
JavaScript

Hands-on Labs 7

I Made Arsa Suyadnya
Program Studi Teknik Elektro, FT - UNUD

A. Pengenalan JavaScript

JavaScript adalah bahasa script yang dapat berjalan pada sisi client (browser) dan termasuk dalam kelompok *client side scripting*. Pada awalnya, JavaScript mulai diperkenalkan di browser Netscape Navigator 2. Namun pada waktu itu namanya bukan JavaScript, melainkan LiveScript. Mengingat saat itu teknologi Java sedang tren, maka pihak Netscape memutuskan untuk mengganti LiveScript menjadi JavaScript.

JavaScript bukanlah bahasa Java dan merupakan dua bahasa yang berbeda. JavaScript diinterpretasikan oleh klien (kodenya bisa dilihat pada sisi klien), sedangkan kode Java dikompilasi oleh pemrogram dan hasil kompilasinyalah yang dijalankan oleh klien.

Alasan utama memilih JavaScript adalah karena faktor penggunaan dan ketersediaan. Maksudnya adalah kebanyakan web browser sudah mendukung bahasa ini, sebut saja IE, Netscape, Firefox, Opera, Chrome dan lain-lain. Sehingga dapat diasumsikan bahwa kebanyakan user atau pengguna membuka halaman web dengan web browser yang telah mendukung JavaScript. Secara umum, penggunaan JavaScript adalah untuk menciptakan keinteraktifan halaman web dengan user, misalnya untuk validasi input data dalam form yang dilakukan user, memberikan efek mouse over pada suatu image atau link, dan lainnya.

B. Penulisan JavaScript

Ada dua jenis cara untuk menuliskan kode JavaScript, yakni pertama kode JavaScript ditulis dalam file yang terpisah dengan HTML, dan kedua kode JavaScript ditulis langsung dalam dokumen HTML.

Kode JavaScript yang ditulis di luar HTML disebut Eksternal JavaScript dengan ekstensi file .js. Dalam HTML, penulisan script diawali dengan `<script> ... </script>`. Script yang akan dijalankan harus diletakkan diantara `<script>` dan `</script>`. Tag `<script>` memiliki beberapa atribut, namun yang terpenting adalah atribut `language` dan `type`. Karena JavaScript bukan satu-satunya bahasa scripting, maka sangatlah perlu untuk memberitahukan kepada browser bahwa bahasa script yang digunakan adalah JavaScript dan selanjutnya browser akan menjalankan modul pendukung Javascript untuk memprosesnya. Sehingga untuk Javascript, pada tag `<script>` perlulah ditambahkan atribut berikut ini:

```
<script language="JavaScript" type="text/javascript">
```

Script dapat diletakkan di tag `<body>` dan atau di tag `<head>` pada bagian halaman HTML.

Contoh 1

```
<script language="JavaScript" type="text/javascript">
alert("Belajar Javascript");
</script>
```

Pada contoh 2 berikut, script di tulis pada bagian `<body>`.

Contoh 2

```
<!DOCTYPE html>
<html>
<body>
...
<script language="JavaScript" type="text/javascript">
document.write("<h1> Belajar Javascript</h1>");
</script>
```

```
...
</body>
</html>
```

Untuk penulisan eksternal JavaScript adalah sebagai berikut.

```
<script type="text/javascript" src="namafile.js"></script>
```

Komentar pada suatu script tidak akan dieksekusi oleh JavaScript. Komentar ditambahkan pada script untuk memberikan penjelasan script atau membuat script tidak dieksekusi. Untuk memberikan komentar yang hanya satu baris gunakan `//`.

Berikut contoh memberi komentar satu baris.

Contoh 3

```
// Write to a heading:
document.getElementById("myH1").innerHTML="Selamat datang di Homepage Saya";
// Write to a paragraph:
document.getElementById("myP").innerHTML="Ini adalah paragraph saya.";
```

Contoh 4

```
var x=5; // mendeklarasikan x dan memasangkannya dengan 5
var y=x+2; // mendeklarasikan y dan memasangkannya dengan x+2
```

Untuk memberikan komentar lebih dari satu baris, gunakan `/*` dan `*/`. Jadi script yang terletak di antara `/*` dan `*/` akan dianggap sebagai komentar. Berikut contoh memberi komentar yang lebih dari satu baris.

Contoh 5

```
/*
Kode berikut akan ditulis ke heading dan paragraph.
Dan akan menampilkan halaman homepage
*/
document.getElementById("myH1").innerHTML="Selamat datang di Homepage Saya";
document.getElementById("myP").innerHTML=" Ini adalah paragraph saya.";
```

C. Tipe Data dan Variabel

Seperti halnya di bahasa pemrograman yang lain, dalam JavaScript juga dikenal dengan beberapa nama tipe data dan variabel. Penggunaan variabel terkait dengan penyimpanan value atau data dalam memori komputer yang senantiasa dapat di-load. Sedangkan tipe data terkait dengan jenis data atau value yang disimpan dalam variabel.

