**Membuat Elemen**



Pada *course* JavaScript Dasar, kita telah mempelajari konsep dasar DOM (Document Object Model). Kita sudah bisa mengakses halaman HTML melalui JavaScript, menangani sebuah *event* yang diberikan dari HTML, hingga mengubah elemen HTML. Pada topik ini kita akan melanjutkan pembahasan bagaimana membuat sebuah elemen HTML melalui JavaScript, membuat *event* melalui JavaScript, hingga menghapus elemen HTML.

***DOM*** (*Document Object Model*) merupakan cara yang dilakukan oleh JavaScript untuk dapat memanipulasi elemen pada HTML. Mulai dari membuat, mengakses, mengubah, dan menghapus elemen HTML melalui JavaScript.

Untuk membuat sebuah elemen menggunakan DOM, kamu memerlukan *method* createElement(). Lalu memberikan sebuah argumen berupa nama elemen yang ingin dibuat. Tentunya elemen di sini adalah elemen yang tersedia pada HTML. *Method* ini akan mengembalikan nilai berupa objek:

document.createElement("NAMA\_ELEMEN");

Tapi, membuat elemen saja tidak cukup. Elemen yang sudah dibuat belum ada isinya. Maka dari itu kamu dapat menggunakan *built-in method* lainnya seperti createTextNode(), createAttribute(), dan setAttribute() untuk membuat elemen yang utuh.

Kita akan membuat elemen *heading* menggunakan DOM pada JavaScript. Berikut langkah-langkah yang akan kita lakukan:

**Langkah 1**: Kita akan membuat *file* bernama index.html yang berisi kode seperti berikut ini:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>DOM - Membuat Elemen</title>

</head>

<body>

<script src="elemen.js"></script>

</body>

</html>

**Langkah 2**: Kita akan membuat *file* bernama elemen.js. Kita tentukan dahulu *parent* dari elemen HTML yang akan kita buat. Pada kasus ini kita sudah mempunyai elemen <body> sebagai *parent*. Maka pada *file* elemen.js kita akan menambahkan kode berikut ini:

// Mengakses elemen <body>

let body = document.body;

