



# Desain & Pemrograman Web

# **Javascript**

Salhazan Nasution, S.Kom, MIT



# Pengenalan Javascript



# Pengenalan Javascript

- Javascript merupakan bahasa *scripting*.
- Dikembangkan oleh Netscape pada tahun 1995.
- Merupakan client side scripting karena eksekusi perintah-perintahnya dilakukan di sisi browser (*client*) bukan di web server.
- Dengan Java Script kita bisa dengan mudah membuat sebuah halaman web yang interaktif.
- Javascript  $\neq$  Java.



# Pengenalan Javascript (2)

Keuntungan menggunakan Javascript :

- Didukung oleh hampir semua browser.
- Membuat halaman web menjadi lebih interaktif.
- Memvalidasi data yang akan dikirim ke server.
- Free dan dapat digunakan siapa saja tanpa harus memiliki license.
- Tidak membutuhkan tools khusus, cukup menggunakan teks editor seperti halnya saat membuat HTML.



# Pengenalan Javascript (3)

Ciri-ciri Javascript :

- Menggunakan blok awal “ { ” dan diakhiri “ } ”.
- Extension file menggunakan “.js”
- Case Sensitive.
- Setiap statement diakhiri dengan “ ; ” tetapi bisa juga tidak.
- Jika script terlalu panjang dalam satu baris, dapat disambung ke baris selanjutnya dengan karakter “ \ ”



# Script Javascript (JS)



# Script JS

Ada beberapa cara untuk meletakkan kode Javascript didalam dokumen HTML :

- Internal dalam dokumen HTML.
- Menggunakan file ekstern.
- Melalui event tertentu.



# Script JS (internal)

⇒ Script diletakkan diantara tag `<script>` pada bagian header (head) dari dokumen HTML. Pemanggilan fungsi javascript diletakkan di bagian badan (body).

contoh :

```
<html>
<head>
  <title>Javascript Sederhana</title>
  <script language="javascript">
    document.write ("Belajar Javascript. ");
  </script>
</head>
<body>Pemrograman Web</body>
</html>
```





# Script JS (ekstern)

⇒ Membuat kode program Javascript dalam file tersendiri dan kemudian dipanggil dari dokumen HTML.

contoh :

```
<script language="javascript" src="kode.js">  
</script
```



# Script JS (event handler)

⇒ Melalui suatu perintah yang dilakukan ketika suatu event dilakukan oleh user. Contohnya saat meng-klik tombol submit/button.

contoh :

```
<input type="button" value="OK"  
  onclick="alert( 'OK' )">
```



# Komentar pada JS

⇒ Seperti bahasa pemrograman atau scripting lainnya, pada Javascript juga dapat dituliskan sebuah komentar dimana script komentar ini tidak akan dieksekusi sebagai program. Komentar satu baris diawali dengan “//”, komentar banyak baris diapit oleh “/\*” dan “\*/”.

contoh :

```
<script language="javascript">  
/* ini adalah komentar dalam javascript yang  
tidak akan dieksekusi */  
document.write ("Belajar Javascript. ");  
</script>
```



# Tag <noscript>

⇒ Tag ini digunakan untuk menampilkan pesan pada browser yang tidak mendukung Javascript. Jika browser sudah mendukung Javascript, maka tag ini akan diabaikan dan isinya tidak akan ditampilkan.

contoh :

```
<NOSCRIPT>
```

```
Browser Anda tidak mendukung Javascript
```

```
</NOSCRIPT>
```

```
<SCRIPT language="javascript">
```

```
document.write ("Browser Anda sudah mendukung Java Script  
dengan baik")
```

```
</SCRIPT>
```



# Operator pada Javascript



# Operator pada Javascript

- Operator Matematika.
- Operator Assignment.
- Operator Pembanding.
- Operator Logika.



# Operator Matematika

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



# Operator Assignment

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





# Operator Perbandingan

Operator	Penjelasan
==	Sama dengan
!=	Tidak sama dengan
>	Lebih besar dari
<	Lebih kecil dari
>=	Lebih besar atau sama
<=	Lebih kecil atau sama



# Operator Logika

Operator	Penggunaan
<b>&amp;&amp;</b>	and
<b>  </b>	or
<b>!</b>	not



# Variabel, Event Handler



# Variabel

⇒ Adalah suatu objek yang berisi data yang digunakan saat pengekseskusion program.

Kriteria penamaan variabel :

- Terdiri dari huruf, angka dan karakter “ \_ ”
- Harus dimulai oleh huruf atau karakter “ \_ ”
- Tidak boleh mengandung spasi.
- Tidak boleh memakai nama-nama yang telah di *reserved* oleh program (*abstract, boolean, break, byte, case, catch, char, dll*).



# Deklarasi Variabel

Mendeklarasikan variabel dapat dilakukan dengan dua cara :

- Eksplisit → dengan menuliskan kata kunci var kemudian diikuti dengan nama & nilai variabel

contoh : `var nama = "budi"`

- Implisit → dengan langsung menuliskan nama variabel dan diikuti oleh nilai variabel

contoh : `nama = "budi"`



# Event Handler

⇒ Suatu perintah yang dilakukan ketika suatu aksi (event) dilakukan oleh user. Event handler tidak perlu dimulai dengan tag `<script>` tapi dituliskan langsung di dalam script HTML.

contoh :

```
<input type="button" value="OK"  
onclick="alert('OK')">
```



# Event Handler

Event	Keterangan
<b>onMouseover</b>	User meletakkan kursor mouse di atas suatu elemen
<b>onFocus</b>	User memberikan fokus pada suatu elemen
<b>onBlur</b>	Elemen kehilangan fokus, user melakukan klik diluar elemen
<b>onChange</b>	User melakukan modifikasi isi dari data pada suatu field
<b>onClick</b>	User melakukan klik terhadap elemen
<b>onLoad</b>	Ketika browser memanggil/load suatu halaman
<b>onMouseout</b>	Kursor mouse meninggalkan posisinya
<b>onKeyPress</b>	Saat mengetikkan sesuatu pada keyboard



# Struktur Kendali & Fungsi





# Struktur Kendali

- Instruksi IF
- Switch Case
- Perulangan FOR
- Perulangan WHILE



# Instruksi IF

```
if (condition1)
{
    code to be executed if condition1 is true
}
else if (condition2)
{
    code to be executed if condition2 is true
}
else
{
    code to be executed if condition1 and
condition2 are not true
}
```



# Switch Case

```
switch(n)
{
case 1: execute code block 1
    break;
case 2: execute code block 2
    break;
default: code to be executed if n is
different from case 1 and 2
}
```



# Perulangan FOR

```
for (var=startvalue; var<=endvalue;  
var=var+increment)  
{  
code to be executed  
}
```



# Perulangan WHILE

```
while (var<=endvalue)
{
  code to be executed
}
```



# Fungsi

⇒ Subprogram yang memungkinkan untuk menjalankan sekelompok instruksi tertentu dengan melakukan pemanggilan terhadap nama fungsi tersebut.

```
function functionname(var1,var2,...,varX)  
{  
some code  
}
```



*Any Question?  
See you next time..*