1. Deklarasi Variabel

Dalam JavaScript, setiap kali akan menggunakan variabel, langkah pertama yang harus dilakukan adalah mendeklarasikan keberadaan nama variabel. Hal ini perlu dilakukan karena dengan adanya deklarasi nama variabel, komputer akan menyediakan beberapa bagian memori untuk menyimpan nilai pada nama variabel tersebut. Untuk mendeklarasikan variabel digunakan kata **var**. Dalam mendeklarasikan nama variabel ada beberapa aturan yang harus diperhatikan, yaitu :

1. Nama variabel harus dimulai dengan huruf
2. Nama variabel juga dapat dimulai dengan \$ dan _
3. Nama variabel adalah *casesensitie* (memperhatikan besar kecilnya huruf)
4. Jangan gunakan *Reserved Word* atau kata tercadang sebagai nama variabel. Kata tercadang adalah kata yang sudah built in dalam Javascript yang sudah bermakna khusus. Pelanggaran hal ini akan mengakibatkan *error*.

Contoh 1

```
var namakota;
namakota="Malang";
```

Contoh 2

```
var namakota="Malang";
```

Contoh 3

```
var namakota="malang", propinsi="Jawa Timur", kode="3", x=6.23;
```

Pada contoh 1 di atas, dideklarasikan variabel `namakota`. Pada awal pendeklarasian, nilai dari variabel `namakota` adalah `null` (kosong). Kemudian variabel `namakota` diberi nilai `Malang`. Untuk memberikan nilai pada suatu variabel digunakan tanda petik dua (") apabila tie datanya berupa string. Pada contoh 2 mendeklarasikan variabel `namakota` yang sekaligus memberi nilai pada variabel `namakota`. Contoh 3 di atas mendeklarasikan beberapa variabel sekaligus. Untuk mendeklarasikan beberapa variabel digunakan tanda koma (,) untuk memisahkan variabel satu dengan yang lainnya.

2. Tipe Data

Tipe data pada JavaScript meliputi : String, Integer, Float, Array, Object dan Booleans. Tipe data string adalah data yang memuat karakter, misalnya "Malang". String adalah sebarang text yang ada di dalam tanda petik, baik petik ganda (") maupun petik tunggal ('). Tipe data integer dan float merupakan tipe data numerik. Dalam mendeklarasikan tipe data object digunakan tanda kurung kurawal ({ ... }). Setiap properti dalam tipe data object dipisahkan dengan menggunakan tanda koma (,). Tipe data Booleans terdiri dari dua nilai, yaitu `true` atau `false`. Berikut beberapa contoh penggunaan tipe data pada JavaScript.

Contoh 4

```
var namakota="Malang";      // tipe data string
var propinsi='Jawa Timur'; // tipe data string
var x1=34;                 // tipe data integer
var x2=3.14;               // tipe data float
var y=123e4;               // tipe data integer
var x=true;                // tipe data boolean
```

Contoh 5

```
//berikut beberapa cara mendeklarasikan Array
var mobil=new Array();
mobil[0]="Toyota";
mobil[1]="Daihatsu";
mobil[2]="Honda";
var bulan=new Array("Januari","Febuari","Maret","April");
var kampus=["UGM", "ITB", "UNUD","UNSOED","UNRAM"];
```

Contoh 6

```
var klien={nama:"Pamungkas", sex:"Laki-Laki", id:"5758"};
```

Pada contoh sebelumnya dideklarasikan beberapa variabel dengan berbagai tipe data. Pada contoh 4 dideklarasikan variabel dengan tipe data string, integer, float dan Boolean. Contoh 5 mendeklarasikan tipe data jenis array, dan beberapa cara variasi penulisannya. Pendeklarasian variabel dengan tipe data object dicontohkan pada contoh 6.

D. Operator pada Javascript**1. Operator Aritmetika**

Operator aritmetika digunakan untuk melakukan operasi aritmatika antara variabel dan atau nilai. Misal diberikan $y = 5$, Tabel 1 berikut menjelaskan tentang operator aritmetika.

Tabel 1. Operator Aritmetika pada JavaScript

| Operator | Deskripsi | Contoh | Nilai x | Nilai y |
|----------|---------------------|----------|---------|---------|
| + | Penjumlahan | $x=y+2$ | 7 | 5 |
| - | Pengurangan | $x=y-2$ | 3 | 5 |
| * | Perkalian | $x=y*2$ | 10 | 5 |
| / | Pembagian | $x=y/2$ | 2.5 | 5 |
| % | Modulus (Sisa bagi) | $x=y\%2$ | 1 | 5 |
| ++ | Increment | $x=++y$ | 6 | 6 |
| | | $x=y++$ | 5 | 6 |
| -- | Decrement | $x=--y$ | 4 | 4 |
| | | $x=y--$ | 5 | 4 |

2. Operator Pemberian Nilai (assignment)

Operator pemberian nilai digunakan untuk memberikan nilai pada variabel. Dalam contoh berikut, diberikan $x = 10$, dan $y = 5$. Tabel 2 berikut menjelaskan operator pemberian nilai.

