

Javascript

Dasar Javascript

- Javascript dirancang untuk membuat halaman HTML menjadi interaktif
- Javascript merupakan scripting language
- Scripting language adalah bahasa pemrograman yang kecil dan ringan
- Javascript tidak memerlukan proses compile
- Javascript berlisensi gratis
- Javascript bersifat case-sensitive

Penulisan Javascript

Javascript ditulis diantara tag `<script></script>`

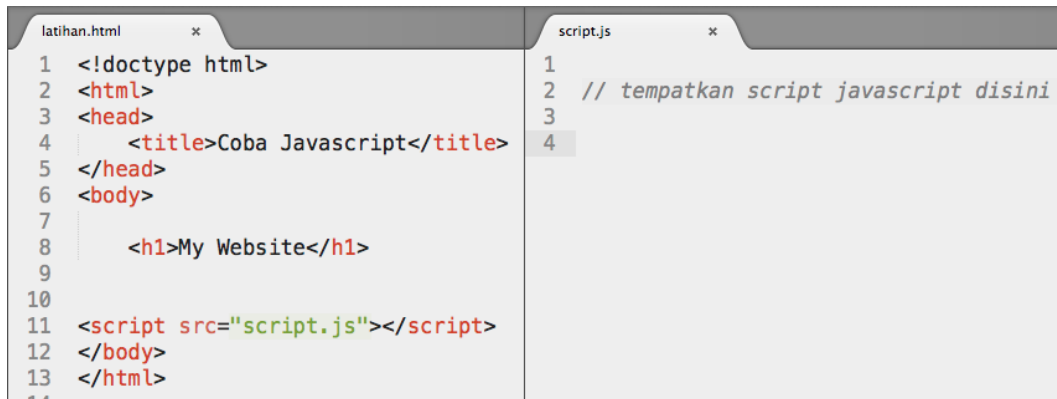
```
<script>  
    //tempatkan script javascript disini  
</script>
```

Beberapa penempatan code Javascript pada website adalah:

a. Di akhir halaman, sebelum `</body>`

```
<!doctype html>  
<html>  
<head>  
    <title>Coba Javascript</title>  
</head>  
<body>  
    <h1>My Website</h1>  
  
    <script>  
        //tempatkan script javascript disini  
    </script>  
</body>  
</html>
```

b. Di file eksternal



The image shows two side-by-side code editors. The left editor, titled 'latihan.html', contains the following HTML code:

```
1 <!doctype html>
2 <html>
3 <head>
4   <title>Coba Javascript</title>
5 </head>
6 <body>
7
8   <h1>My Website</h1>
9
10
11 <script src="script.js"></script>
12 </body>
13 </html>
```

The right editor, titled 'script.js', contains the following JavaScript code:

```
1
2 // tempatkan script javascript disini
3
4
```

Variabel Javascript

- Digunakan untuk menampung data, baik itu integer, string, dll
- Tidak usah mendefinisikan tipe datanya
- Bersifat case-sensitive
- Nama variabel harus diawali dengan huruf atau *underscore*

```
var x = 25; // numerik / integer
var pi = 3.14; // float
var nama = "sandhika"; // string
var myBoolean = true; // boolean

// dapat diubah menjadi:

var x = 25,
    pi = 3.14,
    nama = "sandhika",
    myBoolean = true;
```

Operator Javascript

Ada beberapa operator yang terdapat pada javascript, berikut ini diantaranya:

- Aritmatika

Operator	Fungsi
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Sisa pembagian (mod)
++	Penaikan (increment)
--	Penurunan (decrement)

- Assignment/penugasan

Operator	Contoh	Ekuivalen dengan
=	\$x=2	\$x=2
+=	\$x+=2	\$x=\$x+2
-=	\$x-=2	\$x=\$x-2
=	\$x=2	\$x=\$x*2
/=	\$x/=2	\$x=\$x/2
%=	\$x%=2	\$x=\$x%2

- Perbandingan

Operator	Penggunaan
&&	and
and	and
	r
or	or
!	not

Function

- Sebuah fungsi berisi source code yang akan dieksekusi ketika dipanggil.
- Fungsi bisa mengembalikan nilai ke pemanggilnya.
- Terdiri dari fungsi built-in, fungsi deklarasi, fungsi ekspresi
- Fungsi deklarasi dan fungsi ekspresi merupakan fungsi yang kita buat sendiri untuk membantu mempermudah pekerjaan, perbedaannya hanya pada cara pembuatannya.

