

Javascript Tutorial

Tentang JavaScript

Pada awalnya, JavaScript mulai diperkenalkan di browser Netscape Navigator 2. Namun waktu itu namanya bukan JavaScript, namun LiveScript. Mengingat pada waktu itu teknologi Java sedang panas-panasnya atau sedang tren, maka pihak Netscape memutuskan untuk mengganti namanya menjadi JavaScript, yang sepertinya nama tersebut lebih marketable dibandingkan LiveScript. Selanjutnya pihak Microsoft (rival Netscape) pun mulai ikut-ikutan memfasilitasi web browser buatannya, 'Internet Explorer', supaya bisa mendukung JavaScript. Namun mungkin karena gengsi, pihak Microsoft memberi nama bahasa yang lain, yaitu Jscript. Mulai saat itu, Netscape dan Microsoft mulai berlomba-lomba mengembangkan bahasa tersebut dalam versi yang berlainan. Oleh sebab persaingan itulah terkadang suatu JavaScript mungkin bisa bekerja dengan baik di browser Netscape, tapi tidak demikian halnya di IE, begitu pula sebaliknya.

Berikut ini tabel kompatibilitas versi-versi Javascript pada Netscape dan IE

Versi Javascript	Versi Netscape Navigator yang mendukung	Versi IE yang mendukung
JavaScript 1.0 (ekivalen dengan JScript 1.0)	2.x	3.x
JavaScript 1.1	3.x	N/A
JavaScript 1.2 (ekivalen dengan JScript 3.0)	4.0–4.05	4.x
JavaScript 1.3	4.06+	N/A
JavaScript 1.4 (ekivalen dengan JScript 5.0)		5.0
JavaScript 1.5	6.x and 7.x	5.5 and 6.0

Mengapa Memilih Javascript?

Javascript bukan merupakan satu-satunya bahasa scripting dalam dunia web, contoh bahasa yang lain adalah VBScript dan masih banyak yang lain. Jadi mengapa memilih Javascript?

Alasan utama memilih Javascript adalah karena faktor penggunaan dan ketersediaan. Maksudnya adalah kebanyakan web browser sudah mendukung bahasa ini, sebut saja IE,

Javascript Tutorial

Netscape, Firefox, Opera dan lain-lain. Sehingga dapat kita asumsikan bahwa kebanyakan orang membuka halaman web dengan web browser yang telah mendukung Javascript.

Sedangkan untuk VBScript, bahasa ini hanya bisa dijalankan pada browser IE dan harus menggunakan platform Windows.

Selain itu, Javascript juga bersifat fleksibel artinya bahwa ia tidak hanya digunakan di web browser, namun juga dapat digunakan oleh sistem operasi Windows untuk otomatisasi administration tasks. Dalam file PDF, Javascript juga digunakan untuk mengatur tampilan halaman PDF di web.

Apa yang bisa dilakukan Javascript?

Secara umum, penggunaan Javascript adalah untuk menciptakan keinteraktifan halaman web dengan user. Sebagai contoh, misalkan kita memiliki drop down menu dalam halaman web yang digunakan untuk pindah halaman (Jump Menu). Untuk membuat drop down menu, kita hanya butuh tag HTML. Namun, supaya drop down menu tersebut bisa berfungsi, untuk keperluan jump menu, kita membutuhkan Javascript.

Contoh lain, misalnya untuk validasi input data dalam form yang dilakukan user. Untuk membuat form, tentu kita hanya membutuhkan HTML. Namun untuk bisa melakukan validasi input, kita membutuhkan Javascript. Masih banyak contoh kegunaan Javascript yang lain terkait dengan keinteraktifan user.

Selain itu Javascript juga sering digunakan sebagai trik, misalnya memberikan efek mouse over pada suatu image atau link. Dengan efek ini, suatu image dapat berubah ketika mouse didekatkan ke image tersebut. Kaitannya dengan hal ini, masih banyak bentuk trik-trik lain yang dapat kita lakukan dengan Javascript.

Javascript Tutorial

Memulai Javascript

Code Javascript dapat disisipkan dalam bagian `<head> ... </head>` maupun dalam `<body> ... </body>`.

Untuk menyisipkan suatu script dalam dokumen HTML, caranya hampir sama dengan menyisipkan bagian HTML yang lain, yaitu dengan menggunakan tag untuk menandai awal dan akhir. Tag untuk menandai awal script adalah `<script>` dan diakhiri dengan `</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">
```

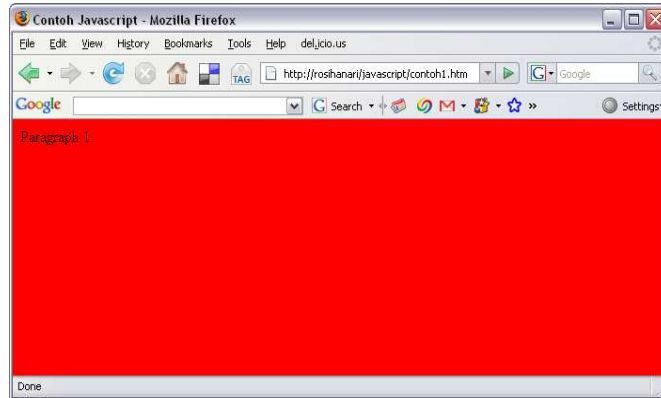
OK.. kita akan membuat contoh Javascript. Pada contoh ini, Javascript digunakan untuk memberi warna merah pada background halaman. Kita dapat membuat script ini melalui Notepad.

contoh1.htm

```
<html>
<head>
  <title>Contoh Javascript</title>
</head>
<body bgcolor="white">
  <p>Paragraph 1</p>
  <script language="JavaScript" type="text/javascript">
    document.bgColor = "red";
  </script>
</body>
</html>
```

Hasilnya adalah sebagai berikut:

Javascript Tutorial



Apabila kita perhatikan pada bagian Javascript, maka pada akhir statement diakhiri dengan tanda semi colon atau titik koma.

Untuk mengubah warna halaman web sendiri, kita menggunakan perintah

```
document.bgColor = "red";
```

Apa yang kita sebut dengan 'halaman' adalah merupakan 'document', sehingga terdapat perintah 'document' pada perintah di atas. Selanjutnya 'document' memiliki banyak properti, termasuk di dalamnya adalah 'bgcolor' yang digunakan untuk mengatur warna background halaman.

Sekarang, kita lanjutkan contoh kedua. Pada contoh kedua ini kita akan melihat flow proses dari Javascript terkait dengan adanya beberapa blok yang digunakan.

contoh2.htm

```
<html>
<head><title>Contoh Javascript</title></head>
<body bgcolor="white">
<p>Paragraph 1</p>

<script language="JavaScript" type="text/javascript">
  // blok script 1
  alert("blok script 1 sedang dijalankan");
</script>

<p>Paragraph 2</p>

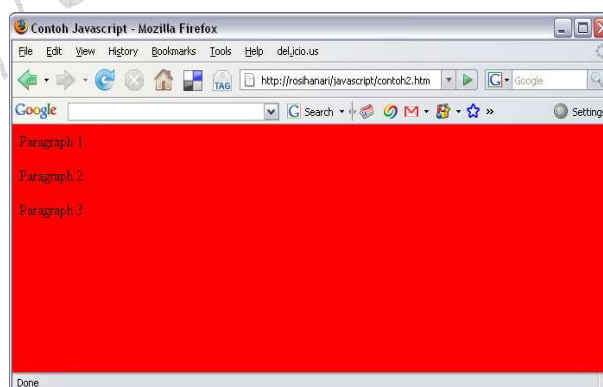
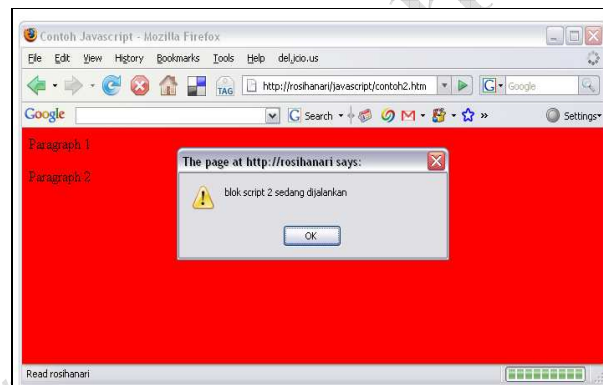
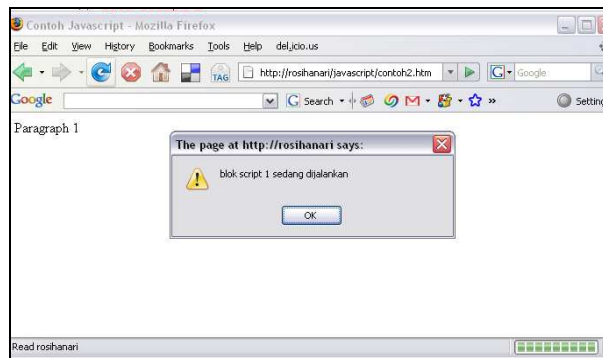
<script language="JavaScript" type="text/javascript">
  // blok script 2
  document.bgColor = "RED";
  alert("blok script 2 sedang dijalankan");
```

Javascript Tutorial