Tabel 2. Operator Pemberian Nilai (Assignment) pada JavaScript

| Operator | Contoh | Sama dengan | Hasil |
|----------|---------|-------------|--------|
| = | $x=y$ | | $x=5$ |
| += | $x+=y$ | $x=x+y$ | $x=15$ |
| -= | $x-=y$ | $x=x-y$ | $x=5$ |
| *= | $x*=y$ | $x=x*y$ | $x=50$ |
| /= | $x/=y$ | $x=x/y$ | $x=2$ |
| %= | $x\%=y$ | $x=x\%y$ | $x=0$ |

3. Operator + yang digunakan pada tipe data String

Operator + juga dapat digunakan untuk menambahkan variabel bertipe data string atau nilai text.

Contoh 1

```
//untuk menambahkan dua atau lebih variabel tipe String, gunakan operator +
txt1="Selamat Datang";

txt2="Di Jurusan Matematika"; txt3=txt1+" "+txt2;
```

Hasil dari script di atas adalah : Selamat Datang Di Jurusan Matematika

Contoh 2

```
//menambahkan String dan Bilangan
x=5+5;
y="5"+5;
z="angka "+5;
```

Hasil dari script di atas adalah :

```
10
55
angka 5
```

4. Operator Pembandingan

Operator pembandingan digunakan dalam pernyataan logika untuk menentukan kesamaan atau perbedaan diantara nilai-nilai. Diberikan nilai X = 5, Tabel 3 di bawah ini menjelaskan operator pembandingan.

Tabel 1. Operator Pembandingan pada JavaScript

| Operator | Deskripsi | Pembandingan | Balikan |
|----------|---|--------------|---------|
| == | Sama dengan | x==8 | False |
| | | x==5 | True |
| === | Sama persis dengan (nilai dan tipe data) | x===5 | False |
| | | x===5 | True |
| != | Tidak sama dengan | x!=8 | True |
| !== | Tidak sama dengan (nilai atau tipe data) | x!==5 | True |
| | | x!=5 | False |
| > | Lebih besar dari | x>8 | False |
| < | Lebih kecil dari | x<8 | True |
| ≥ | Lebih besar sama | x>=8 | False |
| ≤ | Lebih kecil sama dengan | x<=8 | True |

5. Operator Logika

Operator logika digunakan untuk menunjukkan nilai kebenaran antara beberapa variabel atau beberapa nilai. Diberikan nilai x = 6 dan y = 3, Tabel 4 di bawah menjelaskan operator logika.

Tabel 2. Operator Logika pada JavaScript

| Operator | Deskripsi | Pembandingan | Balikan |
|----------|-----------------|--------------------|---------|
| && | Dan | (x < 10 && y > 1) | True |
| | Atau | (x == 5 y == 5) | False |
| ! | Negasi/Ingkaran | !(x==y) | True |

6. Operator Bersyarat

Javascript juga memuat operator bersyarat yang memberikan suatu nilai ke suatu variabel berdasarkan kondisi yang sama.

Sintaksnya:

```
NamaVariabel=(syarat)?nilai 1: nilai 2
```

E. Statement Kondisional dan Perulangan

1. Kondisional

Kondisional berguna untuk mengecek suatu kondisi dan melakukan suatu kode jika kondisi tersebut benar atau salah.

- **if**

Sintaks :

```
if(kondisi){
    kode yang dijalankan jika benar
}
```

Contoh:

```
<script> var x = 10;
if(x == 10){
    alert("Hai Apa kabar");
}
</script>
```

- **if - else**

Sintaks :

```
if(kondisi){
    kode yang dijalankan jika benar
}else{
    kode yang dijalankan jika salah
}
```

Contoh :

```
<script> var x = 5;
if(x == 10){
```

```

    alert("Hai Apa kabar");
}else{
    alert("x tidak sama dengan 10");
}
</script>

```

▪ if - else if - else

Sintaks :

```

if(kondisi 1){
    kode yang dijalankan jika kondisi 1 benar
}else if(kondisi 2){
    kode yang dijalankan jika kondisi 2 benar
}else if(kondisi 3){
    kode yang dijalankan jika kondisi 3 benar
}else{
    kode jika salah satu kondisi di atas tidak ada yang benar
}

```

Contoh :

```

<script>
var nilai = 80;
if(nilai >= 85){
    alert("A");
}else if(nilai >= 70 && nilai < 85){
    alert("B");
}else if(nilai >= 60 && nilai < 70){
    alert("C");
}else{
    alert("D");
}
</script>

```

▪ switch

Sama seperti if - else if - else, berguna jika membutuhkan kondisi yang banyak.

