

# DOM HTML



L. Erawan



# DOM HTML

**DOM** atau *Document Object Model* adalah antarmuka yang tidak berpihak kepada platform dan bahasa tertentu yang membuat program dan script dapat mengakses dokumen HTML secara dinamis dan mengupdate struktur, konten, dan style dokumen.

Ada 3 jenis DOM:

- Core DOM
- XML DOM
- **HTML DOM**

**DOM HTML** mendefinisikan:

- **Elemen-elemen HTML** sebagai obyek
- **Properti** seluruh elemen HTML
- **Metode** untuk mengakses seluruh elemen HTML
- **Event** untuk seluruh elemen HTML

DOM HTML menggunakan bahasa pemrograman untuk mengakses obyek-obyeknya, biasanya **Javascript**. Semua elemen HTML diperlakukan sebagai obyek.



# Metode2 DOM HTML

## **Mencari** elemen HTML:

- `document.getElementById()` = mencari menggunakan elemen id
- `document.getElementsByTagName()` = mencari menggunakan nama tag
- `document.getElementsByClassName()` = mencari menggunakan nama kelas

## **Mengubah** elemen HTML:

- `elemen.innerHTML` = mengubah inner HTML elemen HTML
- `elemen.atribut` = mengubah atribut elemen HTML
- `elemen.setAttribute(atribut, nilai)` = mengubah atribut elemen HTML
- `elemen.style.property` = mengubah style elemen HTML

## **Menambah** dan **menghapus** elemen HTML:

- `document.createElement()` = membuat elemen HTML
- `document.removeChild()` = menghapus elemen HTML
- `document.appendChild()` = menambah sebuah elemen HTML
- `document.replaceChild()` = mengganti elemen HTML
- `document.write(teks)` = menulis teks ke layar



# Contoh Penggunaan DOM HTML

```
<html>
<head><title>Latihan DOM</title></head>
<body>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML
= "Hello World!";
</script>
</body>
</html>
```



# Penjelasan Contoh

- Dalam contoh diatas, **getElementById** (perhatikan penulisannya) adalah sebuah **metode** dan **innerHTML** adalah **properti**.
- Cara yang biasa digunakan untuk mengakses suatu elemen HTML adalah menggunakan **atribut id** dari elemen. Contoh diatas metode `getElementById` menggunakan **id="demo"** untuk menemukan elemen paragraf.
- Sementara cara yang paling mudah untuk mendapatkan konten suatu elemen adalah **menggunakan properti innerHTML**. Properti ini dapat digunakan untuk memperoleh atau mengubah setiap elemen HTML termasuk `<html>` dan `<body>`.



# Event

- DOM HTML memungkinkan javascript bereaksi terhadap event HTML. Sebuah skrip javascript dapat dieksekusi ketika suatu event terjadi, misalnya ketika seorang pengguna melakukan klik terhadap suatu elemen HTML.
- Untuk menambahkan kode javascript ketika pengguna melakukan klik terhadap suatu elemen, dengan perintah:  
**onclick=javascript**
- Macam-macam event:
  - Event yang dapat terjadi:
  - Ketika pengguna melakukan klik mouse
  - Ketika sebuah halaman web telah dimuat (load)
  - Ketika sebuah gambar telah dimuat
  - Ketika mouse melintasi sebuah elemen
  - Ketika form HTML disubmit
  - Ketika pengguna menekan tombol keyboard