```
</script>
```

```
<p>Paragraph 3</p>  
</body>  
</html>
```

Apabila script di atas dijalankan, maka berikut ini beberapa screen shot tampilannya:



Tampilan gambar pertama, adalah ketika script blok 1 dijalankan. Gambar kedua ketika blok script 2 dijalankan. Sedangkan gambar paling bawah adalah tampilan setelah semua

Javascript Tutorial

script blok dijalankan. Dari contoh ini dapat disimpulkan bahwa apabila dalam suatu halaman web terdiri dari beberapa blok Javascript, maka jalannya script mulai dari blok paling atas, selanjutnya sampai paling bawah.

Catatan:

Function `alert()` digunakan untuk menampilkan pesan peringatan.

<http://blog.rosihanari.net>

Javascript Tutorial

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 diload. Sedangkan tipe data terkait dengan jenis data atau value yang disimpan dalam variabel.

Beberapa tipe data yang didukung dalam Javascript adalah:

- Tipe data numerik

Untuk tipe data numerik, Javascript mendukung jenis tipe data real (riil) dan integer (bulat). Jangkauan tipe data integer adalah mulai dari -2^{53} s/d 2^{53} . Sedangkan tipe data real adalah berupa bilangan riil yang memiliki jangkauan sangat besar.

- Tipe data teks atau string

Dalam Javascript, suatu value bertipe data teks atau string ditandai dengan tanda petik ganda yang mengapitnya, misalnya "Hello, World".

Selain itu juga dikenal beberapa karakter khusus yang menyatakan makna tertentu dalam Javascript (Escape Sequences). Berikut ini beberapa di antaranya:

Escape Sequences	Makna
\b	Backspace
\f	Form feed
\n	New line
\r	Carriage return
\t	Tab
\'	Single quote
\"	Double quote
\\	Backslash
\xNN	NN is a hexadecimal number that identifies a character in the Latin-1 character set.

Contoh penggunaan:

Javascript Tutorial

"Hello, World \n I Love Javascript"

- Tipe data boolean

Apabila suatu value bertipe data boolean, maka value tersebut hanya bernilai true atau false saja.

Deklarasi Variabel

Seperti yang telah diberikan di awal, bahwa variabel digunakan untuk keperluan penyimpanan value atau data di dalam memori komputer. Setelah value atau data ini disimpan dalam memori, selanjutnya dapat diload kembali untuk diproses.

Dalam Javascript, setiap kali kita 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 value pada nama variabel tersebut.

Berikut ini adalah statement untuk mendeklarasikan nama variabel bernama 'variabelku'.

```
var variabelku;
```

Setelah nama variabel dideklarasikan, selanjutnya dapat dilakukan proses assignment. Assignment adalah proses penyimpanan value pada memori melalui nama variabel yang telah dideklarasikan.

Contoh:

```
variabelku = 102;
```

Maksud statement di atas adalah menyimpan value 102 dalam variabel bernama 'variabelku'. Dalam contoh ini, tipe data value yang disimpan adalah berupa numerik. Sedangkan berikut ini contoh proses assignment pada value bertipe string dan boolean.

```
a = "Hello World";  
b = true;
```

Berikut ini contoh script dalam mendeklarasikan variabel dan proses assignment

contoh3.htm

```
<html>  
<head>
```


Javascript Tutorial

```
<title>Contoh Javascript</title>
</head>
<body>

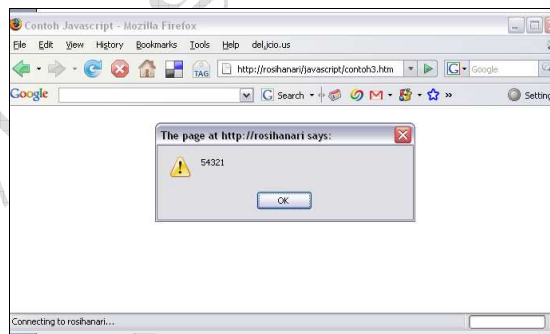
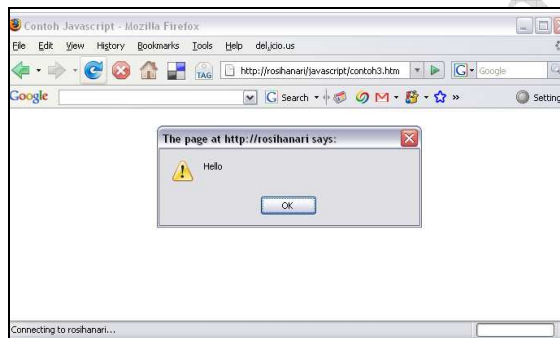
<script language="JavaScript" type="text/javascript">
var variabelku;

variabelku = "Hello"; // bertipe string
alert(variabelku);

variabelku = 54321; // bertipe numerik
alert(variabelku);

</script>

</body>
</html>
```



Catatan penting !!

- Dalam Javascript berlaku sifat case sensitive, artinya penulisan huruf besar dan kecil dalam nama variabel adalah dibedakan. Sebagai contoh adalah misalkan diberikan dua buah nama variabel 'a' dan 'A'. Meskipun dalam pembacaannya adalah sama, namun oleh Javascript, keduanya dianggap berbeda.

Javascript Tutorial

- 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.

<http://blog.rosihanari.net>

Javascript Tutorial

Menampilkan Pesan Kesalahan

Dalam dunia pemrograman, ada 2 jenis kesalahan yang dapat terjadi dalam membuat program, yaitu kesalahan sintaks (syntax error) dan kesalahan algoritmik. Kesalahan sintaks terjadi apabila programmer melanggar ketentuan atau aturan penulisan perintah dalam suatu bahasa pemrograman. Sedangkan kesalahan algoritmik disebabkan adanya logika penyelesaian masalah yang salah.

Apabila terjadi kesalahan sintak, maka biasanya hasil output tidak muncul. Sedangkan untuk kesalahan algoritmik, hasil outputnya akan muncul namun hasilnya salah karena adanya logika penyelesaian yang salah.

Lantas... bagaimana cara mengetahui pesan kesalahan sintaks dalam menuliskan kode Javascript?

Apabila kita menggunakan browser Firefox, maka cara melihat pesan kesalahan sintaks adalah dengan mengklik menu TOOL > ERROR CONSOLE.

Sebagai contoh, misalkan diberikan code berikut ini

```
<html>
<head>
<title>Contoh Javascript</title>
</head>
<body>

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

var with;

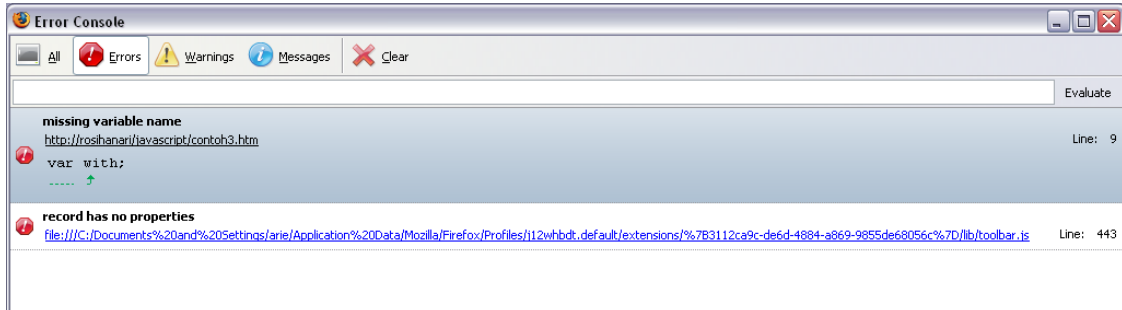
with = "Hello";
alert(with);

</script>

</body>
</html>
```

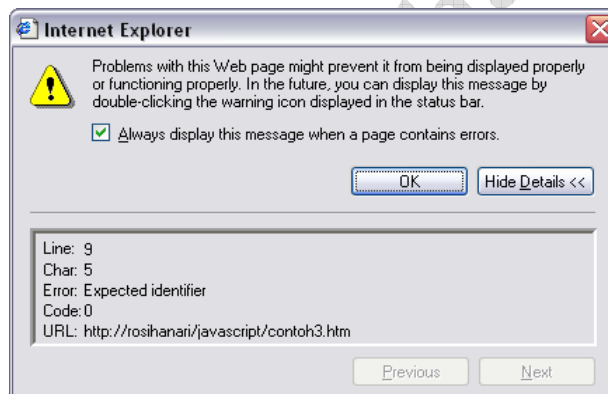
Dalam kode di atas, terdapat kesalahan sintaks karena penggunaan nama variabel yang diambil dari reserved word (with). Apabila code tersebut dijalankan di Firefox, maka tidak akan muncul apa-apa. Namun apabila kita lihat pada menu TOOL > ERROR CONSOLE, maka akan muncul pesan sbb:

Javascript Tutorial



Sedangkan untuk melihat pesan kesalahan sintaks pada IE, terlebih dahulu kita harus mengaktifkan panel untuk menampilkan pesan kesalahan. Caranya adalah klik menu **TOOLS > INTERNET OPTIONS > ADVANCED** lalu beri tanda CEK pada option 'DISPLAY A NOTIFICATION ABOUT EVERY SCRIPT ERROR'.

Apabila script yang error sebelumnya dijalankan di IE, maka berikut ini pesan kesalahannya:



Perhatikan tampilan pesan kesalahan pada kedua jenis browser di atas. Keduanya menunjukkan adanya kesalahan pada baris ke-9 (Line : 9), sehingga kita harus segera memperbaiki pada baris tersebut.

Tips:

Gunakan teks editor yang menampilkan nomor baris setiap perintah. Hal ini akan membantu kita dalam proses debugging sintaks error. Dalam hal ini, penulis merekomendasikan Notepad++ (bukan Notepad biasa) sebagai teks editornya.

Javascript Tutorial

Operasi Aritmatik

Pada modul ini akan diberikan beberapa contoh Javascript yang terdapat beberapa operasi aritmatik di dalamnya.

contoh4.htm