Sintaks :

```

switch(ekspresi){
    case kondisi1 :
        kode yang dijalankan jika kondisi1 benar;
        break;
    case kondisi2 :
        kode yang dijalankan jika kondisi2 benar;
        break;
    case kondisi3 :
        kode yang dijalankan jika kondisi3 benar;
        break;
}

```


Contoh:

```
<script>
  var buah = "mangga";
  switch (buah) {
    case "apple":
      alert("buahnya adalah apple");
      break;
    case "mangga":
      alert("buahnya adalah mangga");
      break;
    case "jambu":
      alert("buahnya adalah jambu"); break;
  }
</script>
```

2. Perulangan

Perulangan adalah suatu blok perintah atau kode yang diulang selama syarat bernilai TRUE. Dalam Javascript, perulangan ini dapat dinyatakan dalam 2 bentuk, yaitu dengan **for** dan **while**.

- **for**
Berguna untuk perulangan dengan nilai awalan tertentu, dan dilakukan hingga kondisi tertentu.

Sintaks :

```
for(awal; kondisi; increment/decrement)
{
  kode untuk dijalankan
}
```

Contoh :

```
<script>
for (i = 1; i <= 10; i++) {
  document.write(i);
}
</script>
```

- **while**
Berguna untuk menjalankan suatu kode terus menerus selama kondisi bernilai TRUE.

Sintaks :

```
while(kondisi){
  kode untuk dijalankan;
}
```

Contoh :

```
<script> var i=1; while(i<=5)
{
  document.write("Nomor : "+i +"<br />");
```

```

    i++;
  }
</script>

```

F. Fungsi

Function adalah bagian subprogram yang melaksanakan suatu tugas tertentu. Setiap saat ketika diperlukan, function ini dapat dipanggil dengan cara menuliskan nama functionnya saja, disertai dengan parameter apabila ada.

Sintaks:

```

function nama_fungsi(parameter){
    kode-kode javascript
}

```

Contoh 1:

```

<script> function tes(){
    document.write("Hello World!");
}

//untuk menjalankan fungsi, cukup tulis namafungsi
tes();
</script>

```

Contoh 2:

```

<script>
function kalikan(x,y){
    z = x * y;
    alert("Hasil kali "+x+" * "+y+" = "+z);
}

kalikan(5,3);
</script>

```

G. Penanganan Event

Penanganan Event atau Event Handler adalah kemampuan JavaScript untuk mendeteksi event atau kejadian-kejadian yang terjadi di halaman web, kemudian menangani atau melakukan suatu proses jika terdeteksi suatu event. Event di web bisa macam-macam, seperti klik, double klik, menggerakkan mouse, bila pointer mouse berada di atas suatu objek HTML, dan sebagainya.

Sintaknya:

```

nama_event="kode javascript"

```

Contoh:

```

<html>
<body>
<a href="http://www.google.com" onclick="alert('hello')">google</a>
</body>

```

```
</html>
```

Pada contoh di atas ada sebuah link google, yang jika diklik (onclick) maka akan dieksekusi kode JavaScript `alert('hello')`.

Ada macam-macam event yang bisa terjadi pada halaman web atau objek HTML, yaitu sebagai berikut:

- onblur
- onchange
- onclick
- ondblclick
- onerror
- onfocus
- onkeydown
- onkeypress
- onkeyup
- onload
- onmousedown
- onmousemove
- onmouseout
- onmouseover
- onmouseup
- onreset
- onresize
- onselect
- onsubmit
- onunload

Berikut ini beberapa contoh penanganan event menggunakan JavaScript.

1. onclick

Pada contoh berikut ketika user mengklik elemen/komponen button akan menampilkan window alert.

```
<script>
function inform(){
    alert("Hai anda mengklik saya");
}
</script>

<form>
<input type="button" name="test" value="Click me" onclick="inform()">
</form>
```

Pada contoh berikut ketika user mengklik elemen/komponen radio button akan mengubah warna latar halaman web.

```
<html>
<body>
<form name="go">
```

```

<input type="radio" name="C1" onclick="document.bgColor='lightblue'">
<input type="radio" name="C1" onclick="document.bgColor='lightyellow'">
<input type="radio" name="C1" onclick="document.bgColor='lightgreen'">
</form>
</body>
</html>

```

2. onload

Event onload akan dieksekusi jika suatu objek telah di-load, pada contoh berikut menempatkan event onload pada tag <body>, artinya jika halaman web sudah di-load semua, maka dieksekusi kode JavaScript.

```

<html>
<head><title>Body onload example</title>
</head>
<body onload="alert('Halaman ini telah selesai di loading')">
Welcome to my page
</body>
</html>

```

3. onmouseover dan onmouseout

Onmouseover berguna untuk mendeteksi apakah pointer mouse berada di atas suatu objek HTML, onmouseout berguna untuk mendeteksi apakah pointer mouse keluar dari objek HTML, contoh:

```

<html>
<body>
<table>
<tr onmouseover="this.bgColor='lightblue'"
onmouseout="this.bgColor='#efefef'" bgcolor="#efefef">
<td>Baris pertama</td>
</tr>
<tr>
<td>Baris kedua</td>
</tr>
</body>
</html>

