JavaScript DOM Manipulation – Detailed Notes with Examples

---

1. DOM Selectors

DOM selectors allow you to pick HTML elements so you can use or modify them.

1.1 getElementById()

- Selects **one element** using its **id**

- Returns **a single object**

✔ Example

```
<p id="title">Hello Students</p>

<script>
let heading = document.getElementById("title");
console.log(heading);
</script>
```

1.2 getElementsByClassName()

- Selects elements using their **class**

- Returns an **HTMLCollection** (like an array)

✔ Example

```
<p class="msg">Hi</p>
<p class="msg">Hello</p>

<script>
let messages = document.getElementsByClassName("msg");
console.log(messages[0]); // First element
</script>
```

1.3 getElementsByTagName()

- Selects elements by tag name (p, div, button, etc.)

- Returns **HTMLCollection**

✔ Example

```
let allParagraphs = document.getElementsByTagName("p");
console.log(allParagraphs.length);
```

1.4 querySelector()

- Selects the **first** matching element

- Uses **CSS selectors**

✔ Example

```
let firstDiv = document.querySelector("div");        // tag
let special = document.querySelector("#main");       // id
let box = document.querySelector(".box");            // class
```

1.5 `querySelectorAll()`

- Selects **all** matching elements

- Returns a **NodeList**

✔ Example

```javascript
let items = document.querySelectorAll(".item");
console.log(items);
```

---

2. Get Input Element Data (value)

Use `.value` to get what the user typed in an input.

✔ Example

```html
<input type="text" id="username">
<button onclick="showName()">Show Name</button>

<script>
function showName() {
  let name = document.getElementById("username").value;
  console.log(name);
}
</script>
```

---

3. Useful JavaScript Methods

3.1 `toFixed()`

- Converts a number to a **fixed decimal format**

✔ **Example**

```
let price = 99.456;
console.log(price.toFixed(2)); // "99.46"
```

3.2 `Math.pow(a, b)`

- Returns **a raised to power b**

✔ **Example**

```
console.log(Math.pow(2, 3)); // 8
```

3.3 `trim()`

- Removes spaces from **start and end**

✔ **Example**

```
let name = "   John   ";
console.log(name.trim()); // "John"
```

3.4 `toLowerCase()`

```
console.log("HELLO".toLowerCase()); // "hello"
```

3.5 `toUpperCase()`

```
console.log("hello".toUpperCase()); // "HELLO"
```

---

4. innerText, textContent, innerHTML

4.1 `innerText`

- Returns **visible text**

- Ignores hidden text

- Affects layout (slow)

✔ **Example**

```
document.getElementById("demo").innerText = "Hello";
```

4.2 `textContent`

- Returns **all text**, including hidden text

- Faster than innerText

✔ **Example**

```
document.getElementById("demo").textContent = "Hello Students";
```

4.3 `innerHTML`

- Reads/sets **HTML code inside an element**

✔ **Example**

```
document.getElementById("demo").innerHTML = "<b>Bold Text</b>";
```

---

5. `onClick` Attribute in HTML

Used to call JavaScript when a button or element is clicked.

✔ **Example**

```
<button onclick="sayHello()">Click Me</button>

<script>
function sayHello() {
  alert("Hello!");
}
</script>
```

---

6. Changing CSS Using JavaScript

Use:

```
element.style.property = "value";
```

✔ **Example**

```
<p id="text">Hello World</p>

<script>
document.getElementById("text").style.color = "blue";
```

```
document.getElementById("text").style.fontSize = "24px";
</script>
```

7. setAttribute() and getAttribute()

7.1 setAttribute(attribute, value)

- Adds or changes an attribute

✔ **Example**
```
let img = document.getElementById("photo");
img.setAttribute("src", "image.jpg");
img.setAttribute("width", "200");
```

7.2 getAttribute(attribute)

- Gets the value of an attribute

✔ **Example**
```
let value = img.getAttribute("src");
console.log(value);
```