🌐DOM Basics – Cheat Sheet

1️⃣What is DOM?

The **DOM (Document Object Model)** is a programming interface for HTML and XML documents. It represents the page so that programs (like JavaScript) can manipulate its structure, style, and content.

💡Think of it like this:

- **HTML** → Skeleton of the page
- **CSS** → Skin / Style
- **JavaScript with DOM** → Hands that can move and change things dynamically

🔔Every HTML element becomes a **node** in the DOM tree, and JavaScript can interact with it.

2️⃣Selecting Elements

To manipulate elements, first we need to **select them** in JavaScript.

a) `getElementById`

Selects an element by its **id**. Returns a **single element**.

```html
<h1 id="title">Original Title</h1>

<script>
  const heading = document.getElementById("title");
  heading.innerText = "Hello Umesh!";   // Change text
  heading.style.color = "blue";         // Change color
  heading.style.fontSize = "30px";      // Change font size
</script>
```

🔧 `document.getElementById("title")` → selects `<h1>` with `id="title"`.

---

b) `getElementsByClassName`

Selects **all elements** with a given class. Returns an **HTMLCollection**.

```
<p class="info">Paragraph 1</p>
<p class="info">Paragraph 2</p>

<script>
  const paragraphs = document.getElementsByClassName("info");
  paragraphs[0].innerText = "Updated Paragraph 1";
  paragraphs[0].style.color = "green";
  paragraphs[1].innerText = "Updated Paragraph 2";
  paragraphs[1].style.color = "red";
</script>
```

🔧Use `paragraphs[0]`, `paragraphs[1]` like an array.

---

c) `querySelector`

Selects the **first element** that matches a CSS selector.

```
<div class="box">Box 1</div>
<div class="box">Box 2</div>

<script>
  const box = document.querySelector(".box");
  box.style.backgroundColor = "yellow";
  box.innerText = "Hello Box!";
</script>
```

---

d) `querySelectorAll`

Selects **all elements** that match a CSS selector.

```
<script>
  const boxes = document.querySelectorAll(".box");
  boxes.forEach((b, index) => {
    b.style.border = "2px solid black";
    b.innerText = `Box ${index + 1} updated`;
  });
</script>
```

3️⃣Changing Content

Two ways to change content:

a) `innerText`

Shows only text (ignores HTML tags).

```
document.getElementById("title").innerText = "Hello Umesh!";
```

b) `innerHTML`

Can insert HTML with tags.

```
document.getElementById("title").innerHTML = "<em>Hello Umesh!</em>";
```

🔧 `innerText` → plain text | `innerHTML` → renders tags

4️⃣Changing Styles

Use `.style` property in **camelCase**.

```
const heading = document.getElementById("title");
heading.style.color = "blue";
heading.style.fontSize = "40px";
heading.style.backgroundColor = "yellow";
```

🔧CSS `font-size` → JS `fontSize`

5️⃣Inspect DOM (DevTools)

- Right-click → **Inspect**
- Open **Elements** tab → view DOM tree
- Edit styles or text live
- Perfect for debugging 💻

6️⃣Mini Exercise

```
<h2 class="subtitle">Welcome!</h2>

<script>
  const sub = document.querySelector(".subtitle");
  sub.innerText = "Hello from JS!";
  sub.style.color = "purple";
  sub.style.fontSize = "25px";
</script>
```

✅Summary

- **DOM** = page structure accessible by JS

Select elements:

- `getElementById()` → single element
- `getElementsByClassName()` → multiple elements
- `querySelector()` → first match
- `querySelectorAll()` → all matches

Change content:

- `innerText` → plain text
- `innerHTML` → HTML allowed

Change styles:

- Use `.style` with camelCase

Inspect DOM:

- Use **DevTools** for live editing