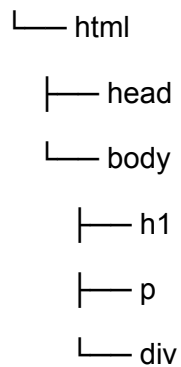


# DOM in JavaScript

## 1. What is DOM?

DOM (Document Object Model) is a programming interface that represents an HTML page as a tree structure so JavaScript can access, change, add, or remove content, styles, and elements.

- When a webpage loads, the browser converts HTML into a DOM tree.



## 2. Why DOM is Important?

Using DOM, JavaScript can:

- Read HTML content
- Change text and attributes
- Change CSS styles
- Create new elements

Delete elements

Handle user events (click, input, etc.)

## 3. Accessing DOM Elements

### 3.1 By ID

```
document.getElementById("title")
```

### 3.2 By Class Name

```
document.getElementsByClassName("box")
```

### 3.3 By Tag Name

```
document.getElementsByTagName("p")
```

### 3.4 Query Selector (Most Used)

```
document.querySelector("#title")    // first match
```

```
document.querySelectorAll(".box")   // all matches
```

## 4. Reading & Changing Content

### 4.1. innerText

```
element.innerText = "Hello"
```

### 4.2. textContent

```
element.textContent = "Hello"
```

### 4.3. innerHTML

```
element.innerHTML = "<b>Hello</b>"
```

Property	Description
innerText	Visible text only
textContent	All text
innerHTML	HTML + text

## 5. Changing Styles

```
element.style.color = "red"
```

```
element.style.backgroundColor = "blue"
```

```
element.style.fontSize = "20px"
```

## 6. Attributes Handling

### 6.1 Get Attribute

```
element.getAttribute("id")
```

### 6.2 Set Attribute

```
element.setAttribute("class", "box")
```

### 6.3 Remove Attribute

```
element.removeAttribute("disabled")
```

## 7. Creating & Adding Elements

### 7.1 Create Element

```
let p = document.createElement("p")
```

```
p.innerText = "New Paragraph"
```

## 7.2 Add to Page

```
document.body.appendChild(p)
```

## 7.3 Insert at Specific Position

```
parent.insertBefore(newElement, referenceElement)
```

# 8. Removing Elements

```
element.remove()
```

or

```
parent.removeChild(child)
```

# 9. DOM Events

## 9.1 Click Event

```
button.onclick = function() {  
    alert("Clicked")  
}
```

## 9.2 addEventListener (Best Method)

```
button.addEventListener("click", function() {  
    alert("Clicked")  
}))
```

Common Events

Event Use

click    mouse click

mouseover    hover

keydown    key pressed

submit form submit

input    typing

## 10. Event Object

```
button.addEventListener("click", function(event){  
    console.log(event.target)  
})
```

## 11. Form Handling

```
let name = document.getElementById("username").value
```

**Prevent reload:**

```
event.preventDefault()
```

## 12. Traversing DOM

element.parentElement

element.children

element.firstElementChild

element.lastElementChild

element.nextElementSibling

element.previousElementSibling

## 13. Class List Manipulation

```
element.classList.add("active")
```

```
element.classList.remove("active")
```

```
element.classList.toggle("active")
```

```
element.classList.contains("active")
```

## 14. Window Object

```
window.alert("Hi")
```

```
window.location.href
```

```
window.innerWidth
```

```
window.innerHeight
```

## 15. DOM Ready

```
document.addEventListener("DOMContentLoaded", function() {
```

```
    console.log("DOM Loaded")
```

```
})
```

```
.
```