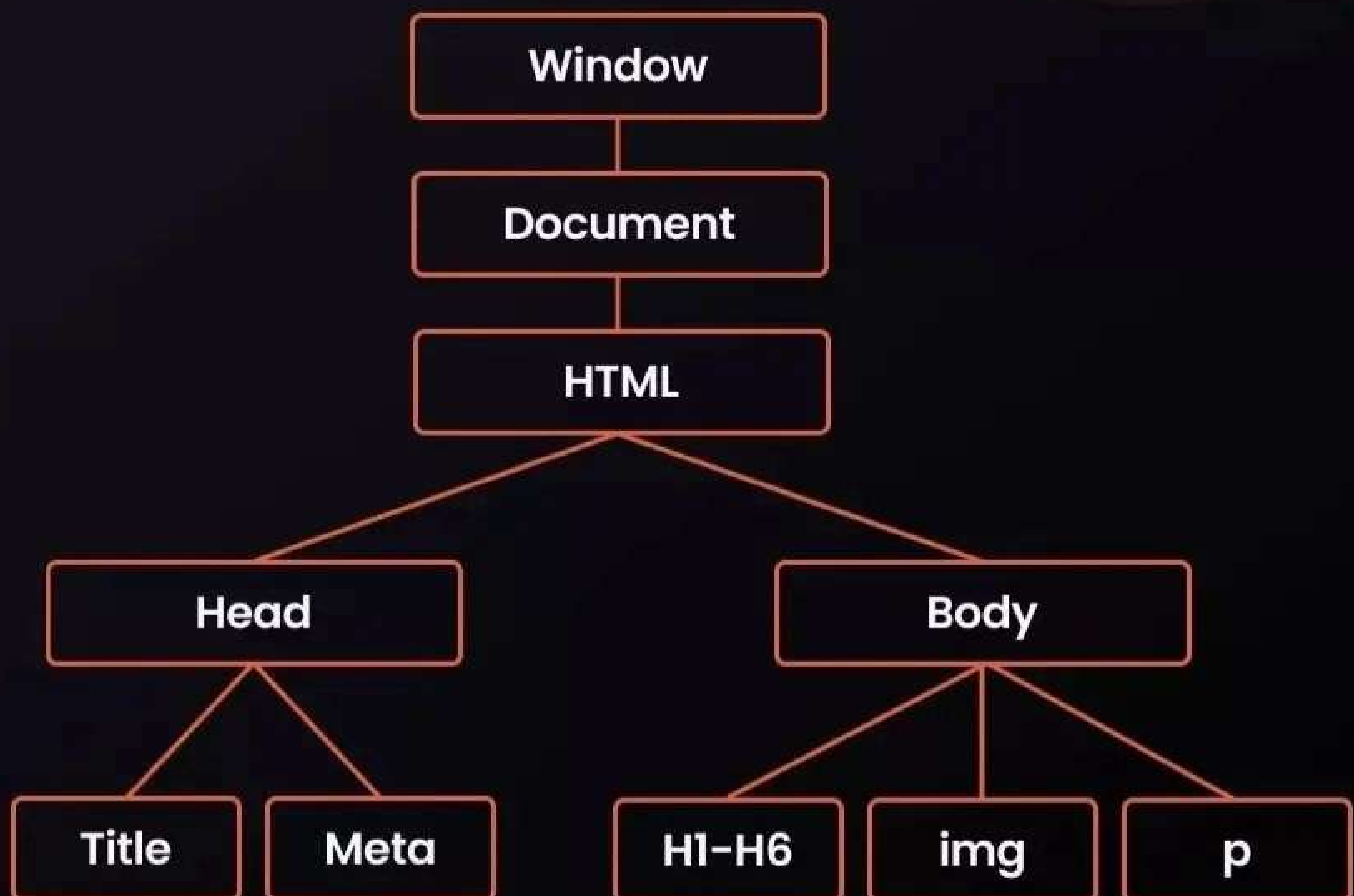
Imagine a **web page** as a house with rooms and furniture. The **DOM**, is like a **blueprint** or **map** of that house.

It helps JavaScript understand and **interact with the elements** of a web page, such as text, images, buttons, and forms.

In the DOM, **everything** on a web page **becomes an object**. Each element, like a **heading** or **paragraph**, and even **the attributes** (like "class") becomes an object.

# How Does The DOM Work?

The browser creates a **tree-like structure** in its memory, where each HTML element **becomes a node**.



# Accessing Elements

JavaScript can talk to the DOM and ask for specific elements.

For example, it can say, "Hey DOM, can you give me the heading with the ID 'mainHeading'?"

So, here is the most common method for accessing DOM elements.

- getElementById
- getElementsByClassName
- getElementsByTagName
- querySelector
- querySelectorAll

# GetElementById

This method allows you to select a single element by its unique id attribute. Since id values must be unique within a document.

This method always return one element (or null if no element matches the provided id).

```
// HTML
<div id="myDiv">This is a div element.</div>

// JS
const myDiv = document.getElementById('myDiv');
console.log(myDiv.textContent);
// Output: "This is a div element."
```

# GetElementsByClassName

This method returns a collection of elements that have the specified class name.

If there are no elements with the given class, an empty collection will be returned.

```
//HTML
<p class="highlight">This is a paragraph 1</p>
<p class="highlight">Another paragraph 2</p>

//JS
const elemClass = document.getElementsByClassName('highlight');
for (const element of elemClass) {
  console.log(element.textContent);
}
// Output:
// "This is a paragraph 1"
// "Another paragraph 2"
```

# GetElementsByTagName

This method returns a collection of elements with the specified tag name.

It's useful when you want to select all elements of a particular type, like `<p>`, `<div>`, `<h1>`, etc.

```
// HTML
<h1>Heading 1</h1>
<p>Paragraph 1</p>
<p>Paragraph 2</p>

// JS
const paragraphs = document.getElementsByTagName('p');
for (const paragraph of paragraphs) {
  console.log(paragraph.textContent);
}
// Output:
// "Paragraph 1"
// "Paragraph 2"
```

# querySelector and querySelectorAll

`querySelector` is a method that selects the first element matching a CSS-like selector and returns it (or null if no match is found).

While `querySelectorAll` returns a `NodeList` with all elements that match the given selector, allowing for iteration using methods like `forEach`.

```
// HTML
<p class="highlight">Hello, World!</p>
<ul><li class="highlight">Item 2</li>
  <li class="highlight">Item 4</li></ul>

const highlightedElement = document.querySelector('.highlight');
console.log(highlightedElement.textContent);
// Output: "Hello, World!"

const highlightedItems = document.querySelectorAll('.highlight');
highlightedItems.forEach((item) => {
  console.log(item.textContent);
});
// "Item 2"
// "Item 4"
```



## Conclusion

DOM is a **fantastic tool** that allows JavaScript to understand, control, and modify web pages.

It enables developers to **create dynamic and interactive websites** that respond to user actions and provide a delightful experience.

As always, I **hope you enjoyed** the post and **learned something new**.

If you have **any queries** then **let me know** in the comment box.

@CODE.CLASH



Check Out **My Profile** For  
Such **Excellent Posts** On  
**Web Development.**



IMTIYAZ NANDASANIYA