### **JAVASCRIPT**

### Pengantar JavaScript

 JavaScript digunakan pada Web pages untuk meningkatkan design, validate forms, detect browsers, create cookies, GUI dsb

- JavaScript adalah bahasa scripting yang paling populer di internet dan bekerja pada banyak browser seperti Internet Explorer, Mozilla, Firefox, Netscape, Opera.
- Menggunakan sintaks seperti C++/Java
- Lihat di <a href="http://www.w3schools.com/js/">http://www.w3schools.com/js/</a>

## Apa itu JavaScript?

- JavaScript dirancang untuk menambah interaktif HTML pages
- JavaScript adalah bahasa scripting (bahasa scripting adalah sebuah lightweight programming language)
- JavaScript terdiri dari baris-baris code executable computer
- JavaScript biasanya embedded secara langsung pada HTML pages
- JavaScript adalah interpreted language (artinya bahwa scripts dijalankan tanpa di kompile terlebih dahulu)
- Setiap orang dpt menggunakan JavaScript tanpa harus membeli license

## Apakah Java dan JavaScript sama?

 Java dan JavaScript adalah dua bahasa yang berbeda baik dari sisi konsep maupun dari sisi desain

 Java (dikembangkan oleh Sun Microsystems) adalah sebuah bahasa pemrograman yang powerful & sangat kompleks – sama dengan C & C++.

# Apa yang dapat dilakukan JavaScript?

- JavaScript dapat bereaksi terhadap events JavaScript dapat di-set untuk menjalankan saat terjadi sesuatu, seperti sebuah page telah selesai dipanggil atau saat seorang user meng-klik pada HTML element
- JavaScript dapat membaca dan menulis HTML elements JavaScript dapat membaca dan mengubah isi dari HTML element
- JavaScript dapat digunakan untuk mem-validasi data JavaScript dapat digunakan untuk mem-validasi form data sebelum di-submitted ke server, hal ini akan mengamankan server dari pemrosesan extra
- JavaScript dapat digunakan untuk mendeteksi browser pengunjung JavaScript dapat digunakan untuk mendeteksi browser pengunjung dan – memanggil page lain yang secara specifik didesain untuk browser tersebut
- JavaScript dapat digunakan untuk membuat cookies JavaScript dapat digunakan untuk menyimpan dan memanggil informasi di komputer pengunjung

### Pemrograman Client-Side

- HTML cocok untuk mengembangkan static pages
  - Dapat digunakan membuat text/image layout, presentation, links, ...
  - Web page tampak sama setiap kali diakses
  - Untuk mengembangkan interactive/reactive pages, harus diintegrasikan dengan programming
- Pemrograman client-side
  - Programs ditulis dalam bahasa pemrograman yang terpisah contoh : JavaScript, JScript, VBScript
  - Programs diembedded pada HTML Web page, untuk mengidentifikasi komponen program
     Contoh: <script type="text/javascript"> ... </script>
  - Untuk menjalankan program guna menampilkan page digunakan browser, yang mengintegrasikan dynamic output dengan static content dari HTML

### Scripts vs. Programs

- Bahasa scripting adalah simple, termasuk bahasa pemrograman interpreted scripts di-embedded sebagai plain text, interpreted by application
  - simpler execution model: tidak membutuhkan compiler / development environment
  - saves bandwidth: source code di-download, tidak mengcompile executable
  - platform-independence : kode di terjemahkan oleh banyak script-enabled browser
  - Tetapi : lebih lambat daripada compiled code, tidak powerful/full-featured

JavaScript: adalah Web scripting language pertama, yang dikembangkan oleh Netscape pada tahun 1995. Mirip dengan Java/C++, tetapi lebih sederhana dan lebih fleksibel (loose typing, dynamic variables, simple objects)

JScript: adalah JavaScript versi Microsoft, diperkenalkan tahun 1996 core language yang sama, tetapi beberapa browser-specific seperti, IE & Netscape umumnya dapat menangani JavaScript & JScript

VBScript: adalah client-side scripting versi Microsoft Visual Basic

## Common Scripting Tasks

- Menambahkan dynamic features pada Web pages
  - validation of form data
  - time-sensitive or random page elements