```

4. onunload

Biasanya berguna untuk mendeteksi jika user meninggalkan atau menutup suatu halaman web.

```

<html>
<body onunload="alert('Thank you. Please come back to this site and visit us soon,
ok?')">
<h1>Welcome</h1>
</body>
</html>

```

H. Object String

1. Memformat Teks dengan JavaScript

Untuk membantu melakukan formatting terhadap teks, JavaScript menyediakan beberapa metode formatting.

| Properti | Deskripsi |
|----------|--|
| length | Menghasilkan jumlah karakter dari suatu string atau teks |

| Metode | Deskripsi |
|------------------|--|
| anchor(nama) | Menghasilkan string dengan diapit tag |
| big() | Menghasilkan string dengan diapit tag <big>. |
| blink() | Menghasilkan string dengan diapit tag <blink> |
| bold() | Menghasilkan string dengan diapit tag |
| fixed() | Menghasilkan string dengan diapit tag <tt> |
| fontcolor(warna) | Menghasilkan string dengan diapit tag |
| fontsize(size) | Menghasilkan string dengan diapit tag |
| italics() | Menghasilkan string dengan diapit tag <i> |
| link(url) | Menghasilkan string dengan diapit tag |
| small() | Menghasilkan string dengan diapit tag <small> |
| strike() | Menghasilkan string dengan diapit tag <strike> |
| sub() | Menghasilkan string dengan diapit tag <sub> |
| sup() | Menghasilkan string dengan diapit tag <sup> |
| toLowerCase() | Mengubah string menjadi huruf kecil semua. |
| toUpperCase() | Mengubah string menjadu huruf besar semua |

Berikut ini beberapa contoh penggunaan metode formatting teks.

Contoh 1:

```
<script>
var teks = "denpasar";
panjang = teks.length;
```

```
alert (panjang);
</script>
```

Output:

Menampilkan alert window yang berisi panjang teks: 8

Contoh 2:

```
<script>
var message="Welcome to our site!";
document.write(message);
</script>
```

Output:

Welcome to our site!

Contoh 3:

```
<script>
var message="Welcome to our site!" document.write(message.toUpperCase())
</script>
```

Output:

WELCOME TO OUR SITE!

Contoh 4:

```
<script>
var message="Welcome to our site!";
var format=message.toUpperCase();
var size=1;
for (i=0;i<message.length;i++){
    document.write(format.charAt(i).fontsize(size).bold());
    if (size<7){
        size++
    }else{
        size=1
    }
}
```

Output:

wELCOME TO OUR **site!**

2. Penanganan Objek String

Sebelumnya telah dibahas metode string yang berkaitan dengan format teks. Berikutnya adalah metode objek string lainnya yang berfungsi untuk menangani dan memanipulasi objek string.

| Metode | Deskripsi |
|---|---|
| <code>charAt(x)</code> | Menghasilkan karakter pada posisi x dari suatu string. |
| <code>charCodeAt(x)</code> | Menghasilkan nilai Unicode value dari karakter pada posisi x dari suatu string. |
| <code>concat(teks1, teks2,...)</code> | Menggabungkan satu atau lebih string-string (teks1, teks2, dan sebagainya). |
| <code>fromCharCode(c1, c2,...)</code> | Menghasilkan string yang dibuat menggunakan urutan dari nilai unicode |
| <code>indexOf(substr, [start])</code> | Mencari dan (jika ditemukan) menghasilkan nomor index number dari karakter atau substring yang di dalam string. Jika tidak ditemukan, hasilnya -1. "Start" adalah argument opsional, posisi awal di string untuk memulai pencarian, defaultnya adalah 0 |
| <code>match(regex)</code> | Mengeksekusi suatu pencarian untuk string berdasarkan pola regular expression. Menghasilkan suatu array informasi, jika tidak ditemukan menghasilkan null. |
| <code>replace(regex, replacetext)</code> | Mencari dan menukar string yang dicari berdasarkan pola regular expression. |
| <code>search(regex)</code> | Mengetes apakah pola regular expression cocok pada suatu string, jika cocok akan menghasilkan index dari yang cocok, jika tidak ada yang cocok menghasilkan -1. |
| <code>slice(start, [end])</code> | Menghasilkan substring dari suatu string berdasarkan "start" dan "end" argument. |
| <code>split(delimiter, [limit])</code> | Memotong-motong string berdasarkan pembatas yang ditentukan, hasilnya dalam bentuk array. |
| <code>substr(start, [length])</code> | Menghasilkan karakter atau substring dari suatu string yang dimulai dari "start" sampai panjang "length" yang ditentukan. |

Berikut ini beberapa contoh penggunaan metode penanganan objek string.

