



NAMA MATA KULIAH



**UNIVERSITAS
INABA**

**MODUL – BAB VI
JAVASCRIPT**

Dedy Aridarma, S.Kom, M.Kom



JAVASCRIPT

Kemampuan yang diharapkan:

1. Mahasiswa mengetahui konsep dasar JavaScript dan kegunaan JavaScript dalam pemrograman web

HTML DOM

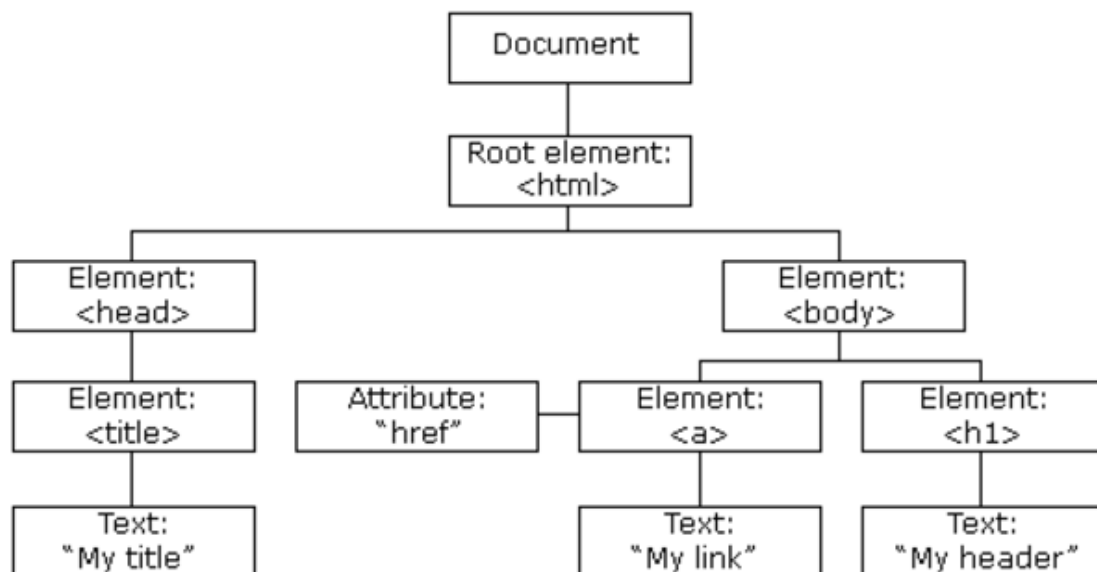
1. HTML DOM (Document Object Model)

yaitu standar yang digunakan untuk manipulasi elemen HTML. Dengan menggunakan HTML DOM, JavaScript dapat mengakses elemen-elemen HTML untuk dimanipulasi.

2. HTML DOM mendefinisikan:

- a) Elemen HTML sebagai objek
- b) Method adalah suatu aksi yang bisa dilakukan pada elemen HTML
- c) Property adalah nilai yang ingin kita ubah dari suatu elemen HTML

HIERARKI OBJEK

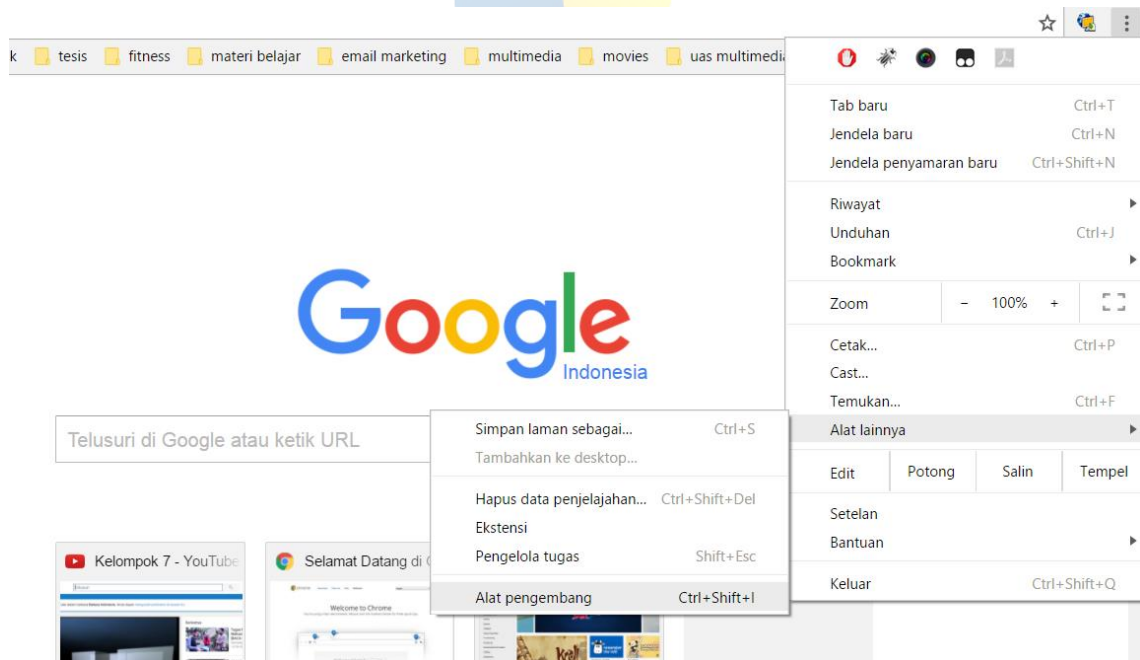


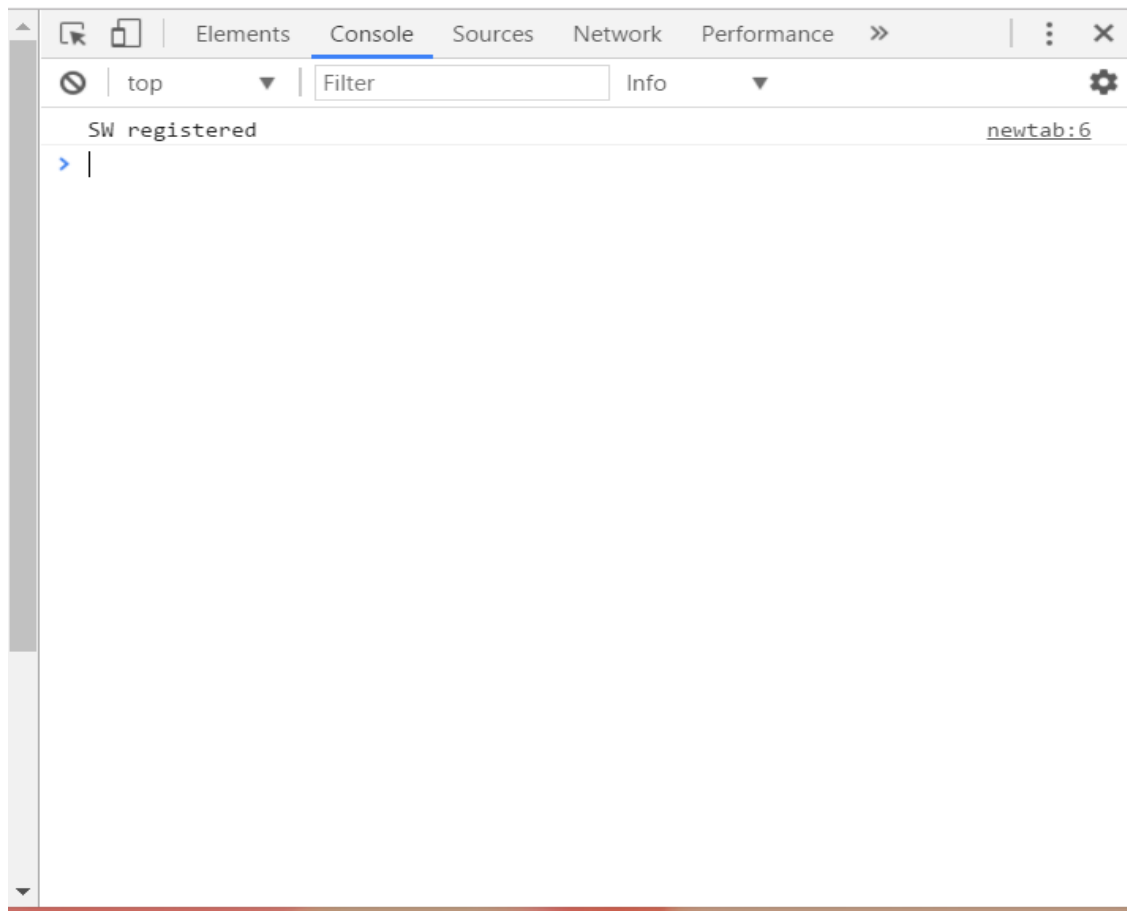
Apa Itu JavaScript?

JavaScript adalah bahasa scripting yang dapat berjalan pada sisi client (client-side programming), yang berarti bahwa perintah JavaScript dieksekusi disisi client sehingga client dapat melihat script dari program yang dibuat menggunakan JavaScript. Beda halnya dengan bahasa scripting PHP yang berjalan pada sisi server (server-side programming).

Melihat Pesan Kesalahan Pada JavaScript

Beda halnya dengan PHP yang dapat menampilkan pesan kesalahan via body web browser, JavaScript menggunakan console web browser untuk menampilkan pesan kesalahan. Untuk menampilkan console, silahkan lihat slide berikutnya.





Pengetahuan Dasar Dalam Pengembangan Web

HTML

Untuk mendefinisikan konten (contoh konten: teks, gambar, tabel, dll) pada suatu halaman web menggunakan elemen HTML.

CSS

Untuk memberi detail visual dan tata letak pada suatu halaman web menggunakan properti CSS.

JavaScript

Untuk memberi kesan suatu halaman web "bernyawa".

Apa yang bisa dilakukan JavaScript?

1. Mengubah konten HTML pakai innerHTML
2. Mengubah atribut HTML pakai atribut
3. Mengubah style CSS pakai style
4. Membuat slide image
5. dll

Menyisipkan JavaScript Pada HTML

- JavaScript dapat disisipkan pada elemen <head> </head> atau elemen <body> </body>.
- Perintah JavaScript disimpan diantara elemen <script> </script>.
- Contoh:

```
<html>
<head>
  <title> Untitled </title>
  <script>
    //cara 1 isi perintah JS
  </script>
</head>
<body>
  <script>
    //cara 2 isi perintah JS
  </script>
```

Penyimpanan JavaScript

1. Internal: disisipkan pada file HTML menggunakan elemen <script></script>
2. External: diluar file HTML, yaitu instruksi JavaScript disimpan pada file berekstensi .js. Kemudian file tersebut dipanggil melalui file HTML dengan menambahkan atribut src="namafile.js" pada elemen <script></script>.

Cara Menampilkan JavaScript

1. Pada elemen HTML menggunakan innerHTML property
2. Pada output HTML menggunakan document.write()
3. Pada suatu box alert menggunakan window.alert() atau alert()
4. Pada console web browser menggunakan console.log()

innerHTML

1. Fungsinya adalah menampilkan JavaScript melalui elemen HTML.
2. Untuk mengakses elemen HTML, JavaScript perlu mengetahui ID, class, atau tagname dari elemen HTML tersebut.
3. Contoh: mengubah elemen HTML untuk ID tertentu. JavaScript dapat menggunakan syntax: `document.getElementById(namaID).innerHTML = hasilTerbaru;` untuk menampilkan hasil instruksinya

Document.write()

1. Syntax `document.write()` biasa digunakan untuk melakukan testing atau debug suatu instruksi JavaScript
2. Contoh: tampilkan tulisan “Selamat Datang!” dengan menggunakan `document.write()`.
3. Untuk melakukan perintah tersebut, penulisan JavaScriptnya adalah: `document.write(“Selamat Datang!”);`

Window.alert()

1. Syntax `window.alert()` digunakan untuk menampilkan box alert pada web browser
2. Contoh: tampilkan tulisan “Selamat Datang!” dengan menggunakan `window.alert()`.



3. Untuk melakukan perintah tersebut, penulisan JavaScriptnya adalah:
`window.alert("Selamat Datang!");`

Console.log()

1. Syntax `console.log()` digunakan untuk menampilkan JavaScript pada console web browser.
2. Perlu dipahami bahwa penggunaan `console.log()` tidak akan menampilkan output pada body web browser.
3. Contoh: tampilkan tulisan "Selamat Datang!" dengan menggunakan `console.log()`.
4. Untuk melakukan perintah tersebut, penulisan JavaScriptnya adalah:
`console.log("Selamat Datang!");`

JavaScript Syntax

1. Syntax JavaScript merupakan satu set aturan untuk membentuk perintah/instruksi program JavaScript
2. Instruksi program pada JavaScript disebut statement
3. Antar statement dipisahkan menggunakan semikolon (titik-dua koma ";")
4. Statement biasanya terdiri dari values, variables, operators, keywords, dan comments

JavaScript Statement

1. Statement adalah instruksi yang harus dieksekusi oleh web browser
2. Contoh statement untuk menampilkan Hello World!:

```
<h2 id="hello">Data akan ditampilkan disini!</h2>
<script>
    document.getElementById("hello").innerHTML = "Hello
World!";
</script>
```

Values

1. Values adalah nilai yang akan ditampilkan browser menggunakan instruksi JavaScript.
2. Posisi values dalam statement JavaScript berada setelah tanda “=”
3. Contoh penulisan values:
 - a) Tipe data number: `document.getElementById(“namaId”).innerHTML = 10;`
-> Value dari instruksi poin a adalah 10
 - a) Tipe data string: `document.getElementById(“namaId”).innerHTML = “Ratih”;`
-> Value dari instruksi poin b adalah Ratih

Variabel

Variabel digunakan untuk menyimpan data Values.

Gunakan keyword “var” untuk mendeklarasikan suatu variabel

```
<h2 id="namaHewan"></h2>
<script>
  var hewan;
  hewan = 'kuda';
  document.getElementById('namaHewan').innerHTML =
  hewan;
</script>
```

Aturan Variabel

1. case-sensitive
2. gunakan huruf, angka, atau underscore
3. tdk boleh ada spasi, bila ada dua kata dalam variabel, gunakan metode camel upper (contoh: NamaVariabel) atau camel lower (contoh: namaVariabel)
4. tidak boleh diawali angka
5. tidak boleh ada dua atau lebih nama variabel yang sama

Tipe Data

Table 13-1 Data Types		
<i>Type</i>	<i>Description</i>	<i>Example</i>
Number	Any numeric value	<code>42</code>
String	Text characters inside quote marks	<code>"My name is Inigo Montoya"</code>
Object	A JavaScript object, which can be defined by the language or else created on your own	<code>window</code>
Function	Value returned by a function	<code>myFunction()</code>
Boolean	A logically true or false value	<code>true</code>
Null	Empty; has no value	<code>null</code>

JavaScript Operators

1. Assignment operators, contoh: menggunakan “=” untuk menetapkan suatu nilai pada variabel atau sebaliknya.

Table 13-2 Assignment Operators		
<i>Operator</i>	<i>Assignment</i>	<i>Description</i>
<code>=</code>	<code>x = y</code>	Sets <code>x</code> to the value of <code>y</code>
<code>+=</code>	<code>x += y</code>	Same as <code>x=x + y</code>
<code>-=</code>	<code>x -= y</code>	Same as <code>x=x - y</code>
<code>*=</code>	<code>x *= y</code>	Same as <code>x=x * y</code>
<code>/=</code>	<code>x /= y</code>	Same as <code>x=x / y</code>

2. Arithmetic operators, penggunaan operasi aritmatika seperti +, -, /, atau *

Table 13-3 Arithmetic Operators

<i>Operator</i>	<i>Example</i>	<i>Description</i>
+	$x + y$ (numeric)	Adds x and y together
-	$x - y$	Subtracts y from x
*	$x * y$	Multiplies x and y together
/	x / y	Divides x by y
-	$-x$	Reverses the sign of x

Table 13-4 Counting Operators

<i>Operator</i>	<i>Description</i>
$++x$	Increases y by 1 (same as $x=x+1$) before an assignment
$x++$	Increases y by 1 after an assignment
$--x$	Decreases y by 1 (same as $x=x-1$) before an assignment
$x--$	Decreases y by 1 after an assignment

Table 13-5 Comparison Operators

<i>Operator</i>	<i>Example</i>	<i>Description</i>
$==$	$x == y$	Returns <i>true</i> if x and y are equal
$!=$	$x != y$	Returns <i>true</i> if x and y are not equal
$>$	$x > y$	Returns <i>true</i> if x is greater than y
$>=$	$x >= y$	Returns <i>true</i> if x is greater than or equal to y
$<$	$x < y$	Returns <i>true</i> if x is less than y
$<=$	$x <= y$	Returns <i>true</i> if x is less than or equal to y
$ $	$x y$	Returns <i>true</i> if either x or y is true
$&&$	$x \&\& y$	Returns <i>true</i> if both x and y are true
$!$	$!x$	Returns <i>true</i> if x is false

Contoh Arithmetic operators

```
<script>
    var a = 15;
    var b = 4;
    var c = 0;

    // pengurangan
    c = a - b;
    document.write(`${a} - ${b} =
    ${c}<br/>`);

    // Perkalian
    c = a * b;
    document.write(`${a} * ${b} =
    ${c}<br/>`);

    // pemangkatan
    c = a ** b;
    document.write(`${a} ** ${b} =
    ${c}<br/>`);

    // Pembagian
    c = a / b;
    document.write(`${a} / ${b} =
    ${c}<br/>`);

    // Modulo
    c = a % b;
    document.write(`${a} % ${b} =
    ${c}<br/>`);
</script>
```



Contoh Assignment operators:

```
<script>
    document.write("Mula-mula nilai score...<br>");
    // pengisian nilai
    var score = 100;
    document.write("score = "+ score + "<br/>");

    // pengisian dan menjumlahkan dengan 5
    document.write("Penjumlahan dengan
sepuluh...<br>");
    score += 10;
    document.write("score = "+ score + "<br/>");

    // pengisian dan pengurangan dengan 5
    document.write("Pengurangan dengan
lima...<br>");
    score -= 5;
    document.write("score = "+ score + "<br/>");

    // pengisian dan perkalian dengan 3
    document.write("Perkalian dengan 3...<br>");
    score *= 3;
    document.write("score = "+ score + "<br/>");

    // pengisian dan pembagian dengan 5
    document.write("Pembagian dengan 5...<br>");
    score /= 5;
    document.write("score = "+ score + "<br/>");

    // pengisian dan pemangkatan dengan 3
    document.write("Pemangkatan dengan 3...<br>");
    score **= 3;
    document.write("score = "+ score + "<br/>");

    // pengisian dan modulo dengan 3;
    document.write("Sisa Bagi dengan 2...<br>");
    score %= 2;
    document.write("score = "+ score + "<br/>");
</script>
```



Contoh Comparison Operators

```
<script>
  var a = 10;
  var b = 8;

  // sama dengan
  var hasil = a == b;
  document.write(`${a} == ${b} = ${hasil}<br/>`);

  // lebih besar
  var hasil = a > b;
  document.write(`${a} > ${b} = ${hasil}<br/>`);

  // lebih besar samadengan
  var hasil = a >= b;
  document.write(`${a} >= ${b} = ${hasil}<br/>`);

  // lebih kecil
  var hasil = a < b;
  document.write(`${a} < ${b} = ${hasil}<br/>`);

  // lebih kecil samadengan
  var hasil = a <= b;
  document.write(`${a} <= ${b} = ${hasil}<br/>`);

  // tidak samadengan
  var hasil = a != b;
  document.write(`${a} != ${b} = ${hasil}<br/>`);
</script>
```

Contoh Dialog Alert:

```
<html>
<head>
  <title> Untitled </title>
</head>
<body>
  <script>
    alert("Selamat datang di tutorial Javascript");
  </script>
</body>
</html>
```



Contoh Dialog Confirm:

```
<html>
<head>
  <title> Untitled </title>
</head>
<body>
  <button      onclick="konfirmasi()">Tombol
Hapus</button>
  <p id="pesan"></p>
  <script>
    function konfirmasi(){
      var tanya = confirm("Apakah Anda Akan
Menghapus Data Ini ?");
      if(tanya === true) {
        pesan = "Ya";
      }else{
        pesan = "Tidak";
      }

      document.getElementById("pesan").innerHTML =
pesan;
    }
  </script>
</body>
</html>
```

Contoh Dialog Prompt:

```
<html>
<head>
  <title> Untitled </title>
</head>
<body>
  <script>
    var nama = prompt("Siapa Nama Anda?","" );
    alert(nama);
    var usia = prompt("Berapa Usia Anda?","" );
    alert(usia);
  </script>
</body>
</html>
```



DAFTAR PUSTAKA

- Abdulloh, R. (2018). *7 in 1 Pemrograman Web Untuk Pemula*. Jakarta: PT Elex Media Komputindo
- Susanto, A. (2017). *Sistem Informasi Perpustakaan Berbasis Web Pada Perpustakaan Universitas Dharmawangsa*. *Djtechno : Journal of Information Technology Research* (Vol. 2). Bandung: Lingga Jaya.
- Marisa, F. (2017). *Web Programming (Client Side and Server Side)*. Deepublish (Cetakan 1). Yogyakarta: Deepublish.
- https://www.google.co.id/books/edition/HTML_PHP_dan_MySQL_untuk_Pemula/v17DwAAQBAJ?hl=id&gbpv=1&dq=html+pemula&pg=PA21&printsec=frontcover
- <https://www.w3schools.com/css/default.asp>

UNIVERSITAS
INABA