- image rollovers
- handling cookies

- Mendefinisikan programs dengan Web interfaces
  - Menggunakan buttons, text boxes, clickable images, prompts, frames
- Keterbatasan dari client-side scripting
  - Karena script code di-embedded dalam page, maka script dapat dilihat
  - Untuk alasan keamanan, apa yang dapat dilakukan oleh script dibatasi
     Contoh: tidak dapat mengakses hard drive client
  - Karena dirancang untuk berbagai platform, maka script tidak berisi perintah khusus untuk setiap platform
  - Script languages tidak full-featured
    - Contoh : Objek JavaScript tidak cocok untuk pengembangan proyek yg besar

### Sintaks JavaScript

- Program JavaScript dituliskan pada file HTML (.html atau .htm) menggunakan tag <SCRIPT>
- Output dari program JavaScript ditampilkan secara langsung seolah-oleh dimasukkan dalam HTML

```
<HTML>
<HEAD>
<SCRIPT language="javascript">
<!--
// Build HTML here
<!-- akhir script -->
</SCRIPT>
</HEAD>
</HTML>
```

# Contoh Program JavaScript

```
<html>
<!-- COMP519 js01.html 7.09.2005 -->
<head>
 <title>JavaScript Page</title>
</head>
<body>
 <script type="text/javascript">
    // silly code to demonstrate output
   document.write("<FONT COLOR='GREEN'>
                  Hello world!</FONT>");
   document.write("How are <br/> +
                  "<i>you</i>?");
 </script>
 Here is some static text as well.
 </body>
</html>
```

document.write menampilkan teks dalam page

Teks yang ditampilkan dapat termasuk HTML tags

tags diterjemahkan oleh browser saat teks ditampilkan

Seperti pada C++/Java, statement diakhiri dengan ;

Komentar pada JavaScript sama dengan C++/Java

```
// awal satu baris komentar
/*...*/ akhir dari beberapa baris komentar
```

### Tipe Data dan Variabel JavaScript

JavaScript hanya mempunyai 3 tipe data primitive

```
      String
      "I said 'hi'."
      ""

      Number
      12
      3.14159
      1.5E6

      Boolean
      true
      false
```

```
<html>
<!-- COMP519 is02.html 07.09.2005 -->
<head>
  <title>Data Types and Variables</title>
</head>
<body>
  <script type="text/javascript">
   var x, y;
   x = 1024;
   y=x; x = "foobar";
   document.write("<p>x = " + y + "</p>");
    document.write("<p>x = " + x + "</p>");
  </script>
</body>
</html>
```

```
Inisialisasi seperti pada C++/Java
  message = "howdy";
  pi = 3.14159;
```

Nama variable terdiri dari letters, digits, dan underscores: diawali dengan *letter* 

Nama variables adalah case-sensitive

you don't have to declare variables, will be created the first time used, but better if you use var statements

```
var message, pi=3.14159;
```

### Operator & Statement Kontrol

```
<html>
<!-- COMP519 js03.html 7.09.2005 -->
<head>
  <title>Folding Puzzle</title>
</head>
<body>
 <script type="text/javascript">
    distanceToSun = 93.3e6*5280*12;
    thickness = .002i
   foldCount = 0;
   while (thickness < distanceToSun) {</pre>
        thickness *= 2;
        foldCount++;
    document.write("Number of folds = " +
                   foldCount);
  </script>
</body>
</html>
```

Operator standard dan statement kontrol pada C++/Java sama seperti pada JavaScript

```
+, -, *, /, %, ++, --,
...
==, !=, <, >, <=, >=
&&, | |, !, ===, !==
```

- if-then, if-then-else, switch
- while, for, do-while, ...

### Operator & Statement Kontrol

Script di bawah ini adalah contoh penggunaan if - else.