1. Validasi email

Pada contoh berikut, melakukan pengecekan validasi email. Cara pengecekannya adalah memeriksa apakah string yang diinput mengandung karakter @ atau titik (.)

```
<form name="test" onSubmit="checkemail(this.test2.value);return false">
<input type="text" size=20 name="test2">
```

```

<input type="submit" value="Submit">
</form>

<script type="text/javascript">
function checkemail(email){
  if (email.indexOf("@")!=-1 && email.indexOf(".")!=-1)
    alert("bagus!");
  else
    alert("Bukan email");
}
</script>

```

2. Menghitung kata

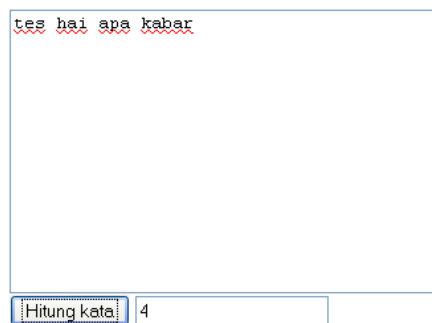
Pada contoh berikut, melakukan penghitungan kata, caranya adalah mengambil isi suatu textarea, lalu memotong-motong isinya dengan metode split. Hasilnya dapat dilihat pada Gambar 1.

```

<form name="wordcount">
<textarea rows="12" name="wordcount2" cols="38"
wrap="virtual"></textarea><br>
<input type="button" value="Hitung kata" onClick="hitung()">
<input type="text" name="wordcount3" size="20">
</form>

<script type="text/javascript">
function hitung(){
  var formcontent=document.wordcount.wordcount2.value;
  isi=formcontent.split(" ");
  document.wordcount.wordcount3.value=isi.length;
}
</script>

```



Gambar 1. Aplikasi sederhana penghitung kata dengan JavaScript

3. Parse String ke Integer dan Float

parseInt, untuk mengubah string menjadi integer dan parseFloat, untuk mengubah string menjadi float.

Contoh :

```

<script type="text/javascript">

```



```

document.write(parseInt("10") + "<br />");
document.write(parseInt("10.00") + "<br />");
document.write(parseInt("10.33") + "<br />");
document.write(parseInt("34 45 66") + "<br />");
document.write(parseInt(" 60 ") + "<br />");
document.write(parseInt("40 years") + "<br />");
document.write(parseInt("He was 40") + "<br />");

document.write("<br />");

document.write(parseInt("10")+ "<br />");
document.write(parseInt("10",10)+ "<br />");

document.write(parseInt("010")+ "<br />");
document.write(parseInt("10",8)+ "<br />");

document.write(parseInt("0x10")+ "<br />");
document.write(parseInt("10",16)+ "<br />");
</script>

```

I. Object Window

1. Membuka Window di JavaScript

Untuk membuka window di JavaScript adalah dengan metode open().

Sintaks:

```
window.open(url, nama_window, konfigurasi);
```

Contoh:

```
window.open("http://www.google.com", "windowku", "width=300, height=200");
```

Untuk konfigurasi yang bisa diset sebagai berikut:

| Konfigurasi | Deskripsi | Nilai |
|-------------|--|---------------------------|
| width | Lebar window | Lebar window dalam pixel |
| height | Tinggi window | Tinggi window dalam pixel |
| toolbar | Menampilkan toolbar browser | 1 atau 0 |
| menubar | Menampilkan menubar browser | 1 atau 0 |
| scrollbars | Apakah menampilkan scrollbars | 1 atau 0 |
| resizable | Men-set apakah bisa window diubah-ubah | 1 atau 0 |

Contoh:

```
<html>
<body>
<script>
function bukawindow() {
    window.open("http://www.google.com", "google", "width=400,
height=300,toolbar=1");
}
</script>
<input type="button" onclick="bukawindow()" value="buka">
</body>
</html>
```

2. Mereload, Menutup, Meloding Halaman Baru, Print

Untuk mereload window adalah dengan cara:

```
window.location.reload()
```

Untuk menutup window adalah dengan cara:

```
window.close()
```

Untuk meloding halaman baru adalah:

```
window.location="urlkamu.com"
```

Kode di atas akan sama jika anda mengklik suatu link:

```
<a href="urlkamu.com">urlkamu</a>
```

Untuk mengeprint halaman web

```
window.print()
```

3. Komunikasi Antar Window

Adakalanya ketika membuka window baru, perlu melakukan komunikasi data ke window parent-nya, atau bisa mengakses dan memanipulasi objek di window parent.

Untuk mengakses window parent gunakan window.opener

Contoh:

induk.html

```
<html>
<head>
<title>Test</title>
<script language="javascript">
function buka() {
    var x = window.open('anak.html', 'newWindow', 'height=300,width=300');
}
```

```

</script>
</head>
<body>
<form name=formulir>
<INPUT type=button value="buka window" name=submit1 onClick="buka();">
<input type=text name=kotak id=kotak>
</form>
</body>
</html>

```

anak.html

```

<html>
<head>
<title>testing</title>
<script language=javascript>
function setInduk() {
    window.opener.document.getElementById("kotak").value =
document.getElementById("inputan").value;
}
</script>
</head>
<body>
<form name=myForm>
<input type=text name="inputan" id="inputan">
<input type=button value="Click Me" onClick="setInduk();">
</form>
</body>
</html>