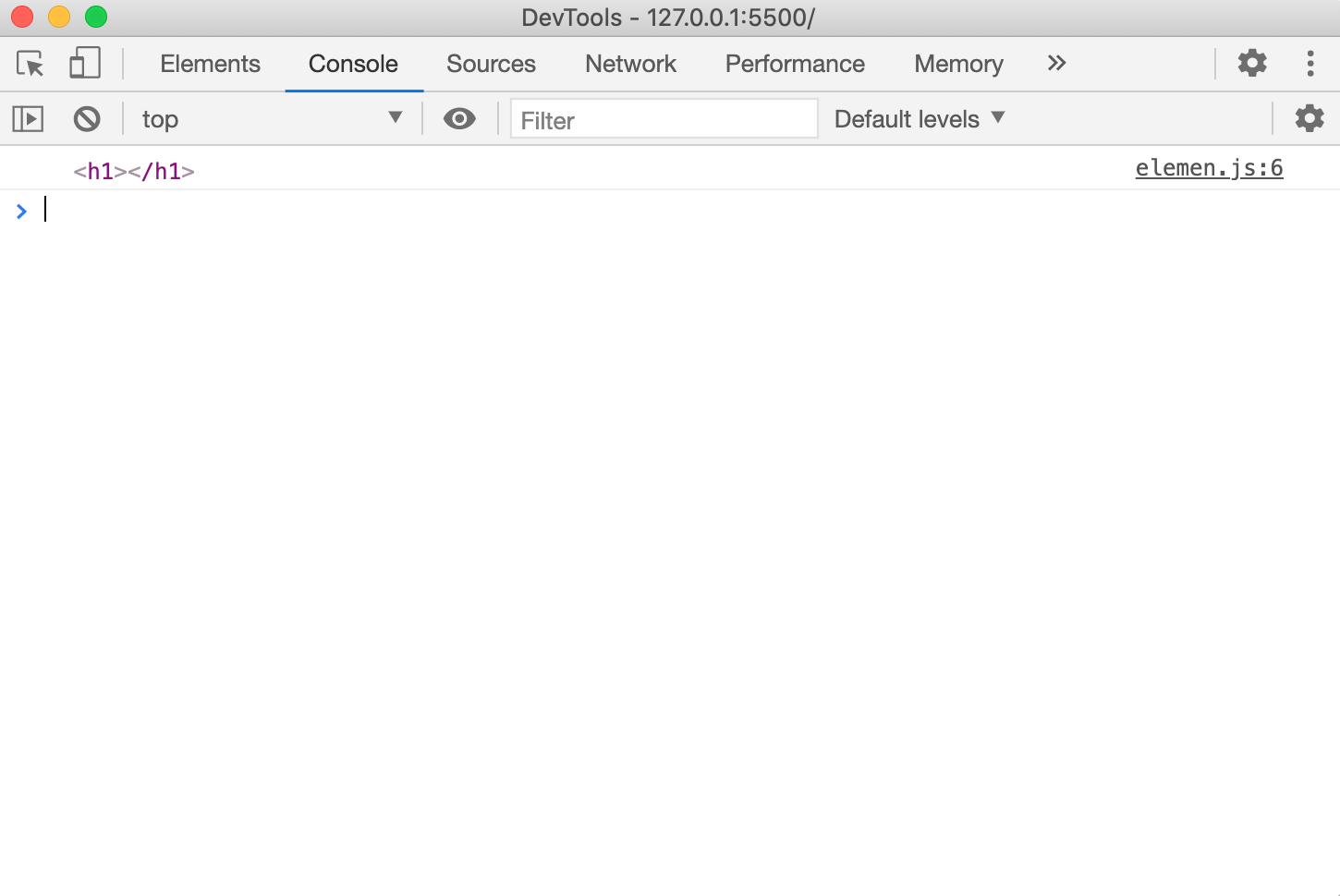
**Langkah 3**: Kita akan membuat elemen baru yaitu elemen *heading* <h1> menggunakan *method* createElement("NAMA\_ELEMEN"). Kita akan menambahkan kode berikut ini di bawah kode yang telah dibuat sebelumnya:

// Membuat elemen <h1>

let heading = document.createElement("h1");

console.log(heading);

Jika file index.html dijalankan, maka pada *console* akan terlihat tampilan seperti berikut ini:



**Langkah 4**: Kita akan memasukkan elemen <h1> di dalam elemen <body> menggunakan *method* appendChild("NODE") yang membutuhkan parameter berupa *node object* dari elemen atau konten teks yang telah dibuat.

// Memasukkan elemen <h1> kedalam elemen <body>

body.appendChild(heading);

**Langkah 5**: Kita dapat menambahkan *attribute* setiap elemen HTML. Kita akan menambahkan *attribute* id dengan menggunakan *method* createAttribute(). Lalu isi *attribute* tersebut dengan nilai "mainHeading" ke dalam *property* value. Setelah itu daftarkan *attribute* tersebut kepada elemen heading menggunakan *method* setAttributeNode("ATTRIBUTE\_NAME").

// Membuat attribut

let id = document.createAttribute('id');

// Memasukkan nilai attribut

id.value = "mainHeading";

// Set attribute pada <h1>

heading.setAttributeNode(id);

**Langkah 6**: Kita akan membuat konten utama berupa teks "Ini Heading" menggunakan method createTextNode("TEKS") yang membutuhkan parameter berupa **teks** yang akan dimasukkan kedalam elemen HTML.

Lalu konten teks tersebut akan kita masukkan pada elemen <h1> menggunakan method appendChild("NODE") yang membutuhkan parameter berupa *node object* dari elemen atau konten teks yang telah dibuat:

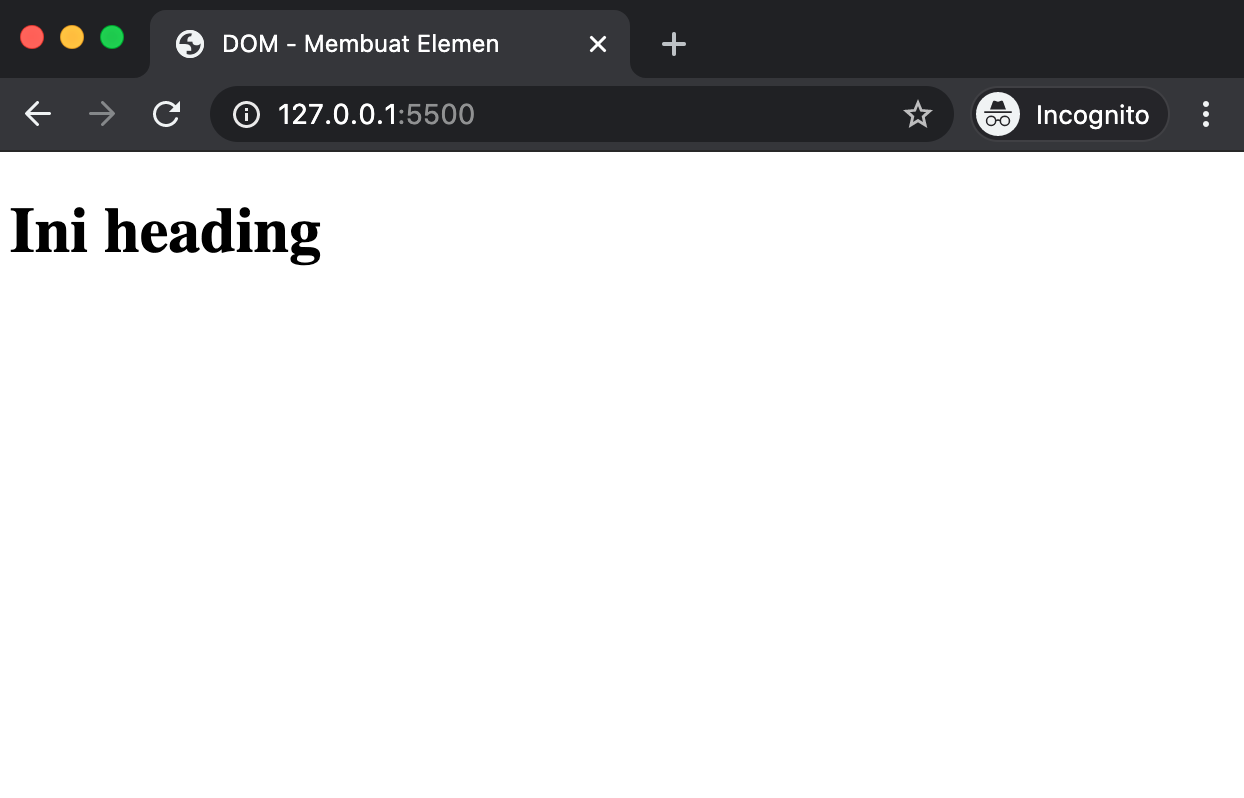
// Membuat konten pada elemen

let contentHeading = document.createTextNode('Ini heading');

// Memasukkan konten yang telah dibuat kedalam elemen heading <h1>

heading.appendChild(contentHeading);

Saat kamu mengakses *file* index.html, maka akan terlihat tampilan web seperti di bawah ini:



# Menghapus Elemen

Pada topik sebelumnya, kamu sudah belajar bagaimana caranya membuat sebuah elemen baru. Tidak hanya sebatas itu, ternyata kamu juga bisa **menghapus elemen** yang sudah ada.

Sebuah elemen dapat dihapus menggunakan method remove() pada objek elemen yang sudah kita buat.

Berikut ini struktur dari penggunaan method remove():

objekElemen.remove();

Untuk dapat memahami lebih mendalam, kita akan membuat sebuah aplikasi web sederhana. Berikut ini langkah-langkah untuk menghapus elemen HTML menggunakan JavaScript.

