

## MATERI

## Hari 5 – DOM Javascript

DOM atau Document Object Model adalah suatu antarmuka untuk merepresentasikan dokumen web sebagai objek. Dengan DOM, seluruh komponen dari halaman web dapat diakses dan dimanipulasi sesuai keinginan kita. Komponen-komponen pada halaman website yang dapat dimanipulasi mencakup elemen HTML, atribut pada elemen, text, dll.

DOM dapat dimanipulasi seperti dibuat baru, diubah, dan dihapus dengan menggunakan javascript.

### Selector DOM

Membuat selektor artinya kita ingin mengakses sebuah element atau komponen pada halaman web untuk kemudian kita dapat lakukan manipulasi di dalamnya. biasanya kita menampung hasil selektor pada sebuah variabel agar dapat mudah dipanggil kembali.

`document.getElementById()`

untuk mengakses sebuah elemen berdasarkan id dari elemen kita dapat gunakan selector `getElementById()`. Hasil selektor ini adalah sebuah elemen yang dapat kita lakukan manipulasi.

cara membuat selektor dengan id yaitu dengan sintaks sbb:

`document.getElementById("<nama id element>")`

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1 id="judul">Berlatih DOM</h1>

  <script>
    var judul = document.getElementById("judul")
  </script>
</body>
</html>
```

`document.getElementsByClassName()`

Untuk mengakses beberapa element sekaligus, kita dapat gunakan selektor menggunakan class. hasil selektor ini yaitu berupa HTML Collection atau kumpulan element HTML yang bisa kita akses sebagai array.

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>

  <ul>
    <li class="list-menu">goreng tempe</li>
    <li class="list-menu">mendoan</li>
    <li class="list-menu">tahu bulat</li>
  </ul>
  <script>
    var listMenu = document.getElementsByClassName("list-menu")
    // Hasil selektor yaitu berupa array collection HTML
    console.log(listMenu.length) // 3
    var gorengTempe = listMenu[0]
  </script>
</body>
</html>
```

## Manipulasi DOM

Melakukan manipulasi DOM adalah salah satu cara agar tampilan web menjadi interaktif. beberapa contoh manipulasi yaitu dengan cara membuat element baru, mengubahnya atau menghapusnya. Terdapat beberapa metode untuk memanipulasi DOM di antaranya :

### .innerHTML

metode innerHTML yaitu metode untuk mengisi atau mengganti isi dari sebuah element HTML dengan apapun yang kita inginkan. Kita dapat isi dengan sebuah teks node atau bahkan kita isi dengan element HTML di dalam DOM yang sedang kita akses.

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

  <div id="menu-makanan">

  </div>

  <script>
    var menuMakanan = document.getElementById("menu-makanan")

    // menambahkan sebuah teks
    menuMakanan.innerHTML = "Ini adalah menu makanannya"

    // menambahkan element HTML
    menuMakanan.innerHTML = "<h2>Ini adalah menu makanannya<h2>"

    // menambahkan banyak element HTML dengan banyak baris dapat gunakan tanda petik backtic ``
    menuMakanan.innerHTML = ` <h2>Ini adalah menu makanannya<h2>
      <ul>
        <li class="list-menu">goreng tempe</li>
        <li class="list-menu">mendoan</li>
        <li class="list-menu">tahu bulat</li>
      </ul>`
  </script>
</body>
</html>
```

### Metode lainnya

metode lainnya dapat dipelajari dari playlist channel Web Pemrograman Unpas berikut:

[Link video](#)

## DOM Events

*Event* adalah sebuah aktivitas atau sebuah kejadian yang terjadi di dalam sebuah DOM. Kejadian tersebut dapat kita tangkap atau didengarkan (*listen*) untuk kemudian kita lakukan sesuatu hal yang lain. hal ini tentunya membuat halaman web kita menjadi sangat interaktif.

Cara menangkap sebuah event ada 2 cara yaitu:

### inline attribute dengan tag HTML

beberapa tag pada HTML memiliki atribut bawaan berupa event, contohnya `<input onchange="alert('hello')">` atau `<button onclick="alert('i am clicked')"> Klik button </button>`

### menambahkan `addEventListener()`

#### 1. menangkap event submit pada form

untuk menangkap kejadian ketika sebuah form disubmit kita dapat tambahkan `addEventListener` untuk event `submit`. contohnya di bawah ini sebuah form dengan id `formConditional` ditangkap event submit nya.

di dalam function `addEventListener` terdapat dua parameter yaitu nama eventnya dan function handler nya.

nama event yang ingin ditangkap adalah "submit". function handler adalah function yang dijalankan ketika event tersebut didengarkan. function handler akan punya satu parameter yaitu event. Agar form tersebut tidak merefresh halaman maka kita gunakan `event.preventDefault()`.

## Rating - Feedback

Berikan Rating pada posting ini:



```
<h1>Berlatih Kondisional</h1>
```

```
<h3>Masukkan nama dan peranmu!</h3>
```

```
<form action="#" id="formConditional">
```

```
<input type="text" id="name" placeholder="Isi nama di sini">
```

All Rights Reserved by SanberCode. Powered by P1 Sanbersy

Perlu bantuan? chat via telegram : [@sanbercode](#)