# Contoh Event

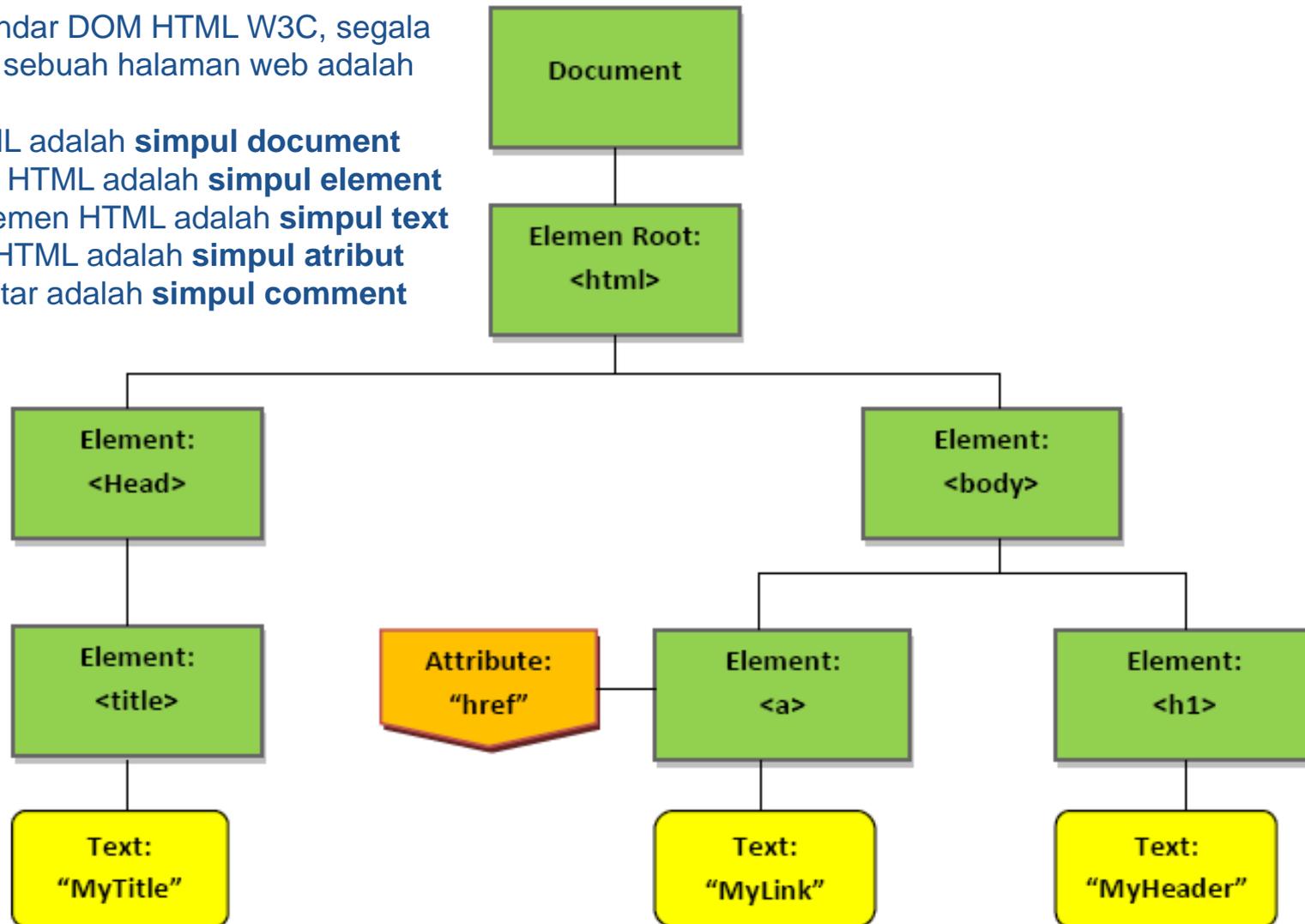
```
<!DOCTYPE html>
<html>
<body>
<h1 onclick="changeText(this)">Klik teks ini!</h1>
<script>
    function changeText(id) {
        id.innerHTML = "Ooops!";
    }
</script>
</body>
</html>
```



# Pohon DOM

Berdasarkan standar DOM HTML W3C, segala sesuatu didalam sebuah halaman web adalah simpel:

- dokumen HTML adalah **simpul document**
- Setiap elemen HTML adalah **simpul element**
- Teks dalam elemen HTML adalah **simpul text**
- Setiap atribut HTML adalah **simpul atribut**
- Semua komentar adalah **simpul comment**





# Menelusuri Simpul dalam Pohon DOM

Untuk menelusuri simpul-simpul dapat kita menggunakan properti-properti dari simpul:

- parentNode
- childNodes[]
- firstChild
- lastChild
- nextSibling
- previousSibling



# Contoh Penerapan Pohon DOM

```
<!DOCTYPE html>
<html>
<body>
<h1 id="intro">My First Page</h1>
<p id="demo">Hello World!</p>
<script>
var myText =
    document.getElementById("intro").childNodes[0].nodeValue;
    document.getElementById("demo").innerHTML = myText;
</script>
</body>
</html>
```