### Operator & Statement Kontrol

```
Penggunaan if dapat digantikan dengan "switch".
   <SCRIPT LANGUAGE="JavaScript">
   function tanya()
          var jawab=" "
          var jawab=prompt("Anda senang bikin hompej ?")
          var JAWAB = jawab.toUpperCaše()
          switch(JAWAB)
   case "YA":
                    var tampil="Sama dong, saya juga :D"
                    break:
   case "YUP":
                    var tampil="Sama dong, saya juga :D"
                    break:
   case "TIDAK":
                    var tampil="Iya euy, bikin suntuk tuh ( )Zz.."
                    break:
   case "NGGAK":
                    var tampil="Iya euy, bikin suntuk tuh (_ _)Zz.."
                    break:
                    var tampil="Jawab dong, ya/yup atau tidak/nggak"
   default:
                    break:
   alert(tampil)
   </SCRIPT> <b>Hobi</b>
   <FORM> <INPUT TYPE="button" VALUE="polling" onClick="tanya()"> </FORM>
```

### Loop For dan While

Adalah perintah untuk melakukan iterasi (loop) atau melakukan pengulangan.

```
<b>Kita cuman menghitung dari 1 sampai 7:</b><br><script language="JavaScript">
for (i=1; i<=7; i=i+1)
{
          document.write("Ini angka " + i + "<BR>");
}
</SCRIPT>
```

Script di atas hanya akan memberikan hasil seperti di bawah ini.

#### Kita cuman menghitung dari 1 sampai 7:

```
Ini angka 1
Ini angka 2
Ini angka 3
Ini angka 4
Ini angka 5
Ini angka 6
Ini angka 7
```

### Loop For dan While

Sedangkan contoh penggunaan while untuk output yang sama adalah sebagai berikut :

```
<SCRIPT LANGUAGE="JavaScript">
iterasi=7
i=1
while (i <= iterasi)
{
    document.write("Ini angka " + i + "<BR>");
    i=i+1
}
</SCRIPT>
```

Hasilnya seperti di bawah ini.

```
Ini angka 1
Ini angka 2
Ini angka 3
Ini angka 4
Ini angka 5
Ini angka 6
Ini angka 7
```

### Memonitor User Events

Bermacam-macam onXxx Attributes

- onClick - onBlur

- onLoad - onSubmit

- onMouseOver - onSelect

- onMouseOut - onFocus

### **Event Handler on Click**

- Biasanya digunakan untuk menampilkan pesan singkat jika seseorang melakukan tindakan tertentu. Dengan pesan tersebut, maka user akan tahu akibat dari tindakannya.
- Misal peringatan dengan Tombol. Penekanan dengan tombol sering digunakan untuk memberikan efek interaktif dengan user

```
<HTMI > <HFAD>
<TITLE>Simple JavaScript Button</TITLE>
<SCRIPT TYPE="text/javascript">
   <l--
Function dontClick()
   alert("I told you not to click !");
// -->
</SCRIPT> </HFAD>
<BODY BGCOLOR="WHITE">
<H1>Simple JavaScript Button</H1>
<FORM>
   <INPUT TYPF="BUTTON"</pre>
          VALUE="Don't Click Me" on Click="alert('hey...I said don't click me'); return value">
</FORM>
</BODY></HTML>
```

### Event Handler onMouseOver

Event handler onMouseOver ini gunanya untuk mengatur apa yang akan terjadi saat mouse kita gerakkan ke atasnya.

```
<a href="http://xxx.com" onMouseOver="window.status='Ke Menu Utama'; return true"> Klik di sini</a>
```

Dengan menggunakan onMouseOver juga dapat mengubah property lain. Misalnya akan mengubah warna latar dokumen, yaitu dengan menggunakan document.bgColor.

```
<a href="http://xxx.com" onMouseOver="document.bgColor= '#ffcc99'; return true"> Klik di sini</a>
```

Untuk menggabung dua efek di atas, perubahan window.status dan document.bgColor - hmm tidak terlalu sulit, cukup memisahkan kedua efek tersebut dengan koma (,) seperti script berikut. Perintah onMouseOver yang kedua berada di antara tanda petik (") untuk membuat efeknya terjadi secara simultan.

```
<a href="http://xxx.com" onMouseOver="document.bgColor='#ffcc99', onMouseOver=window.status='Ke Menu Utama'; return true">Klik di sini</a>
```

### Event Handler onMouseOver & onMouseOut

• Untuk membuat gambar yang bisa berubah saat didekati mouse, cukup menggunakan script sederhana seperti di bawah ini.

```
<a href="http://xxx.com/" onMouseOver="document.gambarku.src='gambar1.gif'" onMouseOut="document.gambarku.src='gambar2.gif'"> <IMG SRC="gambar2.gif" BORDER=0 NAME="gambarku"></a>
```

Saat mouse mendekati gambar, onMouseOver aktif, sehingga gambar yang muncul adalah gambar2.gif.