```
<html>
<body>

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

var hasil;

// deklarasi dan assignment dapat dilakukan sekaligus

var bilPertama = 15;
var bilKedua = 10;

hasil = bilPertama + bilKedua;
alert(hasil); // menampilkan hasil 15

hasil = bilPertama - bilKedua;
alert(hasil); // menampilkan hasil 5

hasil = bilPertama * bilKedua;
alert(hasil); // menampilkan hasil 150

hasil = bilPertama / bilKedua;
alert(hasil); // menampilkan hasil 1.5

</script>

</body>
</html>
```

contoh5.htm

```
<html>
<body>

<script language="JavaScript" type="text/javascript">
var bilangan; var hasil;

bilangan = 5;
bilangan += 4; // ekuivalen dengan statement bilangan = bilangan + 4;
alert(bilangan); // menampilkan 9
```

Javascript Tutorial

```
hasil = bilangan + 2 - 1 * 4;
alert(hasil); // menampilkan 7

</script>

</body>
</html>
```

Perhatikan statement

```
bilangan += 4;
```

Statement tersebut ekuivalen dengan statement

```
bilangan = bilangan + 4;
```

Dalam Javascript juga dikenal beberapa jenis operator khusus yang lain, misalnya ++, --, *=, dan /=

Contoh:

bilangan++;	{ ekuivalen dengan perintah bilangan = bilangan + 1; }
bilangan--;	{ ekuivalen dengan perintah bilangan = bilangan - 1; }
bilangan -= 2;	{ ekuivalen dengan perintah bilangan = bilangan - 2; }
bilangan *= 2;	{ ekuivalen dengan perintah bilangan = bilangan * 2; }
bilangan /= 2;	{ ekuivalen dengan perintah bilangan = bilangan / 2; }

contoh6.htm

```
<html>
<body>

<script language="JavaScript" type="text/javascript">
var harga = prompt("Masukkan harga mula-mula (Rp) : ");
var hargaDiskon;

hargaDiskon = harga - 0.2 * harga;

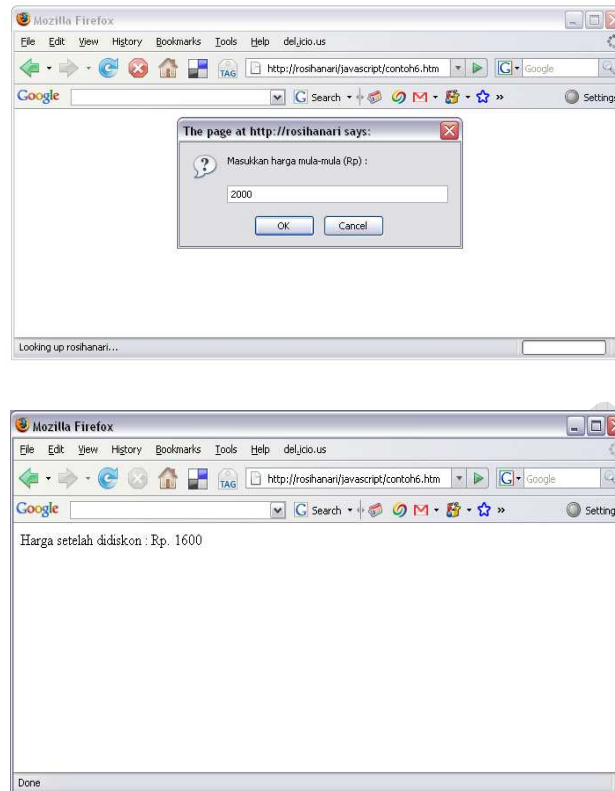
document.write("Harga setelah didiskon : Rp. " + hargaDiskon);
</script>

</body>
</html>
```

Pada contoh di atas, user diminta memasukkan harga mula-mula, lalu Javascript akan menampilkan harga setelah didiskon 20% ke dalam halaman web.

Javascript Tutorial

Tampilan scriptnya adalah sebagai berikut:



Catatan:

Perintah `prompt()` digunakan untuk menampilkan kotak dialog yang meminta user memasukkan input. Selanjutnya input tersebut akan diassign ke dalam variabel.

Sedangkan perintah `document.write()` digunakan untuk menuliskan string ke dalam halaman web.

Javascript Tutorial

Mengkonversi Tipe Data

Suatu tipe data dapat dikonversi ke tipe data yang lain. Misalnya value bertipe data string dapat dikonversi ke integer atau float (real).

Untuk melakukan konversi ke dalam tipe data integer, kita menggunakan perintah `parseInt()`, sedangkan untuk konversi ke tipe data real adalah `parseFloat()`.

Contoh:

contoh7.htm

```
<html>
<body>

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

var teks = "2.3 adalah bilangan riil";
var bil1;
var bil2;

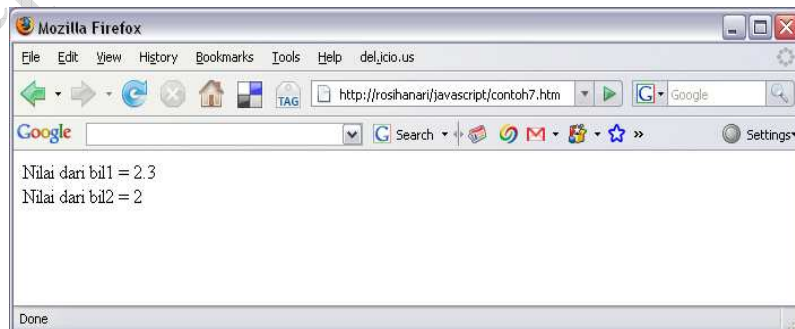
bil1 = parseFloat(teks); // konversi ke riil
document.write("Nilai dari bil1 = " + bil1 + "<br>");

bil2 = parseInt(teks); // konversi ke integer
document.write("Nilai dari bil2 = " + bil2);

</script>

</body>
</html>
```

Hasil dari script di atas adalah



Javascript Tutorial

Array

Array atau larik juga merupakan suatu tipe data, namun dalam bentuk yang lain dari yang lain. Apabila suatu variabel bertipe data array, maka kita dapat mengassign beberapa value ke variabel tersebut. Hal ini berbeda dengan variabel biasa, dimana hanya satu value saja yang dapat diassign.

Perhatikan kembali proses assignment pada variabel biasa berikut ini

```
myVariable = 45;
```

selanjutnya misalkan dilakukan lagi proses assignment pada variabel 'myVariable' dengan nilai 30

```
myVariable = 30;
```

maka akibatnya nilai sebelumnya (45) akan ditimpa dengan nilai yang baru (30). Itu adalah efek penggunaan variabel biasa, yaitu apabila diassign dengan nilai yang baru pada variabel yang sama, maka nilai yang lama akan tertimpa atau terhapus.

Dengan variabel bertipe array, kita dapat mengassign kedua nilai atau lebih dalam satu variabel. Adapun cara mendeklarasikan variabel bertipe data array adalah sebagai berikut:

```
var myVariable = new Array();
```

atau

```
var myVariable;  
myVariable = new Array();
```

Selanjutnya cara melakukan assignment pada variabel bertipe data array adalah sebagai berikut:

```
myVariable[0] = "Agus";  
myVariable[1] = 345;  
myVariable[2] = "Amir";  
myVariable[3] = 112;  
myVariable[4] = "Acong";  
myVariable[5] = 99;
```

atau

```
var myVariable = new Array("Agus", 345, "Amir", 112, "Acong", 99);
```

Javascript Tutorial

Perhatikan contoh script berikut ini tentang pendeklarasian variabel array, assignment, mengganti value elemen array serta menampilkan elemen array.

contoh8.htm

```
<html>
<body>

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

var mhs = new Array(); // deklarasi array

// assignment

mhs[0] = "Amalia";
mhs[1] = "Faza";
mhs[2] = "Nada";

// menampilkan elemen array

document.write("mhs[0] = " + mhs[0] + "<br>");
document.write("mhs[2] = " + mhs[2] + "<br>");
document.write("mhs[1] = " + mhs[1] + "<br>");

// mengubah value elemen

mhs[1] = "Ari";

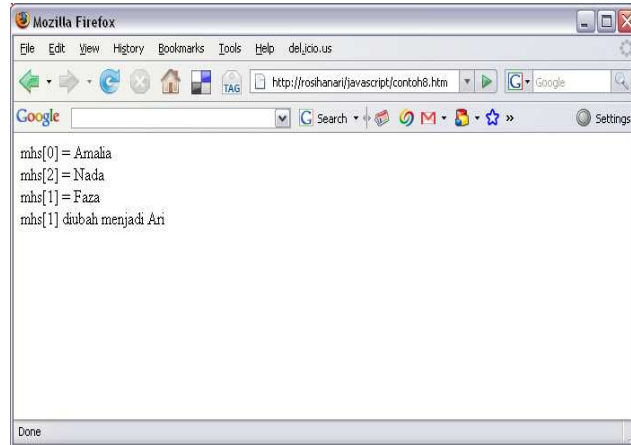
document.write("mhs[1] diubah menjadi " + mhs[1]);