a. Fungsi Built-in, adalah fungsi-fungsi bawaan yang dimiliki oleh javascript yang dapat kita gunakan pada script yang kita buat.

```
var x = "25";
var y = "3.14";

var toNum = parseInt(x); // fungsi built-in untuk mengubah string menjadi numerik
var toFloat = parseFloat(y); // fungsi built-in untuk mengubah string menjadi float
```

b. Fungsi Deklarasi

```
// Fungsi Deklarasi

function jumlah(a, b) {
    return a + b;
}

alert(jumlah(10, 10)); // menampilkan 20
```

c. Fungsi Ekspresi

```
// Fungsi Ekspresi

var jumlah = function(a, b) {
    return a + b;
}

alert(jumlah(5, 20)); // menampilkan 25
```

Object

- "Semua" yang ada di javascript merupakan Object: String, Number, Array, dll
- Javascript memungkinkan kita untuk mendefinisikan Objek kita sendiri
- Javascript juga memiliki built-in Object yang dapat juga dilihat disini.
- Setiap Object dapat memiliki Property dan Method

a. Mengakses property built-in

```
var pesan = "Hello World!"; // variabel yang bisa juga dianggap sebagai Object
var x = pesan.length; // .length adalah property built-in untuk objek di Javascript
// berfungsi untuk menghitung panjang string

alert(x); // akan menampilkan 12
```

b. Mengakses method built-in

```
var pesan = "Hello World!";  
var x = pesan.toUpperCase(); // method javascript untuk mengubah string -> uppercase  
  
alert(x); // akan menampilkan "HELLO WORLD!"
```

c. Membuat Object, Property dan Method sendiri

```
var orang = new Object(); // instansiasi / pembuatan object  
  
orang.namaDepan = "Sandhika"; // properti  
orang.namaBelakang = "Galih"; // properti  
orang.umur = 27; // properti  
orang.namaLengkap = function(){ // method  
    return orang.namaDepan + " " + orang.namaBelakang;  
}  
  
alert(orang.namaLengkap()); // pemanggilan method, menampilkan "Sandhika Galih"
```

Penulisannya dapat diubah menjadi:

```
var orang = {  
    namaDepan : "Sandhika",  
    namaBelakang : "Galih",  
    umur : 27,  
    namaLengkap : function(){  
        return orang.namaDepan + " " + orang.namaBelakang;  
    }  
};  
  
alert(orang.namaLengkap());
```

Array

Array adalah object javascript yang digunakan untuk menyimpan lebih dari satu nilai pada sebuah variabel. Contoh penggunaan Array

```
var foo = new Array(11, "Hello", true);
var value = foo[0]; // mengambil nilai dari array dengan index ke-0

foo[0] = 12; // mengganti nilai dari array dengan index ke-0
foo[3] = 15; // menambahkan nilai baru pada array di index ke-3

foo.push(20); // method untuk mengisi nilai pada elemen baru di akhir array

alert(foo[foo.length-1]); // menampilkan isi array index terakhir
```

Dapat juga ditulis seperti ini:

```
var foo = [11, "Hello", true]; // pendeklarasian array versi baru
var value = foo[0];

foo[0] = 12;
foo[3] = 15;

foo.push(20);

alert(foo[foo.length-1]);
```

Pengkondisian

Terdapat beberapa jenis pengkondisian pada javascript, berikut ini diantaranya:

- if..
- if.. else..
- if.. else if.. else
- switch

a. if..

```
if (5 < 6) {
    alert("5 lebih kecil dari 6");
};

alert("ini diluar pengkondisian");
```

b. if.. else..

```
var x = 5;

// mengecek apakah x bilangan ganjil atau genap
if ((x % 2) == 0) { // x habis dibagi 2, maka bilangan genap

    alert(x + " adalah bilangan GENAP");

} else {

    alert(x + " adalah bilangan GANJIL");

}

alert("ini diluar pengkondisian");
```

c. if.. else if.. else..

```
var jam = 9;

alert("Sekarang jam " + jam);

if ((jam < 11) && (jam > 4)) {

    alert("Selamat Pagi!");

} else if ((jam >= 11) && (jam <= 6)) {

    alert("Selamat Sore!");

} else {

    alert("Selamat Malam!")

}

alert("ini diluar pengkondisian");
```

Pengulangan

Terdapat 2 jenis pengulangan pada javascript, yaitu :

a. for

- Digunakan untuk mengulang sebuah blok perintah sebanyak beberapa kali sesuai dengan kebutuhan.
- Ada kemungkinan untuk tidak mengeksekusi blok perintah jika kondisi pertama bernilai false.

Contoh pengulangan isi array

```
var mahasiswa = ["Andi" , "Budi", "Dian", "Maria"];  
for (var i = 0; i < mahasiswa.length; i++) {  
    alert(mahasiswa[i]);  
}
```

b. while

```
var=nilaiAwal;  
while(var<=nilaiAkhir)  
{  
    code yang akan dieksekusi  
    var=var+increment;  
}
```

```
var mahasiswa = ["Andi" , "Budi", "Dian", "Maria"];  
var i = 0;  
while (i < mahasiswa.length) {  
    alert(mahasiswa[i]);  
    i++;  
}
```

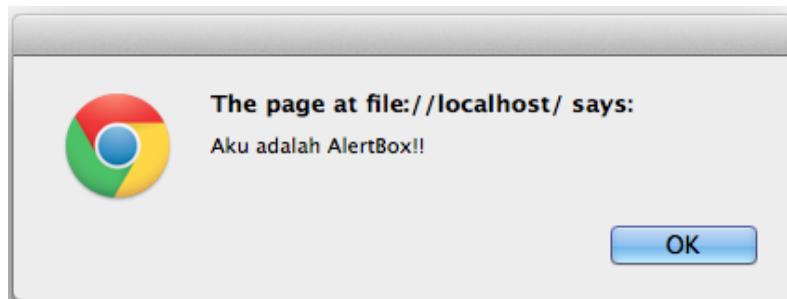
Popup

Javascript memiliki 3 buah popup, yaitu:

a. Alert Box, digunakan untuk memberikan informasi ke pengguna.

```
alert("Aku adalah AlertBox!!");
```

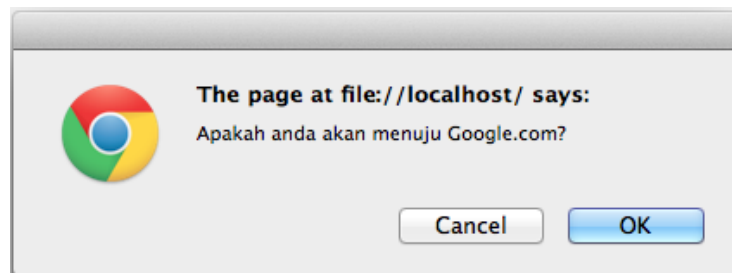
hasil:



b. Confirm Box, digunakan apabila aplikasi membutuhkan verifikasi atau persetujuan dari pengguna.

```
if (confirm("Apakah anda akan menuju Google.com?")) {  
    // jika memilih OK  
    location = "http://google.com"; // digunakan untuk pindah ke halaman lain  
} else {  
    // jika memilih CANCEL  
    alert("Tetap disini!");  
}
```

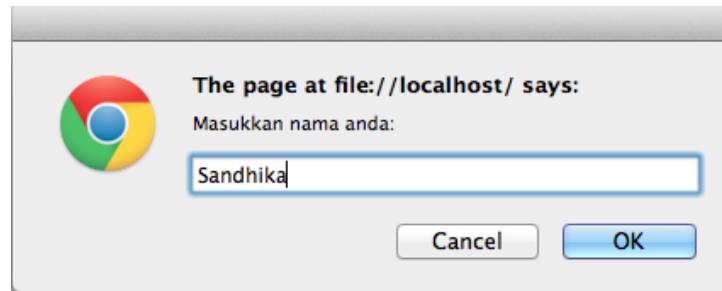
hasil:



c. Prompt Box, digunakan untuk menyimpan inputan dari user

```
var nama = prompt("Masukkan nama anda: "); // inputan user disimpan di variabel nama  
if (nama != null && nama != "") {  
    alert("Halo " + nama + ", apa kabar anda?");  
}
```

hasil:



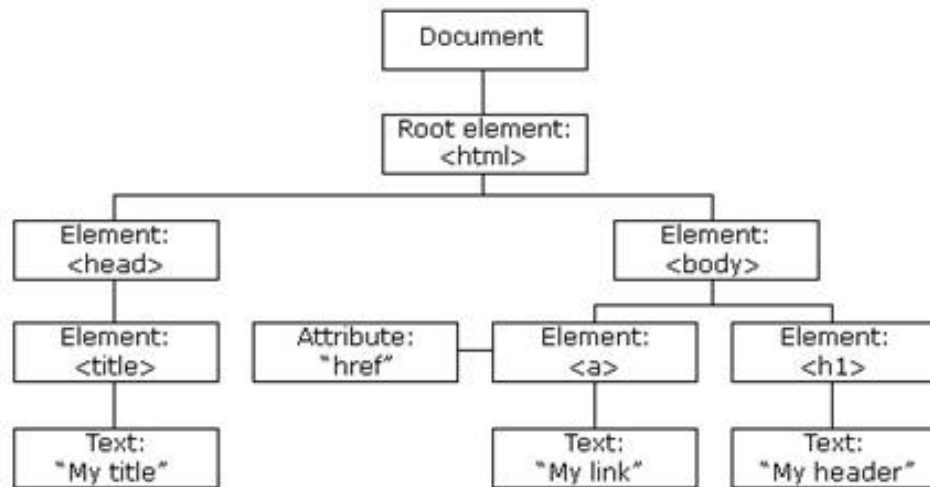
HTML + DOM

- DOM, singkatan dari Document Object Model, adalah sebuah cross-platform dan sebuah bahasa independen untuk merepresentasikan dan berinteraksi dengan objek dalam HTML, XHTML, dan dokumen XML.
- HTML DOM adalah cara baku untuk mengakses dan memanipulasi dokumen HTML.
- Setiap dokumen HTML yang ditampilkan pada browser akan menjadi sebuah Document Object.
- Sebuah Document Object menyediakan akses ke semua elemen HTML di dalam halaman.
- Setiap elemen HTML dapat diakses dengan JavaScript atau bahasa pemrograman lain.

Struktur DOM

Dalam HTML DOM, semua yang berada di dokumen HTML adalah sebuah node (titik)

- Seluruh dokumen HTML disebut document node.
- Setiap elemen HTML disebut element node.
- Text di dalam elemen HTML disebut text node.
- Setiap atribut TAG HTML disebut attribute node.
- Komentar disebut comment node.



Properti DOM

Berikut ini adalah properti dari DOM.

Properti	Fungsi
<code>x.innerHTML</code>	Menuliskan atau mengambil text dari elemen <code>x</code> .
<code>x.nodeName</code>	Memberi atau mengambil nama elemen <code>x</code> .
<code>x.nodeValue</code>	Memberi atau mengambil nilai dari elemen <code>x</code> .
<code>x.parentNode</code>	Mengambil informasi <i>parent</i> dari elemen <code>x</code> .
<code>x.childNodes</code>	Mengambil informasi <i>child</i> dari elemen <code>x</code> .
<code>x.attributes</code>	Memberi atau mengambil informasi atribut dari elemen <code>x</code> .

Dengan `x` adalah suatu elemen HTML (node).

Fungsi DOM

Berikut ini beberapa fungsi yang dimiliki oleh DOM

Nama Fungsi	Kegunaan
<code>x.getElementById(id)</code>	Mengakses elemen dengan <i>ID</i> tertentu.
<code>x.getElementsByTagName(name)</code>	Mengakses seluruh elemen dengan <i>name</i> tertentu.
<code>x.appendChild(node)</code>	Menambahkan elemen (<i>child</i>) ke suatu elemen <i>x</i> .
<code>x.removeChild(node)</code>	Menghapus elemen (<i>child</i>) dari suatu elemen <i>x</i> .

Dengan *x* adalah suatu elemen HTML (*node*).

Javascript DOM Event

Dengan menggunakan DOM, kita dapat mendaftarkan event handler yang berbeda untuk tiap-tiap elemen pada dokumen HTML. Sebuah event pada dasarnya merupakan kombinasi dari fungsi.

Daftar event pada javascript:

a. Javascript Mouse Event

Property	Deskripsi
onclick	sebuah event yang terjadi ketika user melakukan klik terhadap sebuah elemen
ondblclick	sebuah event yang terjadi ketika user melakukan double-klik terhadap sebuah elemen
onmousedown	sebuah event yang terjadi ketika user melakukan klik+menahan tombol terhadap sebuah elemen
onmousemove	sebuah event yang terjadi ketika user melakukan menggerakkan pointer mouse diatas sebuah elemen
onmouseover	sebuah event yang terjadi ketika user melakukan meletakkan pointer mouse diatas sebuah elemen
onmouseout	sebuah event yang terjadi ketika user melakukan menggerakkan pointer mouse keluar/meninggalkan sebuah elemen
onmouseup	sebuah event yang terjadi ketika user melepaskan tombol klik mouse terhadap sebuah elemen

b. Javascript Keyboard Event

Property	Deskripsi
onkeydown	sebuah event yang terjadi ketika user menekan tombol pada keyboard
onkeypress	sebuah event yang terjadi ketika user menekan dan menahan tombol pada keyboard
onkeyup	sebuah event yang terjadi ketika user melepaskan tombol pada keyboard

c. Javascript Form Event

Property	Deskripsi
onblur	sebuah event yang terjadi ketika sebuah elemen kehilangan fokus. (cth: kursor meninggalkan textfield)
onchange	sebuah event yang terjadi ketika konten pada sebuah elemen form berubah keadaan (untuk <input>, <select>, dan <textarea>)
onfocus	sebuah event yang terjadi ketika sebuah elemen mendapatkan fokus (untuk <label>, <input>, <select>, <textarea>, dan <button>)
onreset	sebuah event yang terjadi ketika sebuah form di-reset
onselect	sebuah event yang terjadi ketika user menyeleksi sebagian teks pada <input> atau <textarea>
onsubmit	sebuah event yang terjadi ketika sebuah form di-submit

d. Javascript Event Method

Method	Deskripsi
preventDefault()	Digunakan untuk menggagalkan sebuah event default dari sebuah elemen.