Sedang saat mouse menjauh, onMouseOut aktif, dan gambar yang muncul adalah gambar1.gif.

### Event Handler on Focus dan on Blur

Event handler onFocus ini bekerja saat user terfokus pada sebuah item.

```
<FORM>
<INPUT TYPE="text" SIZE="30" onFocus="window.status='Anda sekarang siap mengisi kotak';">
</FORM>
```

Event handler onBlur adalah kebalikan dari onFocus.

### Event Handler on Select dan on Submit

Event handler ini bekerja saat user memilih (memblok) isi kotak.
 <FORM>

 <INPUT TYPE="text" SIZE="40" VALUE="Tulis nama anda, kemudian pindahkan kursor dari kotak ini" onSelect="alert('Hehehe, gagal deh ngeblok');">
 <IFORM>

 Event Handler onSubmit, bekerja saat user mengklik tombol submit.
 <FORM onSubmit="alert('Anda yakin mau mengirim data ?')";>

 <INPUT TYPE="submit Query">
 <IPORM>

Submit Query

#### **User-Defined Classes**

- Dapat mendefinisikan class baru, tetapi dengan notasi yang awkward
  - Pendefinisian sederhana sebuah fungsi yang bertindak sebagai constructor
  - Menentukan data fields & methods menggunakan this
  - Tidak ada data yang disembunyikan : tidak dapat melindungi data atau methods

define **Die** function (i.e., constructor)

inisialisasi data fields pada fungsi, diawali dengan this

similarly, assign method to separately defined function (which uses this to access data)

## Object dan Class

- Dapat menggunakan Notasi Literal
  - Objek dat dibuat menggunakan notasi "literal" pada form { field1:val1, field2:val2, ..., fieldN:valN }
  - Sebagai contoh, berikut ini memberikan nilai equivalent pada object1 dan object2

```
var object1 = new Object();
object1.x =3;
object1.x =4;
object1.x =5;

object2.x = { x:3, y:4, z:5};
```

### Object dan Class

#### Methods adalah Function-Valued Properties

- Tidak ada sintaks khusus untuk mendefinisikan method dari object
- Berikutnya adalah property yang dimiliki oleh document yaitu seperti pada script berikut:

```
<SCRIPT LANGUAGE="javascript">
var bgc = document.bgColor;
                                      var fgc = document.fgColor;
var lc = document.linkColor:
                                      var al = document.alinkColor:
                                      var url = document.location:
var vlc = document.vlinkColor:
var ref = document.referrer;
                                      var t = document.title;
var lm = document.lastModified:
document.write("Warna latar (background color) halaman ini <B>" +bgc+ "</B>.")
document.write("<BR>Warna teksnya (mmm foreground color) <B>" +fgc+ "</B>.")
document.write("<BR>Warna link adalah <B>" +lc+ "</B>.")
document.write("<BR>Kalau warna link yang aktif <B>" +al+ "</B>.")
document.write("<BR>Dan warna link yang telah dikunjungi <B>" +vlc+ "</B>.")
document.write("<BR>Alamat URL halaman ini <B>" +url+ "</B>.")
document.write("<BR>Halaman yang anda lihat sebelum melihat halaman ini <B>" +ref+ "</B>.")
document.write("<BR>Judul halaman ini (title lah kalau nggak ngerti judul) <B>" +t+ "</B>.")
document.write("<BR>Dokumen ini terakhir diedit: <B>" +lm+ "</B>.")
</SCRIPT>
```

### Object dan Class

Script di atas akan memberikan hasil berikut:

Warna latar (background color) halaman ini #ffffff.

Warna teksnya (mmm foreground color) #000000.

Warna link adalah #0000ff.

Kalau warna link yang aktif #0000ff.

Dan warna link yang telah dikunjungi #800080.

Alamat URL halaman ini file:///F:/14.Ngajar/Web%20Programming/javascript4.htm.

Halaman yang anda lihat sebelum melihat halaman ini .

Judul halaman ini (title lah kalau nggak ngerti judul) Tutorial - Java Script.

Dokumen ini terakhir diedit: 08/29/2006 10:29:36.