</script>

</body>
</html>
```

Hasilnya adalah sebagai berikut

Javascript Tutorial



Javascript Tutorial

Penggunaan IF

Penggunaan IF terkait dengan pernyataan bersyarat. Setiap bahasa pemrograman pasti memiliki statement ini, demikian halnya dengan Javascript.

Secara umum, sintaks penulisan pernyataan IF dalam Javascript adalah sebagai berikut:

```
if (syarat)
{
    Statement jika syarat terpenuhi (bernilai TRUE)
}
```

atau

```
if (syarat)
{
    Statement jika syarat terpenuhi (bernilai TRUE)
}
else
{
    Statement jika syarat tidak terpenuhi (bernilai FALSE)
}
```

Perhatikan contoh berikut ini:

contoh9.htm

```
<html>
<body>

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

var bil;
var status;

bil = prompt("Masukkan sebarang bilangan  : ");

if (bil > 10)
{
    status = "Anda telah memasukkan bilangan lebih besar dari 10";
}

document.write(status);
</script>

</body>
</html>
```

Javascript Tutorial

Pada script di atas, keterangan hanya akan muncul apabila bilangan yang dimasukkan adalah lebih besar dari 10. Selanjutnya perhatikan script yang telah diperbaiki berikut ini

```
<html>
<body>

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

var bil;
var status;

bil = prompt("Masukkan sebarang bilangan : ");

if (bil > 10)
{
    status = "Anda telah memasukkan bilangan lebih besar dari 10";
}
else if (bil < 10)
{
    status = "Anda telah memasukkan bilangan lebih kecil dari 10";
}
else
{
    status = "Anda telah memasukkan bilangan sama dengan 10";
}

document.write(status);
</script>

</body>
</html>
```

Pada script di atas terdapat beberapa alternatif syarat, yaitu jika bilangan yang dimasukkan lebih besar 10, jika bilangan yang dimasukkan lebih kecil 10. Jika dari kedua syarat tersebut tidak ada yang memenuhi, maka pilihan terakhir adalah sama dengan 10.

Untuk menyatakan syarat, kita dapat menggunakan beberapa operator relasional sebagai berikut

- > menyatakan lebih besar dari
- < menyatakan lebih kecil dari
- >= menyatakan lebih besar atau sama dengan
- <= menyatakan lebih kecil atau sama dengan
- == menyatakan sama dengan
- != menyatakan tidak sama dengan

Selain itu kita juga dapat menggunakan operator logika.

Javascript Tutorial

&& menyatakan dan
|| menyatakan atau
! menyatakan not (negasi)

Perhatikan contoh berikut ini

contoh10.htm

```
<html>
<body>

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

var nilaiAngka;
var nilaiHuruf;

nilaiAngka = prompt("Masukkan nilai angka 0 - 100 : ");

if ((nilaiAngka >= 80) && (nilaiAngka <= 100))
{
    nilaiHuruf = "A";
}
else if ((nilaiAngka >= 70) && (nilaiAngka <= 79))
{
    nilaiHuruf = "B";
}
else if ((nilaiAngka >= 60) && (nilaiAngka <= 69))
{
    nilaiHuruf = "C";
}
else if ((nilaiAngka >= 50) && (nilaiAngka <= 59))
{
    nilaiHuruf = "D";
}
else if ((nilaiAngka >= 0) && (nilaiAngka <= 49))
{
    nilaiHuruf = "E";
}
else
{
    nilaiHuruf = "Nilai angka tidak valid";
}

document.write("Nilai huruf : " + nilaiHuruf);

</script>

</body>
</html>
```

Javascript Tutorial

Script di atas digunakan untuk mengkonversi nilai angka ke nilai huruf dengan aturan konversi sebagai berikut

80 – 100	A
70 – 79	B
60 – 69	C
50 – 59	D
0 – 49	E

Apabila nilai yang dimasukkan tidak 0 – 100 maka akan muncul keterangan nilai yang dimasukkan tidak valid.

Sebagai catatan, bentuk IF pada contoh10.htm dapat dimodifikasi menjadi syarat yang lebih sederhana. Perhatikan script berikut ini

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

var nilaiAngka;
var nilaiHuruf;

nilaiAngka = prompt("Masukkan nilai angka 0 - 100 : ");

// mengecek apakah nilai yang dimasukkan 0 -100
if (nilaiAngka >= 0) && (nilaiAngka <= 100)
{
    if (nilaiAngka >= 80)
    {
        nilaiHuruf = "A";
    }
    else if (nilaiAngka >= 70)
    {
        nilaiHuruf = "B";
    }
    else if (nilaiAngka >= 60)
    {
        nilaiHuruf = "C";
    }
    else if (nilaiAngka >= 50)
    {
        nilaiHuruf = "D";
    }
    else if (nilaiAngka >= 0)
    {
        nilaiHuruf = "E";
    }
}
else
{
```

Javascript Tutorial

```
        nilaiHuruf = "Nilai angka tidak valid";  
    }  
  
document.write("Nilai huruf : " + nilaiHuruf);  
</script>
```

<http://blog.rosihanari.net>

Javascript Tutorial

Penggunaan SWITCH

Fungsi statement SWITCH adalah sama dengan IF, yaitu menyatakan persyaratan. Sintaks dari statement ini adalah

```
switch(variabel)
{
case option1 : statement1
               break;
case option2 : statement2
               break;
case option3 : statement3
               break;
.
.
default      : statementx
               break;
}
```

Proses dari SWITCH di atas adalah melakukan cek pada value variabel. Apabila valuenya sama dengan option1 maka statement yang dilakukan adalah statement1. Apabila valuenya sama dengan option2 maka statement2 yang dijalankan, begitu seterusnya. Namun apabila tidak ada value yang sama, maka statementx lah yang dijalankan.

Oya... tipe data dari variabel yang akan dicek haruslah ordinal, misalnya integer. Dalam hal ini tipe data riil tidak dapat digunakan. Untuk memastikan bahwa nilai yang akan dicek pada variabel adalah bertipe integer, sebaiknya gunakan perintah parseInt().

Perhatikan contoh berikut ini:

contoh11.htm

```
<html>
<body>

<script language="JavaScript" type="text/javascript">
var nilaiAngka;
var nilaiHuruf;

nilaiAngka = parseInt(prompt("Masukkan sebuah bil bulat 1 s/d 5 : "));

switch(nilaiAngka)
{
case 1 : document.write("Anda memasukkan bilangan 1");
        break;
```

Javascript Tutorial

```
case 2 : document.write("Anda memasukkan bilangan 2");
        break;
case 3 : document.write("Anda memasukkan bilangan 3");
        break;
case 4 : document.write("Anda memasukkan bilangan 4");
        break;
case 5 : document.write("Anda memasukkan bilangan 5");
        break;
default : document.write("Bilangan yang Anda masukkan tidak 1 s/d 5");
          break;
}

</script>

</body>
</html>
```

Script di atas akan meminta user untuk memasukkan sebuah bilangan bulat 1 s/d 5. Selanjutnya script akan menampilkan status bilangan yang dimasukkan user. **Apabila perintah parseInt() tidak digunakan, maka hasilnya akan salah.**

Javascript Tutorial

Perulangan FOR

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. Pada bagian ini akan dijelaskan penggunaan FOR untuk perulangan terlebih dahulu.

Secara umum sintaks dari FOR adalah sebagai berikut:

```
for(nilai awal; syarat; increment/decrement)
{
    Statement...;
}
```

Maksud dari sintaks di atas adalah, bahwa bagian Statement akan terus diulang selama nilai syarat bernilai TRUE, dengan suatu nilai awal tertentu.

Sebagai contoh, perhatikan script berikut ini:

contoh12.htm

```
<html>
<body>

<script language="JavaScript" type="text/javascript">
var counter;

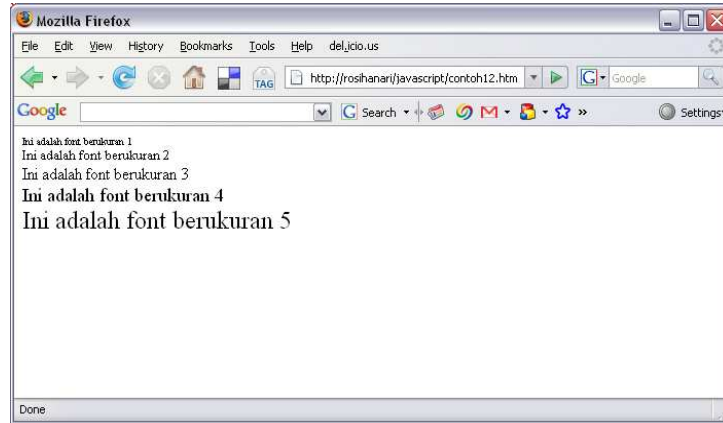
for(counter=1; counter<=5; counter++)
{
    document.write("<font size='" + counter + "'>
        Ini adalah font berukuran " + counter + "</font><br>");
}

</script>
</body>
</html>
```

Dalam contoh di atas, perulangan FOR terjadi mulai dari nilai awal counter = 1, dengan kenaikan nilainya adalah 1 (lihat counter++). Selama nilai counter ≤ 5 , maka perulangan akan terus dilakukan.