```

4. Alert, Confirm, dan Prompt

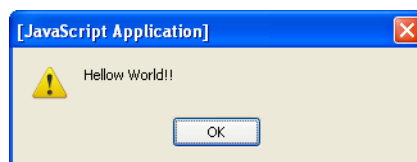
- **alert** : digunakan untuk menampilkan window alert
Contoh:

```

<script>
alert("Hellow World!!");
</script>

```

Hasil:



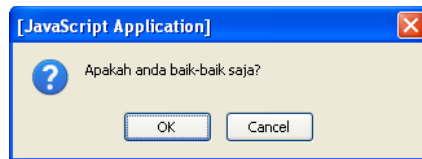
Gambar 2. Window alert dengan JavaScript

- **confirm**, digunakan untuk memunculkan window konfirmasi atau semacam pertanyaan yang jawabannya antara OK atau Cancel

Contoh:

```
<script type="text/javascript">
var x=window.confirm("Apakah anda baik-baik saja?");
if (x)
    window.alert("Good!");
else
    window.alert("Too bad");
</script>
```

Hasil:



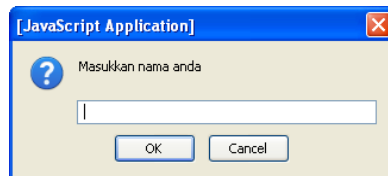
Gambar 3. Window confirm dengan JavaScript

- **prompt**, digunakan untuk meminta inputan melalui window

Contoh:

```
<script type="text/javascript">
var y=window.prompt("Masukkan nama anda");
window.alert(y);
</script>
```

Hasil:



Gambar 4. Window prompt dengan JavaScript

J. Dynamic HTML

1. Mengakses dan Manipulasi Objek HTML

Untuk mengakses objek HTML, dapat menggunakan `document.getElementById("id_objek")` atau `document.getElementsByName("nama_objek")`.

Contoh:

```
<html>
<head>
<script type="text/javascript">
function getElement(){
    var x=document.getElementById("myHeader") alert("Ini adalah elemen " +
x.tagName);
```

```

}
</script>
</head>
<body>
<h1 id="myHeader" onclick="getElement()">
Klik saya untuk lihat nama tag saya</h1>
</body>
</html>

```

Berikut metode-metode untuk memanipulasi objek HTML.

- **innerHTML:** untuk mendapatkan isi teks atau html suatu objek atau tag HTML

Contoh:

```

<html>
<head>
<script type="text/javascript">
function lihatisi(){
    isi = document.getElementById("kepala").innerHTML;
    alert(isi);
}

function tukarisi(){
    document.getElementById("kepala").innerHTML = "Tulisan Ini diganti";
}
</script>
</head>
<body>
<h1 id="kepala">Selamat Datang user!!</h1>
<input type="button" onclick="lihatisi()" value="Lihat isi H1">
<input type="button" onclick="tukarisi()" value="Tukar isi H1">
</body>
</html>

```

- **value:** untuk mengambil atau memanipulasi nilai suatu input form.

Contoh:

```

<html>
<head>
<script type="text/javascript">
function lihatisi(){
    isi = document.getElementById("teks").value;
    alert(isi);
}

function tukarisi(){
    document.getElementById("teks").value = "Tulisan Ini diganti";
}
</script>
</head>
<body>
Ketik tulisan di form di bawah:<br>
<input type="text" name="teks" id="teks">
<input type="button" onclick="lihatisi()" value="Lihat">

```

```
<input type="button" onclick="tukarisi()" value="Tukar">
</body>
</html>
```

2. Memanipulasi Style atau CSS Objek HTML

Untuk melakukan manipulasi style atau CSS suatu objek dengan JavaScript, dapat dilakukan dengan cara:

```
objekHTML.style.namaStyle = "style yang diset"
objekHTML.namaStyle = "style yang diset"
```

atau

```
document.getElementById("namaid").style.namaStyle = "style yang diset"
document.getElementById("namaid").namaStyle = "style yang diset"
```

▪ Mengubah warna latar td

Contoh:

```
<html>
<head>
</head>
<body>
<table border=1 width=500 bgcolor="yellow">
<tr>
<td onmouseover="this.bgColor='red'"
onmouseout="this.bgColor='yellow'">Supriyanto</td>
</tr>
<tr>
<td onmouseover="this.bgColor='red'" onmouseout="this.bgColor='yellow'">Deni
Hariyadi</td>
</tr>
</body>
</html>
```

▪ Mengubah warna latar halaman web

Contoh:

```
<html>
<head>
<script>
function ubah(){
    document.bgColor = "lightblue";
}
</script>
</head>
<body>
<input type="button" onclick="ubah()" value="ubah">
</body>
</html>
```

