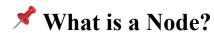
#### **JavaScript**

#### Day 5

#### **JavaScript HTML DOM Elements (Nodes)**



In the DOM, each part of an HTML document is a **node**:

- The document itself → **Document Node**
- HTML elements  $\rightarrow$  Element Nodes
- Text inside elements  $\rightarrow$  Text Nodes
- Comments → Comment Nodes

#### **Creating New Elements**

- **☑** Steps to Add a New Element
  - 1. Create a new element:

const para = document.createElement("p");

2. Create a text node:

const node = document.createTextNode("This is new.");

3. Append text to the element:

para.appendChild(node);

4. Append the new element to an existing element:

const element = document.getElementById("div1");
element.appendChild(para);

## insertBefore() — Insert at Specific Position

Instead of adding the new element at the end, you can insert it **before** another child:

element.insertBefore(para, child);

- **X** Removing Elements
- **✓** Using remove() (Newer Browsers)

document.getElementById("p1").remove();

- Using removeChild() (For Older Browsers)
  - 1. Get the **parent** element.
  - 2. Get the **child** to remove.
  - 3. Use:

parent.removeChild(child);

#### **Example:**

```
const parent = document.getElementById("div1");
const child = document.getElementById("p1");
parent.removeChild(child);
```

✓ Shortcut using parentNode:

const child = document.getElementById("p1"); child.parentNode.removeChild(child);

- Replacing Elements
- **☑** Use replaceChild():

```
const para = document.createElement("p");
const node = document.createTextNode("This is new.");
para.appendChild(node);
```

```
const parent = document.getElementById("div1");
const child = document.getElementById("p1");
parent.replaceChild(para, child);
```

# JavaScript HTML DOM Collections

- What is an HTMLCollection?
  - An HTMLCollection is an array-like object that contains a list of HTML elements.
  - It is **not a true array**, but you can access items using index numbers like an array.

### **★** How to Get an HTMLCollection

Use the getElementsByTagName() method to get a collection of elements with the same tag name:

const myCollection = document.getElementsByTagName("p"); This collects all elements from the document.

## **Accessing Elements by Index**

HTMLCollections use zero-based indexing:

myCollection[0]; // First element myCollection[1]; // Second element



#### NTMLCollection Length

You can check how many elements are in the collection using .length: myCollection.length;

Useful when looping through all elements.

## Loop Through HTMLCollection

You can use a for loop to change or access each element:

Example: Change all text color to red const myCollection = document.getElementsByTagName("p");

```
for (let i = 0; i < myCollection.length; <math>i++) {
 myCollection[i].style.color = "red";
```

### Important Notes

- HTMLCollection updates automatically if the document changes (live collection).
- HTMLCollection can be returned by:
  - getElementsByTagName()
  - getElementsByClassName()
  - children property of an element

#### 🔰 JavaScript HTML DOM Node Lists

- What is a NodeList?
  - A NodeList is a collection of nodes (elements, text, or comments) from the document.
  - It is **similar to an HTMLCollection**, but not the same.

#### **✗** When Do You Get a NodeList?

You get a **NodeList** from:

- document.querySelectorAll() ✓
- childNodes property ✓
- (Sometimes) getElementsByClassName() in old browsers 1

#### **Example: Select all elements**

const myNodeList = document.querySelectorAll("p");

## **Accessing Items in a NodeList**

You can use index numbers just like arrays:

myNodeList[0]; // First node myNodeList[1]; // Second node

## NodeList Length

Use .length to find out how many nodes are in the NodeList: myNodeList.length;

## Loop Through NodeList

You can loop through all nodes using a for loop:

**Example:** Change text color of all tags to red const myNodeList = document.querySelectorAll("p");

```
for (let i = 0; i < myNodeList.length; i++) {
 myNodeList[i].style.color = "red";
```

#### Wey Differences: HTMLCollection vs NodeList

#### **HTMLCollection NodeList Feature**

Returns elements Yes only

Can include text/comments

X No

Live updates (auto-update) ✓ Yes (e.g. getElementsByTagName)

Loop with forEach()

X Not always

Yes (if from querySelectorAll)

Yes (e.g. from childNodes)

X No (static snapshot)

☑ Often (modern browsers)