Sehingga hasilnya adalah:

Javascript Tutorial



Dalam FOR, tidak selalu dalam bentuk increment, namun juga dapat berbentuk decrement. Perhatikan contoh berikut ini.

contoh13.htm

```
<html>
<body>

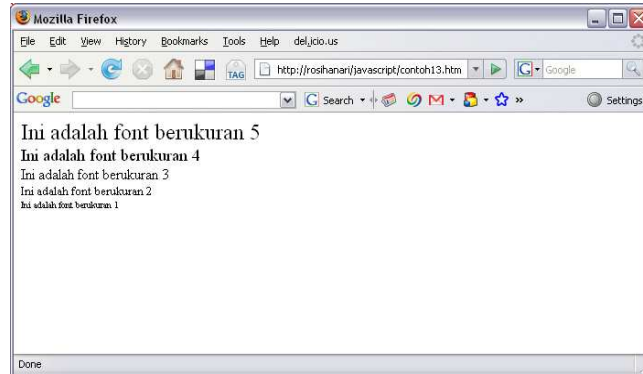
<script language="JavaScript" type="text/javascript">
var counter;

for(counter=5; counter>=1; counter-- )
{
    document.write("<font size='" + counter + "'>
                    Ini adalah font berukuran " + counter + "</font><br>");
}
</script>

</body>
</html>
```

Hasilnya adalah:

Javascript Tutorial



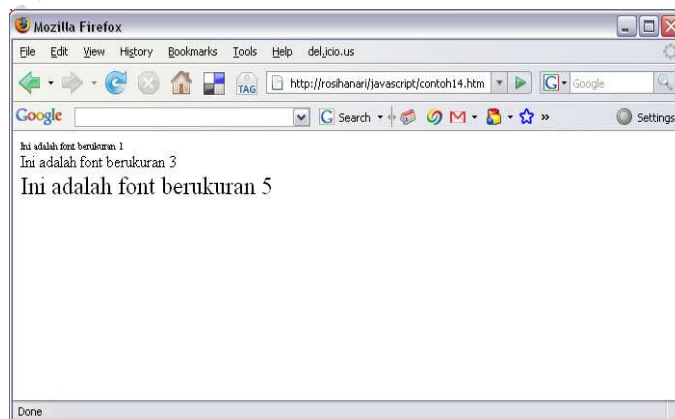
Suatu increment tidak selalu harus dengan tingkat kenaikan 1, begitu pula dengan decrement. Sebagai contoh, perhatikan statement FOR berikut ini yang memiliki tingkat kenaikan 2 untuk incrementnya.

contoh14.htm

```
<html>
<body>

<script language="JavaScript" type="text/javascript">
var counter;

for(counter=1; counter<=5; counter+=2)
{
    document.write("<font size='" + counter + "'>
                    Ini adalah font berukuran " + counter + "</font><br>");
}
</script>
</body>
</html>
```



Javascript Tutorial

Perulangan WHILE

Pada prinsipnya bentuk WHILE adalah mirip dengan FOR. Secara umum sintaks dari WHILE adalah

```
while(syarat)
{
    statement;
}
```

Dalam sintaks di atas, bagian statement akan terus diulang selama syarat bernilai TRUE.

Sebagai contoh, perhatikan statement dalam FOR dan WHILE berikut ini. Kedua statement adalah ekuivalen atau sama (perhatikan perintah dengan warna yang sama).

```
for(counter=1; counter<=5; counter++)
{
    document.write("Hello World <br>");
}
```

```
-----
counter = 1;
while(counter<=5)
{
    document.write("Hello World <br>");
    counter++;
}
```

Meskipun semua statement FOR dapat juga dinyatakan sebagai WHILE, namun tidak berlaku kebalikannya. Tidak semua statement WHILE dapat dinyatakan dalam FOR. Nah... inilah kelebihan WHILE.

Kelebihan WHILE dibandingkan FOR adalah bahwa WHILE dapat digunakan apabila jumlah perulangan tidak diketahui. Sedangkan untuk FOR harus diketahui jumlah perulangannya.

Sebagai contoh perumpaan, misalkan ada statement begini: 'Saya akan terus makan selama saya masih lapar'. Statement tersebut adalah juga termasuk konsep perulangan dalam kehidupan sehari-hari. Dalam hal ini, jumlah berapa kali harus makan tidak diketahui, melainkan hanya syarat makannya saja yang diketahui, yaitu selama masih lapar.

Javascript Tutorial

Nah... perhatikan contoh berikut ini. Script ini akan terus meminta user untuk mengisi sebuah angka, selama angka yang dimasukkan bukan 5.

contoh15.htm

```
<html>
<body>

<script language="JavaScript" type="text/javascript">
var angka;

while(angka != 5)
{
    angka = prompt("Masukkan sebuah angka : ");
}
document.write("Anda telah memasukkan angka 5");

</script>
</body>
</html>
```

Perhatikan syarat (yang dicetak biru) pada script di atas. Syarat tersebut bermakna, selama nilai angka tidak sama dengan 5, maka user akan selalu diminta memasukkan angka. Sehingga dalam kasus ini tidak diketahui berapa jumlah perulangan yang harus dilakukan, namun perulangannya tergantung dari syarat. Nah... untuk kasus seperti ini, kita tidak bisa membuatnya dengan FOR.

Javascript Tutorial

Function

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.

Secara umum, sintaks penulisan function dalam Javascript adalah sebagai berikut:

```
function namafunction(parameter)
{
    Statement...
}
```

Dalam hal ini, parameter dalam function tidak harus ada.

Perhatikan contoh function berikut ini yang digunakan untuk menghitung diskon 50% dari harga mula-mula, dimana harga mula-mula merupakan parameternya.

contoh16.htm

```
<html>
<head>
    <script language="JavaScript" type="text/javascript">
        function diskon(hargaAwal)
        {
            var diskon;
            diskon = 0.5 * hargaAwal;
            hargaDiskon = hargaAwal - diskon;

            return hargaDiskon;
        }
    </script>
</head>

<body>
    <script language="JavaScript" type="text/javascript">
        var harga = prompt("Masukkan harga mula-mula (Rp) : ");
        document.write("Harga stlh diskon 50% : Rp. " + diskon(harga));
    </script>
</body>
</html>
```

Dalam contoh di atas, function diletakkan pada bagian <head>, namun dapat pula kita meletakkannya dalam <body>, seperti berikut ini

Javascript Tutorial

```
<html>
<head>
</head>

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

    function diskon(hargaAwal)
    {
      var diskon;

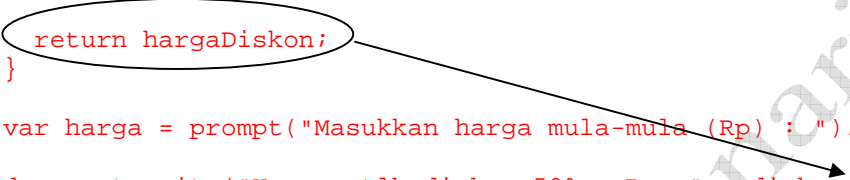
      diskon = 0.5 * hargaAwal;
      hargaDiskon = hargaAwal - diskon;

      return hargaDiskon;
    }

    var harga = prompt("Masukkan harga mula-mula (Rp) : ");

    document.write("Harga stlh diskon 50% : Rp. " + diskon(harga));
  </script>

</body>
</html>
```

A diagram consisting of an oval around the 'return hargaDiskon;' line in the 'diskon' function, with an arrow pointing from this oval to the 'diskon(harga)' argument in the 'document.write' statement.

Proses pada script di atas adalah sebagai berikut:

Mula-mula user diminta memasukkan harga mula-mula, harga ini diassign ke dalam variabel harga, misalnya 1000. Selanjutnya script mencetak harga setelah didiskon. Dalam hal ini, script akan memanggil function diskon(harga) atau diskon(1000). Karena function diskon memiliki parameter hargaAwal, maka nilai dari hargaAwal ini sama dengan nilai dari variabel harga, pada contoh ini adalah bernilai 1000. Selanjutnya akan dihitung nilai diskon yaitu $0.5 * 1000 = 500$, dan hargaDiskon yaitu $1000 - 500 = 500$.

Perintah return hargaDiskon, artinya akan mengembalikan nilai hargaDiskon (500) ke function yang memanggilnya, yaitu diskon(harga), lihat tanda panah pada script di atas. Sehingga akan tercetak 'Harga stlh diskon 50% : Rp. 500'.

