

## Cos'è il DOM?

Il **DOM (Document Object Model)** è un'interfaccia di programmazione per i documenti HTML e XML. Esso rappresenta la struttura della pagina web come un albero di oggetti, dove ogni nodo nell'albero corrisponde a una parte della pagina: un elemento HTML, un attributo, un testo, ecc.

Il DOM fornisce una rappresentazione strutturata di una pagina web che può essere manipolata tramite JavaScript. Grazie al DOM, puoi leggere, modificare, aggiungere o rimuovere elementi HTML direttamente dalla pagina in modo dinamico, creando esperienze interattive.

### Esempio 1: Modificare il contenuto di un elemento HTML

Immagina di voler cambiare il contenuto di un paragrafo all'interno di una pagina web. Il DOM ti permette di farlo facilmente.

#### HTML:

```
<!DOCTYPE html>
<html lang="it">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Manipolare il DOM</title>
</head>
<body>

  <h1>Benvenuto sulla mia pagina web</h1>
  <p id="paragrafo">Questo è un paragrafo che cambierà.</p>
  <button onclick="cambiaTesto()">Cambia il testo</button>

  <script>
    function cambiaTesto() {
      // Selezioniamo l'elemento con id 'paragrafo'
      var paragrafo = document.getElementById("paragrafo");

      // Modifichiamo il suo contenuto
      paragrafo.textContent = "Il testo del paragrafo è stato cambiato!";
    }
  </script>

</body>
</html>
```

#### Spiegazione:

- In questo esempio, abbiamo un paragrafo con `id="paragrafo"`.
- Quando l'utente clicca sul pulsante, viene chiamata la funzione `cambiaTesto()`, che usa `document.getElementById()` per selezionare l'elemento `<p>` con l'ID "paragrafo".
- La proprietà `textContent` viene quindi utilizzata per cambiare il testo all'interno di quel paragrafo.

### Esempio 2: Modificare gli attributi di un elemento HTML

In questo esempio, cambieremo l'attributo `src` di un'immagine tramite il DOM.

## HTML:

```
<!DOCTYPE html>
<html lang="it">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Modificare gli attributi</title>
</head>
<body>

  <h1>Modifica dell'immagine</h1>
  
  <button onclick="cambiaImmagine()">Cambia immagine</button>

  <script>
    function cambiaImmagine() {
      // Selezioniamo l'elemento immagine
      var immagine = document.getElementById("immagine");

      // Cambiamo l'attributo 'src' dell'immagine
      immagine.src = "https://via.placeholder.com/300";
      immagine.alt = "Immagine modificata";
    }
  </script>

</body>
</html>
```

## Spiegazione:

- Abbiamo un'immagine iniziale con un URL di esempio.
- Quando l'utente clicca sul pulsante, la funzione `cambiaImmagine()` seleziona l'elemento immagine usando `document.getElementById()`.
- Successivamente, modifichiamo l'attributo `src` per cambiare l'immagine visualizzata. Modifichiamo anche l'attributo `alt` per descrivere la nuova immagine.

## Esempio 3: Aggiungere nuovi elementi al DOM

In questo esempio, vedremo come aggiungere nuovi elementi HTML (un nuovo paragrafo) alla pagina dinamicamente.

## HTML:

```
<!DOCTYPE html>
<html lang="it">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Aggiungere elementi al DOM</title>
</head>
<body>

  <h1>Aggiungere un paragrafo</h1>
```

```

<div id="contenitore">
  <p>Questo è il paragrafo esistente.</p>
</div>
<button onclick="aggiungiParagrafo()">Aggiungi un nuovo paragrafo</button>

<script>
  function aggiungiParagrafo() {
    // Creiamo un nuovo elemento paragrafo
    var nuovoParagrafo = document.createElement("p");

    // Impostiamo il testo del nuovo paragrafo
    nuovoParagrafo.textContent = "Questo è un paragrafo aggiunto dinamicamente.";

    // Selezioniamo il contenitore dove vogliamo aggiungere il nuovo paragrafo
    var contenitore = document.getElementById("contenitore");

    // Aggiungiamo il nuovo paragrafo al contenitore
    contenitore.appendChild(nuovoParagrafo);
  }
</script>

</body>
</html>

```

### Spiegazione:

- In questo caso, creiamo un nuovo paragrafo quando l'utente clicca sul pulsante.
- Utilizziamo `document.createElement("p")` per creare un nuovo elemento `<p>`.
- Impostiamo il testo del paragrafo usando la proprietà `textContent`.
- Poi selezioniamo un contenitore esistente (div con id `contenitore`) e aggiungiamo il nuovo paragrafo con `appendChild()`.

### Esempio 4: Rimuovere un elemento dal DOM

Infine, vediamo come rimuovere un elemento esistente dalla pagina.

#### HTML:

```

<!DOCTYPE html>
<html lang="it">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Rimuovere un elemento</title>
</head>
<body>

  <h1>Rimuovere un elemento dal DOM</h1>
  <div id="contenitore">
    <p id="paragrafoDaRimuovere">Questo paragrafo verrà rimosso.</p>
  </div>
  <button onclick="rimuoviParagrafo()">Rimuovi il paragrafo</button>

  <script>
    function rimuoviParagrafo() {

```

```
// Selezioniamo l'elemento paragrafo che vogliamo rimuovere
var paragrafo = document.getElementById("paragrafoDaRimuovere");

// Rimuoviamo il paragrafo dal DOM
paragrafo.remove();
}
</script>

</body>
</html>
```

**Spiegazione:**

- Abbiamo un paragrafo con l'id `paragrafoDaRimuovere`.
- Quando l'utente clicca sul pulsante, la funzione `rimuoviParagrafo()` seleziona il paragrafo e lo rimuove dal DOM usando il metodo `.remove()`.