- **Mengubah posisi objek**

Contoh:

```
<html>
<head>
<script>
function ubah() {
    document.getElementById("kotak").style.top = 300;
    document.getElementById("kotak").style.left = 300;
}
</script>
</head>
<body>
<input type="button" onclick="ubah()" value="ubah">
<div id="kotak" style="position:absolute;top:50;left:50;background-
color:yellow;width:50;height:50">
</div>
</body>
</html>
```

K. Penanganan Form

1. Penanganan CheckBox

- Mendeteksi apakah checkbox sudah di-check

Sintaksnya:

```
checkboxObject.checked=true|false
```

Contoh:

```
<html>
<head>
<script type="text/javascript">
function check() {
    document.getElementById("myCheck").checked=true
}

function uncheck() {
    document.getElementById("myCheck").checked=false
}
</script>
</head>
<body>
<form>
<input type="checkbox" id="myCheck" />
<input type="button" onclick="check()" value="Check Checkbox" />
<input type="button" onclick="uncheck()"
value="Uncheck Checkbox" />
</form>
</body>
</html>
```

- Mengambil nilai checkbox

Contoh:

```
<html>
<head>
<script type="text/javascript">
function createOrder(){
  coffee=document.forms[0].coffee;
  txt="";
  for (i=0;i<coffee.length;++ i){
    if (coffee[i].checked){
      txt=txt + coffee[i].value + " "; }
  }
  document.getElementById("order").value="You ordered a coffee with " + txt;
}
</script>
</head>
<body>
<p>How would you like your coffee?</p>
<form>
<input type="checkbox" name="coffee" value="cream">With cream<br />
<input type="checkbox" name="coffee" value="sugar">With sugar<br />
<br />
<input type="button" onclick="createOrder()" value="Send order">
<br /><br />
<input type="text" id="order" size="50">
</form>
</body>
</html>
```

2. Penanganan Input Radio

Contoh:

```
<html>
<head>
<script type="text/javascript">
function setkelamin(jenis){
  document.getElementById("kelaminmu").value = jenis;
}
</script>
</head>
<body>
jenis Kelamin:
<br>
<input type="radio" value="Laki-laki" name="kelamin"
onclick="setkelamin(this.value)">Laki-laki<br>
<input type="radio" value="Perempuan" name="kelamin"
onclick="setkelamin(this.value)">Perempuan<br>
<input type="text" id="kelaminmu">
</body>
</html>
```


3. Penanganan Select Box

- Mengambil nilai select box

Contoh:

```
<html>
<head>
<script type="text/javascript">
function favBrowser(){
    var mylist=document.getElementById("myList");
    document.getElementById("favorite").value=mylist.options[mylist.selectedIndex].
text;
}
</script>
</head>
<body>
<form>
Select your favorite browser:
<select id="myList" onchange="favBrowser()">
<option>Internet Explorer</option>
<option>Netscape</option>
<option>Opera</option>
</select>
<p>Your favorite browser is: <input type="text" id="favorite" size="20"></p>
</form>
</body>
</html>
```

- Menghapus option dari dropdown list

Contoh:

```
<html>
<head>
<script type="text/javascript">
function removeOption(){
    var x=document.getElementById("mySelect");    x.remove(x.selectedIndex);
}
</script>
</head>
<body>
<form>
<select id="mySelect">
<option>Apple</option>
<option>Pear</option>
<option>Banana</option>
<option>Orange</option>
<option>Melon</option>
</select>
<input type="button" onclick="removeOption()" value="Remove option">
</form>
</body>
</html>
```

Latihan:

1. Cobalah kode HTML dan JavaScript yang ada dalam modul ini, kemudian perhatikan dan pahami maksud dari kode HTML dan JavaScript tersebut.

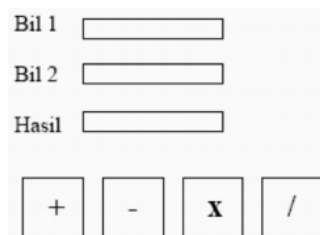
Tugas:

1. Buatlah halaman html (dengan form) untuk mengkonversi nilai angka menjadi nilai huruf dengan menggunakan JavaScript. Berikut aturan konversi yang harus dipenuhi.

Konversi:

0-40 = E
 41-55 = D
 56-60 = C
 61-65 = BC
 66-70 = B
 71-80 = AB
 81-100 = A

2. Buatlah halaman html untuk menampilkan aplikasi program kalkulator sederhana dengan menggunakan JavaScript. Contoh tampilannya adalah sebagai berikut.



Bil 1 dan Bil 2 diinputkan melalui text box, diisi dengan angka, bila tombol + atau – atau x atau / ditekan, maka akan dilakukan proses perhitungan dan hasilnya langsung ditampilkan pada text box hasil, (bilangan hasil ini merupakan operasi aritmetika sesuai dengan tombol yang ditekan).

Petunjuk:

Uploadlah tugas tersebut pada folder “modul-7” pada web hosting masing-masing. Buatlah laporan pembahasan singkat mengenai kode yang telah dibuat dalam format pdf dan laporan pembahasan tersebut diupload pula pada folder “modul-7”.