Suatu function tidak harus terdapat return value. Perhatikan contoh berikut ini.

contoh17.htm

```
<html>
<head>
  <script language="JavaScript" type="text/javascript">
    function cetakTeks(teks)
    {
      document.write(teks + "<br>");
    }
  </script>
</head>
<body>
  <script>
    cetakTeks("Hello World!");
  </script>
</body>
</html>
```

Javascript Tutorial

```
        </script>
</head>

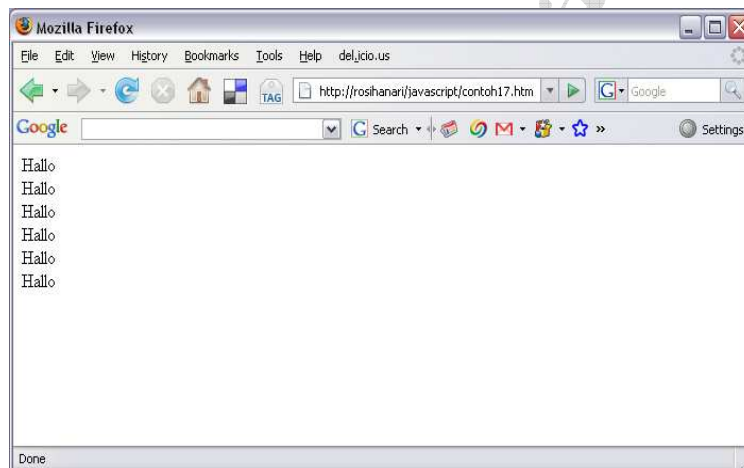
<body>
    <script language="JavaScript" type="text/javascript">

        var kata = "Hallo";
        var i;

        for(i=0; i<=5; i++)
        {
            cetakTeks(kata);
        }

    </script>
</body>
</html>
```

Output dari script di atas adalah



Javascript Tutorial

JavaScript - Bahasa Pemrograman Berbasis Obyek

Bagi Anda yang belum familiar dengan pemrograman berbasis obyek, berikut ini akan dijelaskan apa yang dimaksud dengan istilah tersebut.

Sebagai permulaan, akan dijelaskan tentang apa itu obyek. Dalam realitas sehari-hari, terdapat banyak sekali obyek di sekitar kita, misal saja mobil kita, rumah, buku, kulkas dll.

Misalkan Anda memandang suatu obyek, katakanlah mobil Mercedes Benz Anda. Apa yang bisa dilihat? Mungkin ada yang mengatakan mobilnya berwarna merah, beroda 4, memiliki 4 pintu misalnya, dll. Ketika Anda menyebut seperti itu, berarti Anda sedang menjelaskan properti suatu obyek.

Lantas... misalkan ada pertanyaan, apa yang bisa dilakukan dengan mobil Mercedes Benz Anda? Mungkin ada yang mengatakan mobilnya dapat dijalankan mundur, maju, dapat dipercepat hingga 300 km/jam, atau dapat dijalankan dengan 2 roda ☺ dan seterusnya. Nah... ketika Anda menyebut hal ini, maka Anda sedang menyebutkan method suatu obyek.

Sehingga dari contoh di atas, dapat dikatakan bahwa suatu obyek terdiri dari beberapa properti dan method. Dalam bahasa pemrograman berbasis obyek, konsep seperti ini akan diadopsi.

Membuat Obyek dalam Javascript

Setiap obyek dalam Javascript memiliki beberapa properti dan method untuk memanipulasi data. Sebagai contoh obyek dalam Javascript adalah Array(), kita sudah bahas materi ini sebelumnya. Obyek ini memiliki properti length yang digunakan untuk memperoleh banyaknya item data dalam array. Obyek Array() juga memiliki suatu method bernama sort() yang digunakan untuk mengurutkan data dalam array tersebut. Masih banyak obyek lain dalam Javascript yang dapat digunakan.

Pada pembahasan sebelumnya, setiap kali kita membuat sebuah variabel bertipe array, kita menggunakan perintah berikut ini

```
var namaVariabel = new Array();
```

atau dapat pula berbentuk seperti di bawah ini, apabila langsung disebutkan elemennya.

Javascript Tutorial

```
var myArray = new Array("Joko", "Amir", "Budi");
```

Perintah di atas pada dasarnya adalah membuat obyek. Dalam hal ini namaVariabel adalah suatu obyek berbentuk Array().

Secara umum, perintah untuk membuat obyek dalam Javascript adalah

```
var namaObyek = new bentukObyek();
```

Dalam pemrograman berbasis obyek, bentukObyek() dinamakan *constructor*.

Selain obyek berbentuk Array(), terdapat obyek lain dalam Javascript, yaitu Date(), String(), Math(), dan beberapa lagi yang lain. Berikut ini contoh membuat obyek berbentuk Date().

```
var tanggal = new Date("1 Jan 2000");
```

Menggunakan Properti Obyek

Untuk menggunakan properti dari obyek, caranya cukup mudah yaitu hanya dengan menuliskan nama obyek diikuti dengan dot (titik), lalu pilih properti yang diinginkan. Sebagai contoh, kita akan menggunakan properti obyek berbentuk Array untuk menghitung jumlah elemen yang tersimpan dalam array. Properti ini bernama length.

contoh18.htm

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

    var arrayKu = new Array();

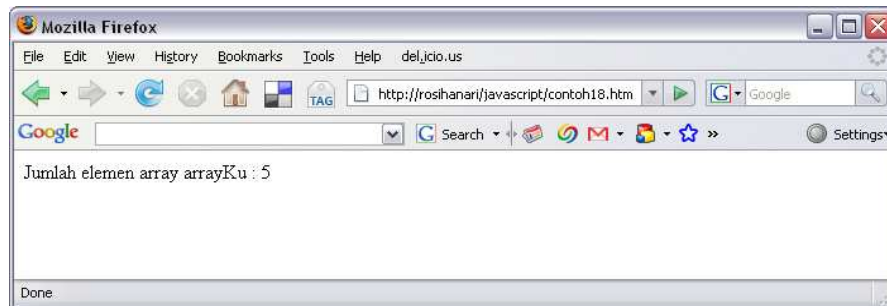
    arrayKu[0] = "Jaka";
    arrayKu[1] = "Jiki";
    arrayKu[2] = "Juku";
    arrayKu[3] = "Jeke";
    arrayKu[4] = "Joko";

    document.write("Jumlah elemen array arrayKu : " + arrayKu.length);

  </script>
</body>
</html>
```

Output dari script di atas adalah

Javascript Tutorial



Menggunakan Method dari Obyek

Cara menggunakan method dari obyek sama dengan menggunakan properti, yaitu hanya dengan menuliskan nama obyek, kemudian diikuti dengan dot (titik), lalu tulis nama method yang diinginkan. Beberapa method merupakan suatu function sehingga akan mengembalikan suatu nilai.

Berikut ini cara menggunakan salah satu method dari obyek berbentuk Array().

contoh19.htm

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

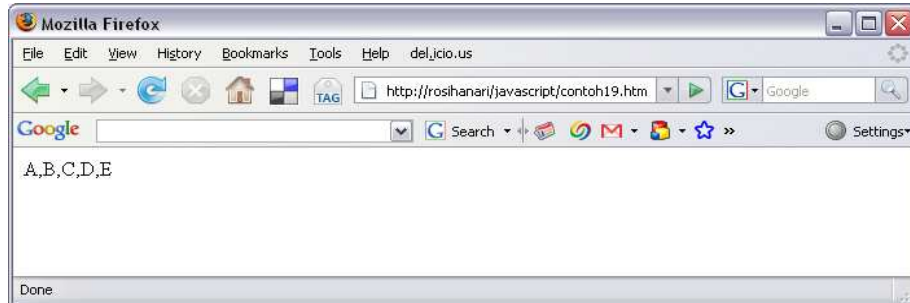
    var arrayKu = new Array();

    arrayKu[0] = "A";
    arrayKu[1] = "C";
    arrayKu[2] = "E";
    arrayKu[3] = "D";
    arrayKu[4] = "B";

    document.write(arrayKu.sort());
  </script>
</body>
</html>
```

Script di atas menunjukkan cara menggunakan method sort() pada obyek berbentuk Array(). Method ini berfungsi untuk mengurutkan/sorting elemen array. Perhatikan, bahwa dalam urutan elemen arrayKu adalah "A", "C", "E", "D", "B". Selanjutnya akan kita lihat hasil scriptnya di browser.

Javascript Tutorial



Beberapa Obyek Dalam JavaScript

Nama Obyek : Array

Properties :

Nama Properti	Mulai Digunakan Pada	Deskripsi
length	JavaScript 1.1 Jscript 2.0	Menghasilkan jumlah elemen dari array

Methods :

Method	Mulai Digunakan Pada	Deskripsi
concat()	JavaScript 1.2 Jscript 3.0	Menggabung elemen dari dua array
join()	JavaScript 1.1 Jscript 2.0	Menggabung semua elemen array menjadi sebuah string tunggal.
pop()	JavaScript 1.2	Mengambil elemen terakhir array. Method ini tidak didukung di IE
push()	JavaScript 1.2	Menambahkan elemen pada bagian terakhir array. Tidak didukung di IE.
reverse()	JavaScript 1.1 Jscript 2.0	Membalik urutan elemen array, sehingga elemen terakhir menjadi paling awal, dan juga kebalikannya.
shift()	JavaScript 1.2	Menghapus elemen pertama dari array dan mengembalikan elemen pertama tersebut. Tidak didukung di IE.