**Langkah 1**: Kita akan membuat file bernama index.html yang berisi kode seperti berikut ini:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>DOM - Elemen</title>

</head>

<body>

<h1 id="title">Skilvul</h1>

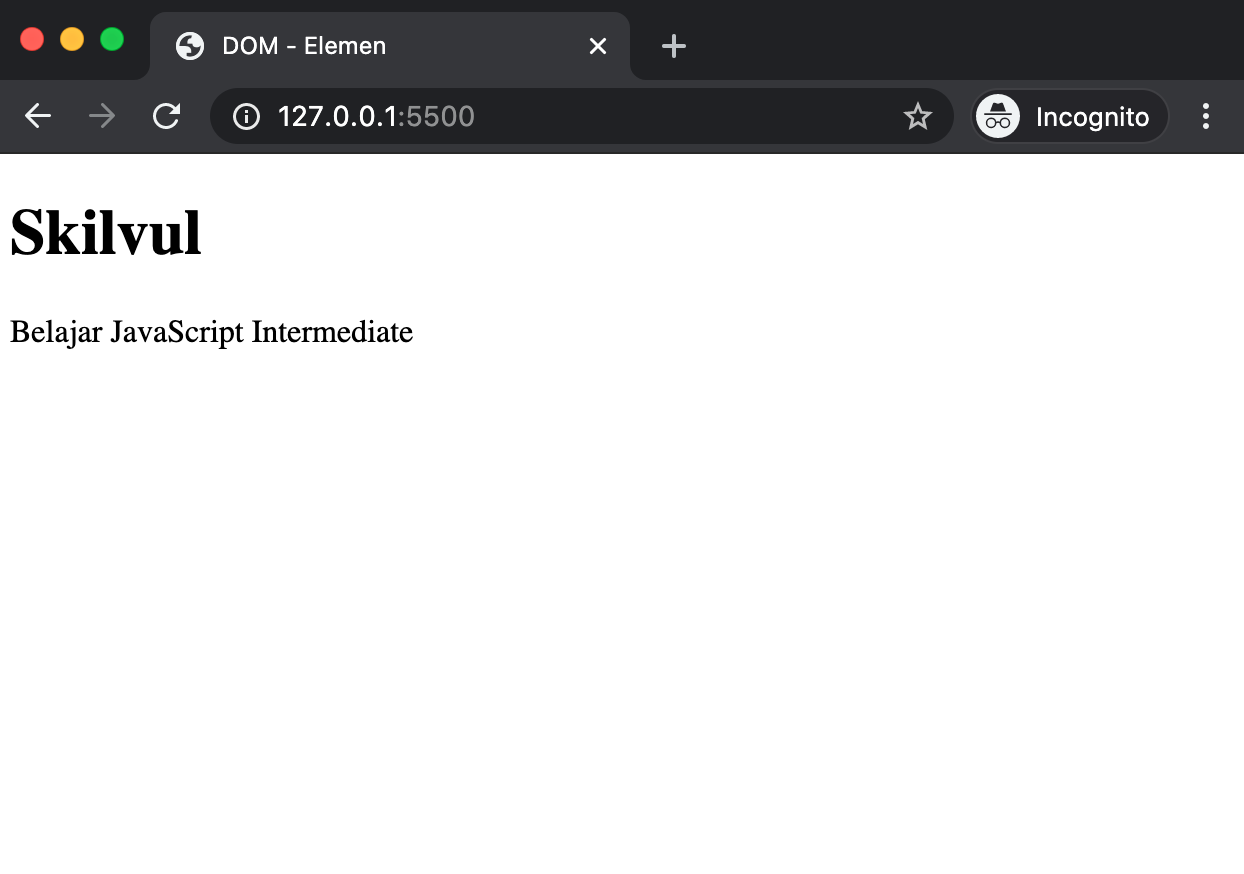
<p>Belajar JavaScript Intermediate</p>

<script src="elemen.js"></script>

</body>

</html>

Jika file index.html dijalankan, maka akan terlihat halaman web seperti ini:



**Langkah 2**: Kita akan membuat file bernama elemen.js. Lalu kita akan menghapus elemen <h1> yang memiliki id="title" menggunakan method remove(). Kita harus mendapatkan terlebih dahulu elemen HTML yang ingin kita hapus menggunakan method getElementById("ID\_ELEMEN\_HTML") yang memerlukan parameter berupa value dari attribute id elemen HTML tersebut.

let h1 = document.getElementById("title");

h1.remove();

Jika telah berhasil menghapus elemen, saat menjalankan file index.html kembali maka akan terlihat halaman seperti berikut ini:

