

MODUL PRAKTIKUM PEMROGRAMAN WEB

MINGGU – 5



JavaScript, DOM, XML & JSON

Program Studi Teknik Informatika

Jurusan Teknik Elektro, Informatika, dan Sistem Fisis

Institut Teknologi Sumatera

2020

MODUL 10. JAVASCRIPT & DOM

PENGERTIAN JAVASCRIPT

Javascript adalah bahasa pemrograman yang awalnya dirancang untuk berjalan di atas browser. Namun, seiring perkembangan zaman, javascript tidak hanya berjalan di atas browser saja. Javascript juga dapat digunakan pada sisi Server, Game, IoT, Desktop, dsb. Javascript awalnya bernama Mocha, lalu berubah menjadi LiveScript saat browser Netscape Navigator 2.0 rilis versi beta (September 1995). Namun, setelah itu dinamai ulang menjadi Javascript. Terinspirasi dari kesuksesan Javascript, Microsoft mengadopsi teknologi serupa. Microsoft membuat 'Javascript' versi mereka sendiri bernama JScript. Lalu di tanam pada Internet Explorer 3.0. Hal ini mengakibatkan 'perang browser', karena JScript milik Microsoft berbeda dengan Javascript racikan Netscape. Akhirnya pada tahun 1996, Netscape mengirimkan standarisasi ECMA-262 ke Ecma International. Sehingga lahirlah standarisasi kode Javascript bernama ECMAScript atau ES. Saat ini ECMAScript sudah mencapai versi 8 (ES8).

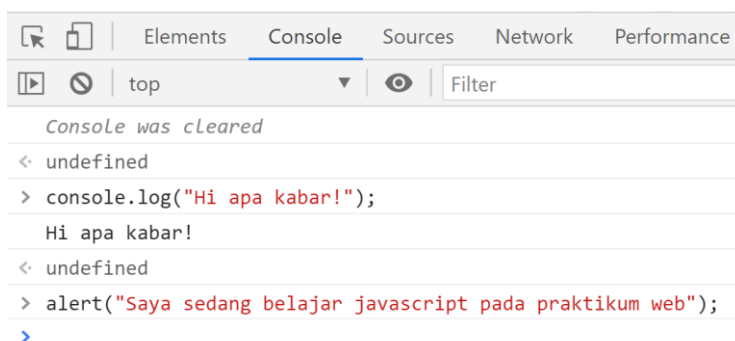
PENGERTIAN DOM (DOCUMENT OBJECT MODEL)

DOM merupakan singkatan dari Document Object Model. Artinya, dokumen (HTML) yang dimodelkan dalam sebuah objek. Objek dari dokumen ini menyediakan sekumpulan fungsi dan atribut/data yang bisa kita manfaatkan dalam membuat program Javascript. Inilah yang disebut API (Application Programming Interface).

JAVASCRIPT DASAR

1. Menggunakan Console

Console Javascript dapat kita buka melalui klik kanan->Inspect Element->Console (ctrl+shift+i).



Hasilnya

An embedded page at new-tab-page says
Saya sedang belajar javascript pada praktikum web

OK

2. Embed/Internal Script (Kode Javascript Ditempel Langsung Pada Html)

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Hello World Javascript</title>
5 </head>
6 <body>
7   <script>
8     alert("contoh embedded/internal script");
9   </script>
10 </body>
11 </html>
```

Hasilnya

127.0.0.1:5500 says
contoh embedded/internal script

OK

3. Inline (Kode Javascript Ditulis Pada Atribut Html)

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Hello World Javascript</title>
5 </head>
6 <body>
7   <a href="#" onclick="alert('contoh inline script')">Klik aku!</a>
8 </body>
9 </html>
```

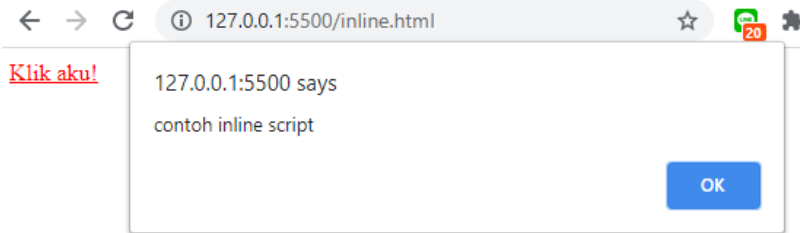
Atau

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Hello World Javascript</title>
5 </head>
6 <body>
7   <a href="javascript:alert('contoh inline script')">Klik aku!</a>
8 </body>
9 </html>

```

Hasilnya



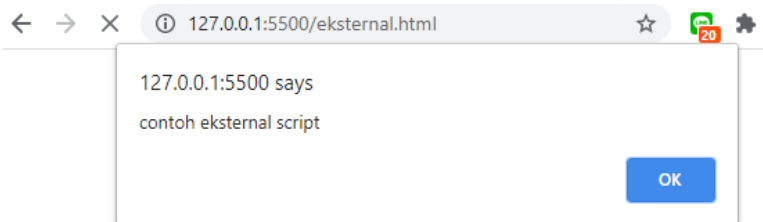
4. Eksternal (Kode Javascript Ditulis Terpisah Dengan File Html)

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Hello World Javascript</title>
5 </head>
6 <body>
7   <script src="eksternal.js"></script>
8 </body>
9 </html>

```

Hasilnya



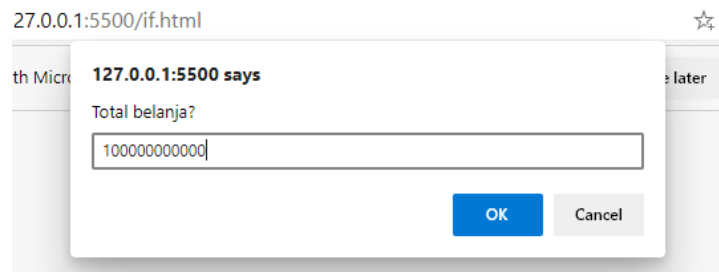
PERCABANGAN

1. percabangan if

Percabangan if merupakan percabangan yang hanya memiliki satu blok pilihan saat kondisi bernilai benar.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Percabangan if</title>
5 </head>
6 <body>
7   <script>
8     var totalBelanja = prompt("Total belanja?", 0);
9
10    if(totalBelanja > 100000){
11      document.write("<h2>Selamat Anda dapat hadiah</h2>");
12    }
13
14    document.write("<p>Terimakasih sudah berbelanja di toko kami</p>");
15  </script>
16 </body>
17 </html>
```

Hasilnya



Selamat Anda dapat hadiah

Terimakasih sudah berbelanja di toko kami

2. Percabangan If-Else

Percabangan if/else merupakan percabangan yang memiliki dua blok pilihan. Pilihan pertama untuk kondisi benar, dan pilihan kedua untuk kondisi salah (else).

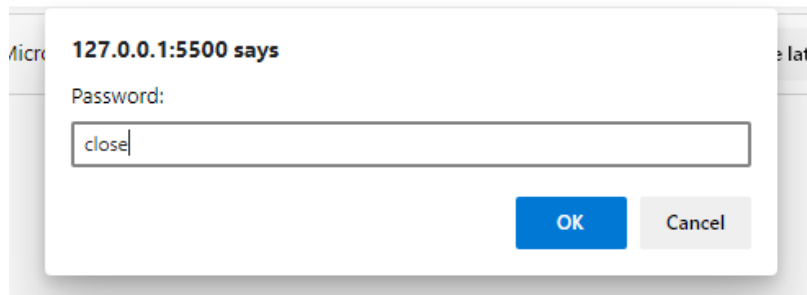
```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Percabangan if/else</title>
5 </head>
6 <body>
7   <script>
8     var password = prompt("Password:");
9
10    if(password == "open"){
11      document.write("<h2>Selamat datang mahasiswa praktikum!</h2>");
12    } else {
13      document.write("<p>Password salah, coba lagi!</p>");
14    }
15
16    document.write("<p>Terima kasih sudah menggunakan aplikasi ini!</p>");
17  </script>
18 </body>
19 </html>
20

```

Hasilnya

0.0.1:5500/if-else.html



Password salah, coba lagi!

Terima kasih sudah menggunakan aplikasi ini!

3. Switch case

Pada <value>, kita bisa isi dengan nilai yang nanti akan dibandingkan dengan varabel. Setiap case harus diakhiri dengan break. Khusus untuk default, tidak perlu diakhiri dengan break karena dia terletak di bagian akhir.

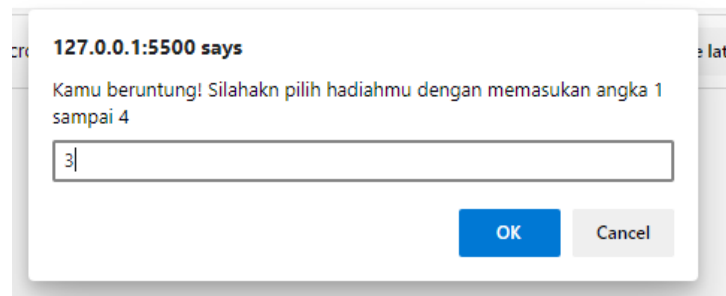
```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Percabangan switch/case</title>
5 </head>
6 <body>
7   <script>
8     var jawab = prompt("Kamu beruntung! Silahakn pilih hadiahmu dengan memasukan angka 1 sampai 5");
9     var hadiah = "";
10    switch(jawab){
11      case "1":
12        hadiah = "Tisu";
13        break;
14      case "2":
15        hadiah = "1 Kotak Kopi";
16        break;
17      case "3":
18        hadiah = "Sticker";
19        break;
20      case "4":
21        hadiah = "Minyak Goreng";
22        break;
23      default:
24        document.write("<p>Oops! anda salah pilih</p>");
25    }
26
27    if(hadiah === ""){
28      document.write("<p>Kamu gagal mendapat hadiah</p>");
29    } else {
30      document.write("<h2>Selamat kamu mendapatkan " + hadiah + "</h2>");
31    }
32  </script>
33 </body>
34 </html>

```

Hasilnya

0.1:5500/switch-case.html



Selamat kamu mendapatkan Sticker

PERULANGAN

1. For

Perulangan for merupakan perulangan yang termasuk dalam counted loop, karena sudah jelas berapa kali ia akan mengulang.

```
Select an element in the page to inspect it Ctrl+Shift+C
> for(counter = 0; counter < 10; counter+=2){
    document.write("<p>Perulangan ke-"+counter+"</p>");
}
< undefined
```

Hasilnya

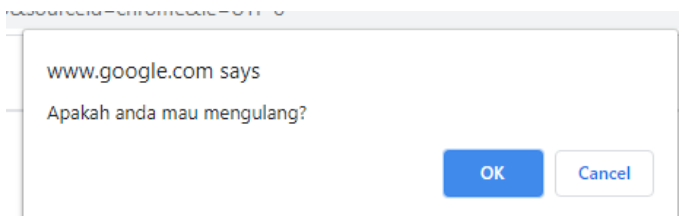
Perulangan ke-0
Perulangan ke-2
Perulangan ke-4
Perulangan ke-6
Perulangan ke-8

2. While

Perulangan while merupakan perulangan yang termasuk dalam perulangan uncounted loop. Perulangan while juga dapat menjadi perulangan yang counted loop dengan memberikan counter di dalamnya.

```
top Filter Default levels ▼
> var ulangi = confirm("Apakah anda mau mengulang?");
    var counter = 0;
    while(ulangi){
        counter++;
        ulangi = confirm("Apakah anda mau mengulang?");
    }
    document.write("Perulangan sudah dilakuakn sebanyak "+ counter +" kali");
```

Hasilnya



Jika menekan ok 2 kali

Perulangan sudah dilakuakn sebanyak 2 kali

3. Do/While

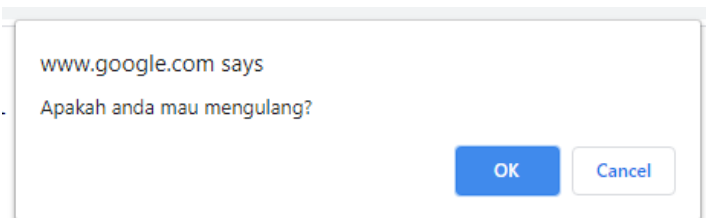
Perulangan do/while sama seperti perulangan while. Perbedaanya perulangan do/while akan melakukan perulangan sebanyak 1 kali terlebih dahulu, lalu mengecek kondisi yang ada di dalam kurung while.

```
Select an element in the page to inspect it Ctrl + Shift + C Default levels ▼
> var ulangi = confirm("Apakah anda mau mengulang?");;
  var counter = 0;

  do {
    counter++;
    ulangi = confirm("Apakah anda mau mengulang?");
  } while(ulangi)

  document.write("Perulangan sudah dilakuakn sebanyak "+ counter + " kali");|
```

Hasilnya



Jika menekan cancel

Perulangan sudah dilakuakn sebanyak 1 kali

4. Foreach

Perulangan foreach biasanya digunakan untuk mencetak item di dalam array. Perulangan ini termasuk dalam perulangan counted loop, karena jumlah perulangannya akan ditentukan oleh panjang dari array.

Select an element in the page to inspect it Ctrl+Shift+C

```
> var languages = ["Javascript", "HTML", "CSS", "Typescript"];

for(i in languages){
    document.write(i+". " + languages[i] + "<br/>");
}
```

Hasilnya

← → ↻ google.com/search?c

0. Javascript

1. HTML

2. CSS

3. Typescript

Menggunakan method forEach()

Toggle device toolbar Ctrl+Shift+M Filter Default levels ▼

```
> // kita punya array seperti berikut
var days = ["Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu", "Minggu"];

// Kemudian kita tampilkan semua hari
// dengan menggunakan method foreach
days.forEach(function(day){
    document.write("<p>" + day + "</p>");
});|
```

Hasilnya

← → ↻ google.com

Senin

Selasa

Rabu

Kamis

Jum'at

Sabtu

Minggu

5. Dengan Method repeat()

Perulangan dengan method atau fungsi repeat() termasuk dalam perulangan counted loop. Fungsi ini khusus digunakan untuk mengulang sebuah teks (string). Bisa dibilang ini merupakan singkatan dari perulangan for.

```
document.write("Harta, Tahta, Nilai A. ".repeat(10));
```

Hasilnya

Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A. Harta, Tahta, Nilai A.

FUNGSI

1. Tanpa Parameter

```
// membuat fungsi
function sayHello(){
  console.log("Hello World!");
}

// memanggil fungsi
sayHello() // maka akan menghasilkan -> Hello World!

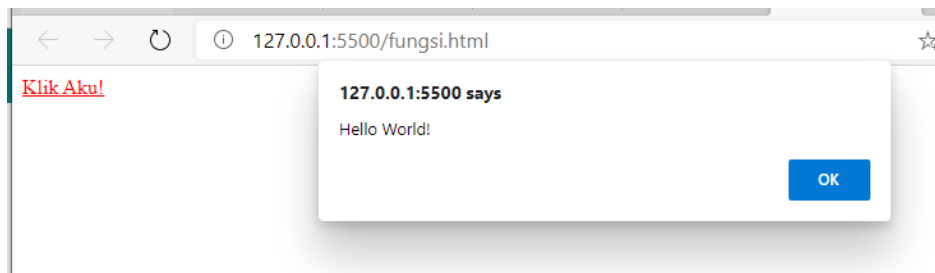
Hello World!

< undefined
```

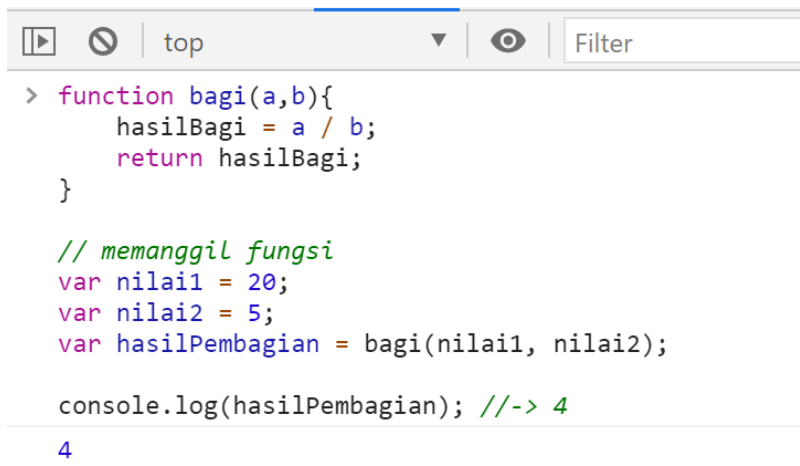
Atau

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <script>
5     // membuat fungsi
6     var sayHello = () => alert("Hello World!");
7   </script>
8 </head>
9 <body>
10   <!-- Memanggil fungsi saat link diklik -->
11   <a href="#" onclick="sayHello()">Klik Aku!</a>
12 </body>
13 </html>
```

Hasilnya



2. Dengan Parameter



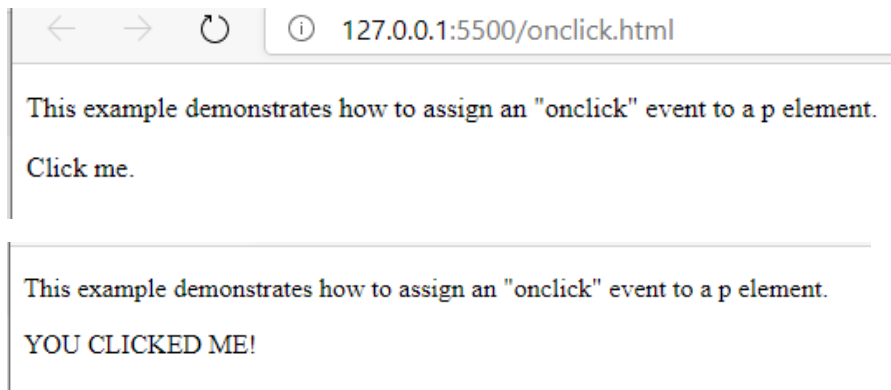
EVENT

1. OnClick

OnClick adalah suatu kejadian yang terjadi ketika sebuah elemen html di klik.

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <p>This example demonstrates how to assign an "onclick" event to a p element.</p>
5   <p id="demo" OnClick="myFunction()">Click me.</p>
6
7   <script>
8     function myFunction() {
9       document.getElementById("demo").innerHTML = "YOU CLICKED ME!";
10    }
11  </script>
12 </body>
13 </html>
```

Hasilnya

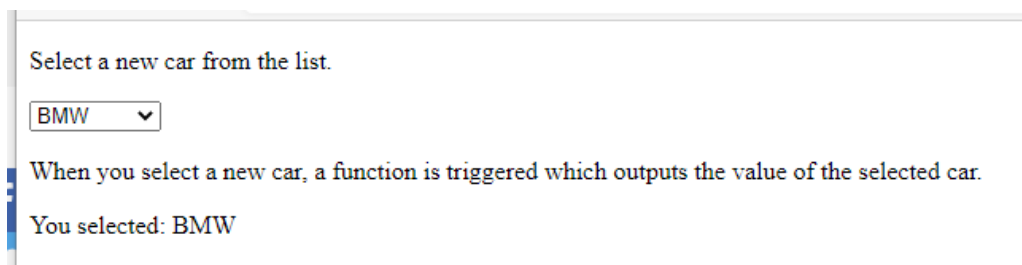


2. OnChange

OnChange adalah suatu kejadian yang terjadi ketika sebuah elemen html berubah.

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <p>Select a new car from the list.</p>
5   <select id="mySelect" OnChange="myFunction()">
6     <option value="Audi">Audi</option>
7     <option value="BMW">BMW</option>
8     <option value="Mercedes">Mercedes</option>
9     <option value="Volvo">Volvo</option>
10  </select>
11  <p>When you select a new car, a function is triggered which outputs the value of the selected car.</p>
12  <p id="demo"></p>
13  <script>
14    function myFunction() {
15      var x = document.getElementById("mySelect").value;
16      document.getElementById("demo").innerHTML = "You selected: " + x;
17    }
18  </script>
19 </body>
20 </html>
```

Hasilnya



3. OnMouseOver & OnMouseOut

OnMouseOver adalah suatu kejadian yang terjadi ketika sebuah elemen html di letakkan cursor mouse. OnMouseOut adalah suatu kejadian yang terjadi ketika cursor mouse meninggalkan elemen html.

```

1  <!DOCTYPE html>
2  <html>
3  <body>
4      <p>This example demonstrates how to assign an "onmouseover" and "onmouseout" event to a h1 element.</p>
5      <h1 id="demo" OnMouseOver ="mouseover()" OnMouseOut="mouseout()">Mouse over me</h1>
6
7      <script>
8          function mouseOver() {
9              document.getElementById("demo").style.color = "red";
10         }
11         function mouseOut() {
12             document.getElementById("demo").style.color = "black";
13         }
14     </script>
15 </body>
16 </html>

```

Hasilnya

This example demonstrates how to assign an "onmouseover" and "onmouseout" event to a h1 element.

Mouse over me

Ketika mouse mengenai teks mouse over me

This example demonstrates how to assign an "onmouseover" and "onmouseout" event to a h1 element.

Mouse over me

4. OnKeyDown

OnKeyDown adalah suatu kejadian yang terjadi ketika ada pengetikan pada elemen html.

```

1  <!DOCTYPE html>
2  <html>
3  <body>
4      <p>A function is triggered when the user is pressing a key in the input field.</p>
5      <input type="text" id="input" onkeydown="myFunction()">
6      <div id="text"></div>
7
8      <script>
9          function myFunction() {
10              var inputText = document.getElementById("input").value;
11              document.getElementById("text").innerHTML = inputText;
12          }
13     </script>
14 </body>
15 </html>

```

Hasilnya

A function is triggered when the user is pressing a key in the input field.

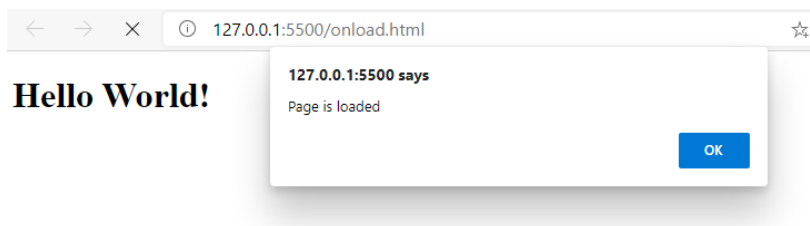
selamat datang |
selamat datang

5. OnLoad

OnLoad adalah suatu kejadian yang terjadi ketika elemen dan halaman di buka.

```
1 <!DOCTYPE html>
2 <html>
3 <body OnLoad="myFunction()">
4   <h1>Hello World!</h1>
5
6   <script>
7     function myFunction() {
8       alert("Page is loaded");
9     }
10  </script>
11 </body>
12 </html>
```

Hasilnya



Contoh event lainnya

| EVENT | KETERANGAN |
|-------------------|---|
| Abort (onAbort) | Terjadi ketika user menghentikan proses load image ke halaman web |
| Blur (onBlur) | Terjadi ketika elemen kehilangan fokus, artinya user mengklik di luar elemen tersebut |
| Change (onChange) | Terjadi ketika user memodifikasi isi data dari sebuah elemen input |
| Click (onClick) | Terjadi ketika user melakukan klik mouse terhadap suatu elemen |

| EVENT | KETERANGAN |
|----------------------------|---|
| Double Click (onDblclick) | Terjadi ketika user melakukan double click pada elemen |
| Drag and Drop (onDragdrop) | Terjadi ketika user melakukan drag and drop pada elemen |
| Error (onError) | Terjadi ketika terdapat error pada saat memuat halaman |
| Focus (onFocus) | Terjadi ketika user memberikan fokus pada elemen |
| Load (onLoad) | Terjadi pada saat browser memuat suatu halaman web |
| Reset (onReset) | Terjadi ketika user ingin mengembalikan data ke posisi awal pada form |
| Submit (onSubmit) | Terjadi ketika user ingin mengirimkan data yang telah diinput pada form |

| EVENT | OBJECT |
|---------------------------|--|
| Abort (onAbort) | Image |
| Blur (onBlur) | FileUpload, Password, Button, Checkbox, Radio, Reset, Select, Submit, Textarea |
| Change (onChange) | FileUpload, Select, Submit, Textarea |
| Click (onClick) | Document, Link, Button, Checkbox, Radio, Reset, Submit |
| Double Click (onDblclick) | Document, Link |

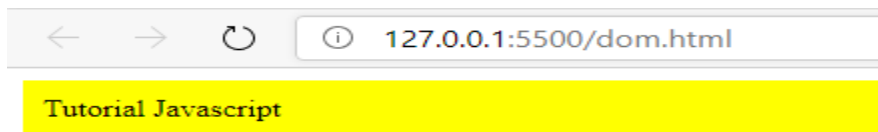
| EVENT | OBJECT |
|----------------------------|--|
| Drag and Drop (onDragdrop) | Window |
| Error (onError) | Image, Window |
| Focus (onFocus) | FileUpload, Password, Button, Checkbox, Radio, Reset, Select, Submit, Textarea, Window |
| Load (onLoad) | Image, Window |
| Reset (onReset) | Form |
| Submit (onSubmit) | Form |
| Resize (onResize) | Window |

DOM

getElementById() fungsi untuk memilih elemen berdasarkan atribut id.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Memilih Elemen Berdasarkan ID</title>
5 </head>
6 <body>
7
8   <!-- Elemen div yang akan kita pilih dari JS -->
9   <div id="tutorial"></div>
10
11
12   <script type="text/javascript">
13     // mengakses elemen tutorial
14     var tutorial = document.getElementById("tutorial");
15
16     // mengisi teks ke dalam elemen
17     tutorial.innerHTML = "Tutorial Javascript";
18
19     // memberikan CSS ke elemen
20     tutorial.style.backgroundColor = "yellow";
21     tutorial.style.padding = "10px";
22
23   </script>
24
25 </body>
26 </html>
```

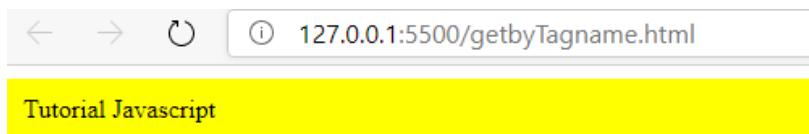
Hasilnya



getElementsByTagName() fungsi untuk memilih elemen berdasarkan nama tag.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Memilih Elemen Berdasarkan ID</title>
5 </head>
6 <body>
7
8   <!-- Elemen div yang akan kita pilih dari JS -->
9   <div></div>
10
11
12   <script type="text/javascript">
13     // mengakses elemen tutorial
14     var tutorial = document.getElementsByTagName("div")[0];
15
16     // mengisi teks ke dalam elemen
17     tutorial.innerText = "Tutorial Javascript";
18
19     // memberikan CSS ke elemen
20     tutorial.style.backgroundColor = "yellow";
21     tutorial.style.padding = "10px";
22
23   </script>
24
25 </body>
26 </html>
```

Hasilnya



Fungsi dom lainnya yang sering digunakan

getElementsByTagName() fungsi untuk memilih elemen berdasarkan atribut name.

getElementsByTagName() fungsi untuk memilih elemen berdasarkan atribut class.

getElementsByTagNameNS() fungsi untuk memilih elemen berdasarkan atribut class.

querySelector() fungsi untuk memilih elemen berdasarkan query.

querySelectorAll() fungsi untuk memilih elemen berdasarkan query.

TUGAS

1. Buatlah sebuah 1 file Javascript dan 1 file html yang berisi sebuah array yang menyimpan data alat yang ada di sekolah. Pengguna bisa menambah, mengubah dan menghapus data barang tersebut, semua barang tersebut ditampilkan pada halaman web.

Contoh output

Tambah barang

Input Form

- Buku Tulis [\[Edit | Hapus\]](#)
- Pensil [\[Edit | Hapus\]](#)
- Spidol [\[Edit | Hapus\]](#)

Input Form

- Buku Tulis [\[Edit | Hapus\]](#)
- Pensil [\[Edit | Hapus\]](#)
- Spidol [\[Edit | Hapus\]](#)
- kursi [\[Edit | Hapus\]](#)

Edit barang

Nama baru

Input Form

- Buku Tulis ada 100 [[Edit](#) | [Hapus](#)]
- Pensil [[Edit](#) | [Hapus](#)]
- Spidol [[Edit](#) | [Hapus](#)]
- kursi [[Edit](#) | [Hapus](#)]

Hapus barang

Input Form

- Buku Tulis ada 100 [[Edit](#) | [Hapus](#)]
- Spidol [[Edit](#) | [Hapus](#)]
- kursi [[Edit](#) | [Hapus](#)]

2. Buat contoh dari masing masing fungsi dom yang telah disebutkan diatas untuk memanipulasi file html.

getElementByName()

getElementByClassName()

getElementByTagNameNS()

querySelector()

querySelectorAll()

Modul 11. XML & JSON

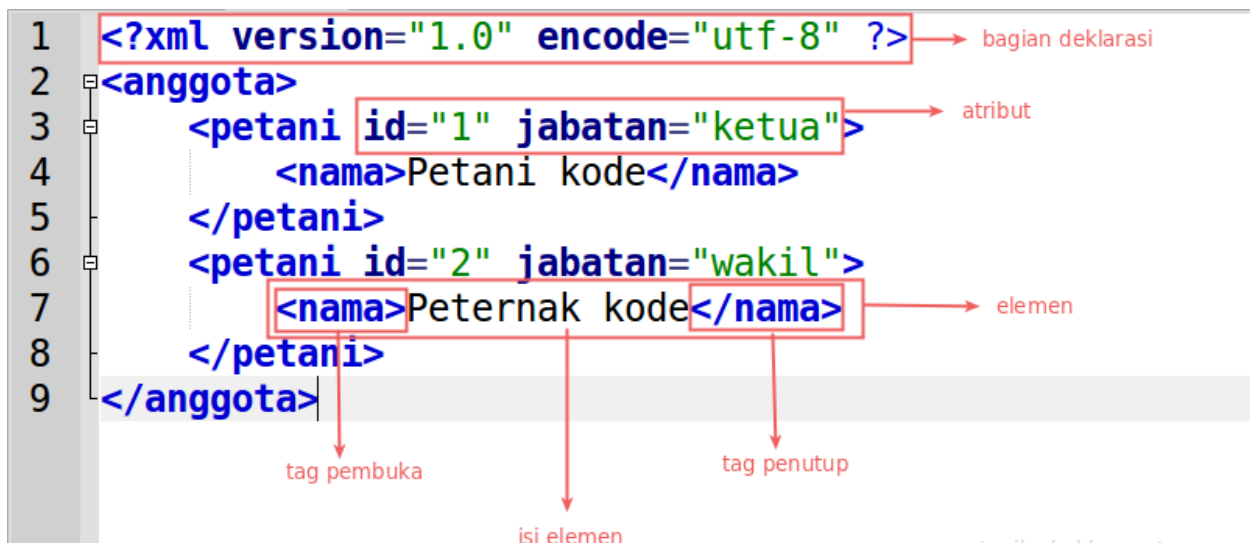
PENGERTIAN XML

Menurut W3Schools:

XML (eXtensible Markup Language) adalah sebuah bahasa markup seperti HTML yang didesain untuk menyimpan dan mengantarkan data. Ada beberapa contoh penerapan XML dalam pemrograman yang pernah di temukan:

- XML digunakan untuk membawa data pada API seperti SOAP;
- XML digunakan untuk membentuk struktur form pada pemrograman desktop seperti delphi, lazarus, vb, gtk, dll;
- XML digunakan untuk membuat layout aplikasi android;
- XML bertugas untuk membentuk struktur proyek pada java yang menggunakan Maven;
- XML digunakan sebagai format file SVG, DOCX, dsb.
- dan masih banyak lagi contoh penerapan XML.

STRUKTUR XML



- Deklarasi: Adalah bagian penting dalam XML, biasanya digunakan untuk menentukan versi XML yang akan digunakan.
- Elemen: berisi tag-tag yang mendefinisikan sebuah data objek.
- Atribut: berisi keterangan tambahan dari objek.

OBJEK DI XML

Objek adalah jenis data yang menyimpan properti dan method. Pada XML, kita tidak bisa menyimpan method. Karena itu, objek dalam XML adalah elemen yang hanya menyimpan properti saja.

Contoh Objek:

```
<mahasiswa nama="Dian" nim="12345678" semester="IV" />
```

ARRAY DI XML

Array adalah jenis data yang berisi kumpulan dari beberapa data. Array biasanya memiliki indeks untuk mengakses data di dalamnya. Array pada XML dapat kita buat dengan cara membuat elemen dengan nama yang sama.

```
<members>
  <user>Yafi</user>
  <user>Telaga</user>
  <user>Edi</user>
  <user>Agus</user>
</members>
```

PENGERTIAN JSON

JSON (JavaScript Object Notation) adalah sebuah format data yang digunakan untuk pertukaran dan penyimpanan data. Kata kunci yang perlu diingat: “pertukaran & penyimpanan data”. JSON merupakan bagian (subset) dari Javascript. JSON bisa dibaca dengan berbagai macam bahasa pemrograman seperti C, C++, C#, Java, Javascript Perl, Python, dan banyak lagi. Hal ini membuat JSON menjadi bahasa yang ideal untuk pertukaran data antar aplikasi. JSON bahkan mendominasi pendahulunya si XML karena JSON lebih sederhana dan mudah dibaca.

```
biodata.xml
1 <?xml version="1.0"?>
2 <biodata>
3   <name>Dian</name>
4   <gender>Male</gender>
5   <age>24</age>
6   <socialMedia>
7     <github>ardianta</github>
8     <twitter>@ardiantapargo</twitter>
9   </socialMedia>
10 </biodata>

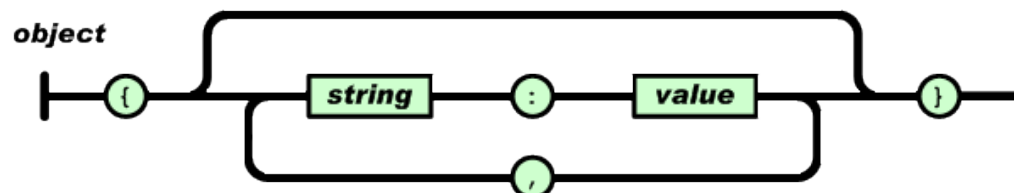
biodata.json
1 {
2   "biodata": {
3     "name": "Dian",
4     "gender": "Male",
5     "age": 24,
6     "socialMedia": {
7       "github": "ardianta",
8       "twitter": "@ardiantapargo"
9     }
10   }
11 }
```

STRUKTUR JSON



OBJECT

JSON selalu dimulai dengan tanda kurung kurawal { dan ditutup dengan kurung }. Lalu di dalam kurung kurawal, berisi data yang format key dan value. Jika terdapat lebih dari satu data, maka dipisah dengan tanda koma dan di data terakhir tidak diberikan koma.

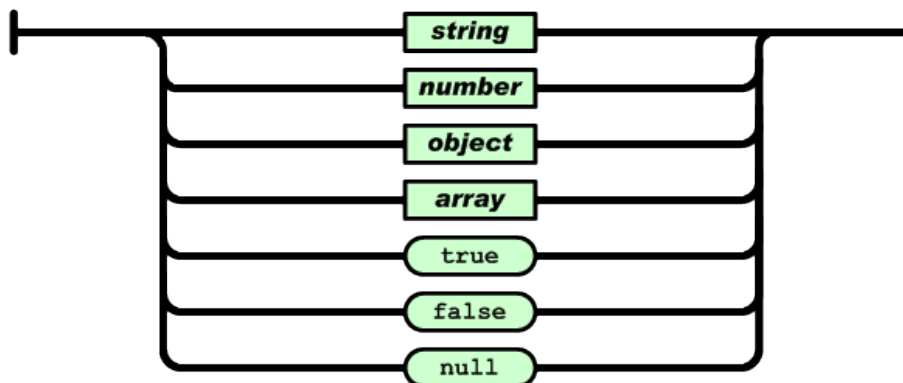


```
{
  "no": 1,
  "name": "Edi",
  "status": "Mahasiswa",
  "socialmedia": {
    "facebook": "edikurniawan",
    "twitter": "edikurniawan",
    "instagram": "edikurniawan",
    "youtube": "edikurniawan",
    "github": "edikurniawan"
  }
}
```

VALUE

Untuk value kita bisa memberikan tipe data apa pun. Bahkan juga bisa kita isi dengan array dan objek. Berikut ini tipe data yang didukung oleh JSON.

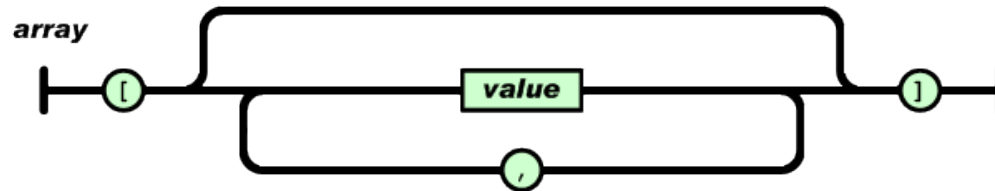
value



```
{
  "no": 1,
  "name": "Edi",
  "status": "Mahasiswa",
  "socialmedia": {
    "facebook": "edikurniawan",
    "twitter": "edikurniawan",
    "instagram": "edikurniawan",
    "youtube": "edikurniawan",
    "github": "edikurniawan"
  }
}
```


ARRAY

Untuk array dibuat dengan tanda kurung siku [...]



```
1  {  
2    "name": "Edi",  
3    "hobbies": ["Coding", "Blogging", "Drawing"]  
4  }
```

CARA SEDERHANA PRODUKSI JSON

Setiap bahasa pemrograman memiliki cara yang berbeda-beda untuk memproduksi (membuat) dan mengonsumsi data JSON. Pada Javascript, kita bisa gunakan fungsi `JSON.stringify()` untuk membuat JSON dari objek Javascript.

Contoh:

```
var person = {  
  name: "Edi",  
  age: 21,  
  status: "Mahasiswa"  
}  
  
var jsonString = JSON.stringify(person);  
console.log (jsonString);  
{"name":"Edi","age":21,"status":"Mahasiswa"}
```

TUGAS

Buat data diri lengkap kalian menjadi kedalam bentuk xml dan json, berupa nama, saudara, orang tua, hobby, alamat, social media dll.

Contoh

```
1  <?xml version="1.0"?>
2  <dataDiri>
3    <nama>
4      <awal>Edi</awal>
5      <akhir>Kurniawan</akhir>
6    </nama>
7
8    <jenisKelamin>Laki-laki</jenisKelamin>
9
10   <alamat>
11     <desa>Korpri Jaya</desa>
12     <kecamatan>Sukarame</kecamatan>
13     <kabupaten>bandar Lampung</kabupaten>
14   </alamat>
15
16   <sosialMedia>
17     <facebook>Edi Kurniawan</facebook>
18     <instagram>edi.kurniawan27</instagram>
19   </sosialMedia>
20 </dataDiri>
```