Javascript Tutorial

Method	Mulai Digunakan Pada	Deskripsi
slice()	JavaScript 1.2 Jscript 3.0	Mengembalikan subarray dari suatu array.
sort()	JavaScript 1.1 Jscript 2.0	Sorting elemen array
splice()	JavaScript 1.2	Menambahkan, menghapus dan mengganti elemen suatu array. Tidak didukung di IE.
toString()	JavaScript 1.1 Jscript 2.0	Mengkonversi obyek array menjadi string

Nama Obyek : Date

Methods :

Method	Introduced	Description
getDate()	JavaScript 1.0 Jscript 1.0	Mendapatkan tanggal dari obyek date (1-s/d 31)
getDay()	JavaScript 1.0 Jscript 1.0	Mendapatkan hari dalam seminggu (1 s/d 7). 1 = Senin, 2 = Selasa, dst...
getFullYear()	JavaScript 1.3 Jscript 3.0	Mendapatkan tahun
getHours()	JavaScript 1.0 Jscript 1.0	Mendapatkan jam (00 s.d 23)
getMilliseconds()	JavaScript 1.3 Jscript 3.0	Mendapatkan jumlah milisekon
getMinutes()	JavaScript 1.0 Jscript 1.0	Mendapatkan jumlah menit (0 s/d 59)
getMonth()	JavaScript 1.0 Jscript 1.0	Mendapatkan bulan (0 s/d 11), 0 = Januari, 1 = Pebruari, dst...
getSeconds()	JavaScript 1.0 Jscript 1.0	Mendapatkan jumlah detik (0 s/d 59).
getTime()	JavaScript 1.0 Jscript 1.0	Mendapatkan jumlah milisecon sejak January 1 1970 00:00:00

Perhatikan contoh script berikut yang menggunakan beberapa method untuk obyek Date.

Javascript Tutorial

contoh20.htm

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

    var hariIni = new Date();

    hari = hariIni.getDay();

    if (hari = 1) namaHari = "Senin";
    else if (hari = 2) namaHari = "Selasa";
    else if (hari = 3) namaHari = "Rabu";
    else if (hari = 4) namaHari = "Kamis";
    else if (hari = 5) namaHari = "Jumat";
    else if (hari = 6) namaHari = "Sabtu";
    else if (hari = 7) namaHari = "Minggu";

    tanggal = hariIni.getDate();
    bulan = hariIni.getMonth() + 1;
    tahun = hariIni.getFullYear();

    document.write("Hari ini adalah " + namaHari +
                  ", Tanggal " + tanggal + "/" + bulan +
                  "/" + tahun);

  </script>
</body>
</html>
```

Script di atas akan menampilkan nama hari ini, dan tanggalnya (tanggal/bulan/tahun). Berikut ini adalah outpurnya



Javascript Tutorial

Masih banyak obyek lain dalam Javascript yang dapat digunakan, namun karena keterbatasan kemampuan penulis, maka untuk bab ini hanya dipilih beberapa obyek saja. Pembahasan obyek yang lain akan dilanjutkan pada bab tersendiri.

<http://blog.rosihanari.net>

Javascript Tutorial

Pemrograman Browser

Browser dalam Javascript juga terdiri dari beberapa obyek, sehingga kita dapat melakukan pemrograman pada browser. Sebagai contoh adalah adanya obyek bernama window. Dengan obyek ini, kita dapat memanipulasi tampilan terkait dengan window browser.

Kumpulan obyek yang terdapat dalam browser ini secara umum dinamakan BOM (Browser Object Model).

Berikut ini akan dikenalkan beberapa obyek yang terkait dengan browser.

Obyek Window

Obyek window terkait dengan frame window tempat halaman web ditampilkan. Beberapa properti yang dapat kita gunakan dalam obyek ini misalnya, ukuran window browser, jenis browser yang digunakan user, ukuran screen user dan banyak lagi.

Obyek window ini merupakan obyek global, artinya kita tidak perlu menyebutnya dalam script, akan tetapi langsung disebutkan nama method atau propertinya. Sebagai contoh adalah penggunaan perintah alert(). Method alert() ini sebenarnya adalah method dalam obyek window, namun kita langsung dapat menggunakannya tanpa menuliskan nama obyek, misal:

```
alert("Hello World");
```

tapi kita juga dapat menuliskannya secara utuh

```
window.alert("Hello World");
```

Selanjutnya kita akan mencoba properti di obyek window, yaitu defaultStatus. Properti ini digunakan untuk mengubah status browser. Biasanya status ini terletak di bagian kiri bawah window browser.

contoh21.htm

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">
window.defaultStatus = "Hello and Welcome";
</script>
</head>
```

Javascript Tutorial

</html>

Karena obyek window adalah global obyek, kita dapat juga menuliskan script di atas seperti berikut

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">
defaultStatus = "Hello and Welcome";
</script>
</head>
</html>
```

Berikut ini adalah tampilan pengubahan efek status window.



Perhatikan script berikut ini

```
<script language="JavaScript" type="text/JavaScript">
    defaultStatus = "Hello and Welcome";
</script>
```

Apabila Anda perhatikan, seolah-olah defaultStatus adalah sebuah variabel. Tapi ingat !!! ini bukanlah suatu nama variabel, namun suatu properti dari obyek window. Sehingga apabila kita menambahkan `var defaultStatus;`, hasilnya tidak seperti yang diharapkan.

```
<script language="JavaScript" type="text/JavaScript">
    var defaultStatus;
    defaultStatus = "Hello and Welcome";
</script>
```

Dengan penambahan `var` seperti di atas, status browser tidak akan berubah atau tidak ada efeknya sama sekali.

Javascript Tutorial

Beberapa properti dari obyek window, juga merupakan suatu obyek. Misalnya history, document, navigator, screen dan location.

Obyek History

Obyek ini sebenarnya adalah properti dari obyek window. Obyek history digunakan untuk mengakses history halaman web yang telah dibuka/dikunjungi. Beberapa method yang dapat digunakan dalam obyek ini adalah back() dan forward(). Penggunaan kedua method tersebut mirip dengan tombol BACK dan FORWARD dalam browser, yaitu kembali ke halaman berikutnya serta menuju ke halaman berikutnya. Contoh

```
<script language="JavaScript" type="text/JavaScript">
    history.back();
</script>
```

Terdapat pula method go() dalam obyek history ini. Contoh penggunaan:

```
history.go(2);
```

Perintah di atas digunakan untuk menuju ke 2 halaman berikutnya (sama hal nya seperti menekan tombol FORWARD 2x)

```
history.go(-1);
```

Perintah di atas sama hal nya seperti menekan tombol BACK sekali).

Ingat!! Karena pada prinsipnya obyek history merupakan properti dari window, maka kita pun dapat menambahkan nama obyek window di awal perintah history, misal:

```
window.history.go(-1);
window.history.back();
```

Berikut ini contoh script yang menampilkan tombol untuk BACK dan FORWARD.

contoh23.htm

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">

function kembali()
{
    history.back();
}
```

Javascript Tutorial

```
function lanjut()  
{  
    history.forward();  
}  
  
</script>  
  
</head>  
<body>  
    <input type="button" name="back" value="BACK" onclick="kembali()">  
    <input type="button" name="forward" value="FORWARD" onclick="lanjut()">  
</body>  
</html>
```

Pada contoh script di atas, kita sudah menggunakan event handler, dalam hal ini adalah onclick, artinya apabila ada suatu komponen diberikan event ini sebagai atributnya, maka kita dapat membuat action ketika komponen tersebut diklik. Lebih lanjut tentang macam-macam event handler akan dijelaskan pada bab lain.

Obyek Location

Dengan obyek ini, kita bisa melakukan navigasi ke halaman-halaman web lain. Untuk melakukan navigasi dengan obyek ini, terdapat method dan properti yang dapat kita gunakan, yaitu

```
location.replace("target");  
location.href = "target";
```

Keterangan: target adalah nama halaman web yang ingin dituju.

Lantas apa perbedaan dari method dan properti di atas? Apabila kita menggunakan replace() maka halaman yang sedang aktif akan dihapus dari data history dan diganti dengan halaman target. Sedangkan properti href tidak akan menghapus data history halaman yang sedang aktif, dan menyimpan halaman target ke dalam data history yang baru. Sebagai contoh:

```
location.replace("myPage.htm");  
location.href = "myPage.htm";
```

Karena obyek window bersifat global, dan location sebenarnya adalah properti dari window, maka dapat kita tuliskan sebagai berikut

```
window.location.replace("myPage.htm");  
window.location.href = "myPage.htm";
```

Javascript Tutorial

Obyek Screen

Untuk obyek screen , terdapat 2 properti yang dapat kita gunakan untuk mengetahui size dari screen browser, yaitu width dan height. Berikut ini contoh penggunaan properti ini untuk menampilkan informasi ukuran browser, melalui sebuah klik tombol.

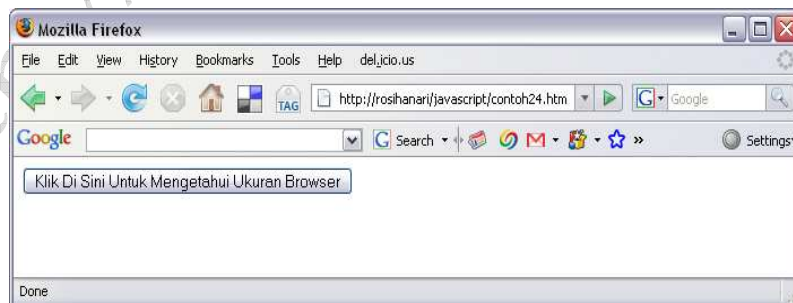
contoh24.htm

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">
function sizeBrowser()
{
    var panjang, lebar;

    panjang = screen.width;
    lebar    = screen.height;

    alert("Panjang browser Anda: " + panjang + " px \n
        Size lebar browser Anda: " + lebar + " px" );
}
</script>
</head>
<body>
    <input type="button" name="klik" value="Klik Di Sini Untuk
        Mengetahui Ukuran Browser" onclick="sizeBrowser()">
</body>
</html>
```

Berikut ini hasil tampilan di browser dan juga informasi ukurannya.



Javascript Tutorial



Javascript Tutorial

Pemrosesan FORM

Pada bagian ini akan dipaparkan bagaimana memproses suatu input data yang berasal dari komponen form, seperti text box, textarea, combo box, radio button dll.. Pada dasarnya untuk semua tag komponen form disisipkan dalam tag form

```
<form name="namaform">
.
.
.
</form>
```

Sebagai contoh adalah

```
<form name="namaform">
  <input type="text" name="edit1">
  <input type="text" name="edit2">
.
.
</form>
```

Untuk membaca value yang diinputkan melalui komponen dalam form menggunakan JavaScript menggunakan obyek document dengan sintaks berikut ini:

```
document.namaform.namakomponen.value;
```

Sebagai contoh, script berikut ini adalah script untuk membaca value yang diinputkan user melalui komponen text lalu menampilkan sebagai alert.

contoh26.htm

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">

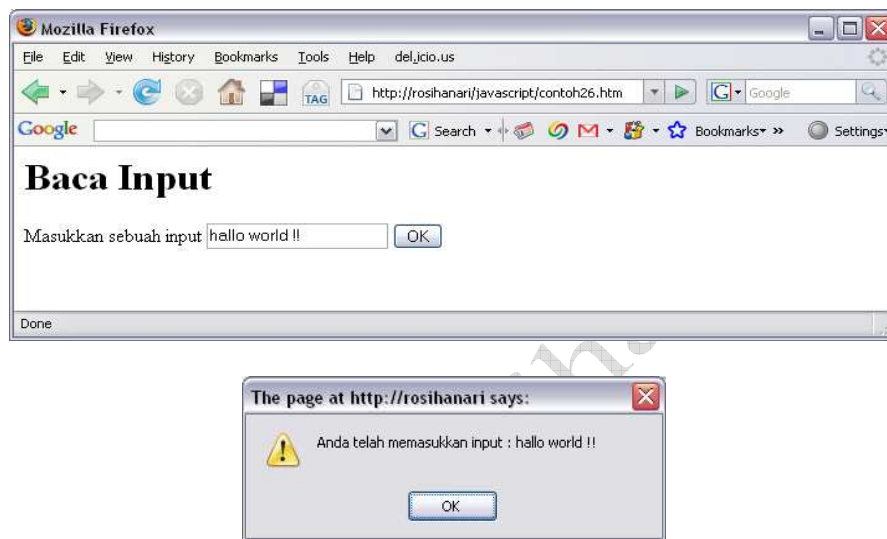
function baca()
{
  var nilai = document.formku.input.value;
  alert("Anda telah memasukkan input : " + nilai);
}

</script>
</head>
<body>
  <h1>Baca Input</h1>
  <form name="formku">
```


Javascript Tutorial

```
Masukkan sebuah input <input type="text" name="input">
<input type="button" name="button" value="OK" onclick="baca()">
</form>
</body>
</html>
```

Apabila script di atas dijalankan, maka berikut ini adalah tampilannya:



Selanjutnya, perhatikan contoh berikut ini. Pada contoh ini, akan dibuat semacam kalkulator sederhana yang mengoperasikan penjumlahan, pengurangan, perkalian dan pembagian dari dua buah bilangan.

contoh27.htm

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">

function hitung()
{
    var bilangan1 = parseFloat(document.formku.bil1.value);
    var bilangan2 = parseFloat(document.formku.bil2.value);
    var op = document.formku.operasi.value;
    var hasil;

    if (op == "jumlah") hasil = bilangan1 + bilangan2;
    else if (op == "kurang") hasil = bilangan1 - bilangan2;
    else if (op == "kali") hasil = bilangan1 * bilangan2;
    else if (op == "bagi") hasil = bilangan1 / bilangan2;
```

Javascript Tutorial

```
        document.formku.hasil.value = hasil;
    }

</script>
</head>
<body>
    <h1>Kalkulator Sederhana</h1>
    <form name="formku">
        Bil Pertama <input type="text" name="bil1">

        <select name="operasi" onchange="hitung()">
            <option value="jumlah">Dijumlahkan</option>
            <option value="kurang">Dikurangkan</option>
            <option value="kali">Dikalikan</option>
            <option value="bagi">Dibagi</option>
        </select>

        Bil Kedua <input type="text" name="bil2"> =
        <input type="text" name="hasil">
    </form>
</body>
</html>
```

Berikut ini adalah hasil tampilannya:



Pada contoh ini, hasil operasi akan muncul setelah user memilih jenis operasi yang diinginkan pada combobox. Untuk itu event yang digunakan pada contoh ini adalah onchange. Event tersebut diberikan pada combobox.

Selanjutnya, perhatikan pada perintah

```
var bilangan1 = parseFloat(document.formku.bil1.value);
var bilangan2 = parseFloat(document.formku.bil2.value);
```

Mungkin Anda bertanya, mengapa harus ada parseFloat()? Jawabannya adalah karena apabila tidak menggunakannya, maka input value dari bilangan pertama dan kedua dianggap sebagai string. Sehingga apabila tidak menggunakan perintah parseFloat()

Javascript Tutorial

akibatnya adalah mengoperasikan string. Misalkan akan menjumlahkan bilangan pertama adalah 5 dan bilangan kedua adalah 7, maka hasilnya adalah 57 (penjumlahan string).

<http://blog.rosihanari.net>

Javascript Tutorial

Event Handler

Pada bab sebelumnya telah diberikan beberapa contoh script yang telah menggunakan event handler onclick() yaitu apabila suatu elemen diklik maka akan terjadi suatu efek/action tertentu. Apakah ada event yang lain selain onclick()? Jawabannya jelas ada, berikut ini beberapa event yang dapat kita gunakan.

Mouse Events

Event	Introduced	Description
onclick	JavaScript 1.0	Muncul ketika user mengklik elemen/komponen
ondblclick	JavaScript 1.2	Muncul ketika user mengklik ganda pada elemen/komponen
onmousedown	JavaScript 1.2	Muncul ketika user menekan tombol pada mouse
onmousemove	JavaScript 1.2	Muncul ketika user menggerakkan pointer mouse
onmouseout	JavaScript 1.1	Muncul ketika user menggerakkan pointer mouse keluar dari elemen/komponen control.
onmouseover	JavaScript 1.0	Muncul ketika user menggerakkan pointer mouse melalui/melewati elemen/komponen
onmouseup	JavaScript 1.2	Muncul ketika user melepas tombol mouse

Keyboard Events

Event	Introduced	Description
onkeydown	JavaScript 1.2	Muncul ketika user menekan sebuah tombol pada keyboard
onkeypress	JavaScript 1.2	Muncul ketika user menekan sebuah tombol keyboard dan akan terus muncul sampai user melepas tombol
onkeyup	JavaScript 1.2	Muncul ketika user melepas tombol yang telah ditekan

Javascript Tutorial

HTML Control Events

Event	Introduced	Description
onblur	JavaScript 1.0	Muncul ketika HTML control kehilangan focus.
onchange	JavaScript 1.0	Muncul ketika HTML control kehilangan focus and nilainya telah berubah.
onfocus	JavaScript 1.0	Muncul ketika HTML control sedang focus
onreset	JavaScript 1.1	Muncul ketika user menekan tombol reset dalam form
onselect	JavaScript 1.0	Muncul ketika user memilih teks pada HTML control.
onsubmit	JavaScript 1.0	Muncul ketika user melakukan submit form

Window Events

Event	Introduced	Description
onload	JavaScript 1.0	Muncul ketika window telah selesai loading
onresize	JavaScript 1.2	Muncul ketika user mengubah ukuran window
onunload	JavaScript 1.0	Muncul ketika user keluar dari halaman/dokumen web

Berikut ini contoh penggunaan event handler

contoh25.htm

```
<html>
<head>
<script language="JavaScript" type="text/JavaScript">

function dekati()
{
    document.formku.status.value = "Anda mendekati obyek link";
}

function jauhi()
{
    document.formku.status.value = "Anda menjauhi obyek link";
}

</script>
</head>
<body>
    <h1><a href="" onmouseover="dekati()" onmouseout="jauhi()">
```

Javascript Tutorial

```
Obyek Link</a>
</h1>

<form name="formku">
  <textarea name="status"></textarea>
</form>
</body>
</html>
```

Script di atas menunjukkan efek penggunaan event onmouseover dan onmouseout pada suatu obyek berbentuk link. Apabila obyek link didekati kursor mouse, maka akan muncul keterangan pada texearea. Begitu pula ketika kursor mouse dijauhkan.

Untuk mengubah-ubah keterangan pada textarea dalam contoh ini menggunakan obyek document. Penjelasan lebih lanjut tentang obyek ini akan disampaikan di lain waktu.