Script di atas menjelaskan tentang kegunaan tiap property dari document.

## Interactive Pages menggunakan Prompt

```
<html>
<!-- COMP519 is05.html 08.09.2005 -->
<head>
  <title>Interactive page</title>
</head>
<body>
 <script type="text/javascript">
    userName = prompt("What is your name?", "");
   userAge = prompt("Your age?", "");
    userAge = parseFloat(userAge);
    document.write("Hello " + userName + ".")
    if (userAge < 18) {
      document.write(" Do your parents know " +
                     "you are online?");
</script>
  The rest of the page...
</body>
</html>
```

1<sup>st</sup> argument: prompt message tampak pada dialog box

2<sup>nd</sup> argument: default value akan muncul pada kotak

3<sup>rd</sup> Fungsi mengembalikan nilai yang dimasukkan oleh user ke dalam dialog box

Jika value adalah sebuah number, harus menggunakan parseFloat untuk mengubahnya

### Deklarasi Fungsi

- Function dideklarasikan menggunakan reserved word
- Nilai yang dikembalikan tidak dideklarasikan, begitu juga tipe dari argumentasi

### **User-Defined Functions**

- Pendefinisian fungsinya sama seperti pada C++/Java, kecuali :
  - Tidak ada return type untuk fungsi (karena variabelnya adalah loosely typed)
  - Tidak ada types untuk parameters (karena variabelnya adalah loosely typed)
  - by-value parameter passing only (parameter gets copy of argument)

```
function isPrime(n)
// Assumes: n > 0
  Returns: true if n is prime, else false
  if (n < 2) {
    return false;
 else if (n == 2) {
    return true;
 else {
      for (var i = 2; i \le Math.sqrt(n); i++)
        if (n % i == 0) {
          return false;
      return true;
```

#### can limit variable scope

Jika penggunaan pertama dari variable diawali dengan var, maka variabel tersebut adalah local pada fungsi

Untuk modularity, sebaiknya dibuat semua variabel ada pada sebuah fungsi local

### Contoh Fungsi

```
<ht.ml>
<!-- COMP519 js06.html 08.09.05 -->
<head> <title>Prime Tester</title>
  <script type="text/javascript">
    function isPrime(n)
    // Assumes: n > 0
    // Returns: true if n is prime
      // CODE AS SHOWN ON PREVIOUS SLIDE
  </script> </head>
<body>
 <script type="text/javascript">
    testNum = parseFloat(prompt("Enter a positive integer", "7"));
    if (isPrime(testNum)) {
     document.write(testNum + " <b>is</b> a prime number.");
    else {
     document.write(testNum + " <b>is not</b> a prime number.");
  </script>
</body> </html>
```

Definisi fungsi dimulai pada HEAD

HEAD dipanggil pertama, jadi fungsi didefinisikan sebelum code dalam BODY dijalankan

#### **Date Class**

- String & Array adalah class yg paling sering digunakan pada JavaScript
  - special purpose classes & objects juga ada
- Date class dapat digunakan untuk mengakses date dan time
  - Utk membuat Date object, gunakan new & supply year/month/day/... yg diinginkan

methods yang termasuk adalah :

```
newYear.getYear()
newYear.getMonth()
newYear.getDay()
newYear.getHours()
newYear.getMinutes()
newYear.getSeconds()
newYear.getMilliseconds()
dapat mengakses komponen
individual dari sebuah date
```

### Contoh Date (1)

Script kedua yang akan kita coba adalah menampilkan waktu seperti di bawah ini: Hari ini 3-9-2006, jam: 7:16.40

Untuk menampilkan waktu seperti di atas kita dapat menggunakan script berikut:

```
< html>
<!-- COMP519 is11.html 09.09.2005 -->
<head> <title>Time page</title> </head>
   <body>
 Time when page was loaded:
<SCRIPT LANGUAGE="JavaScript"> //Script tentang waktu
       Sekarang = new Date();
       document.write("Hari ini " + Sekarang.getDate() + "-" +
       (Sekarang.getMonth()+1)+ "-" + Sekarang.getFullYear() + ",
       jam: " + Sekarang.getHours() + ":" + Sekarang.getMinutes() +
       "." + Sekaranq.qetSeconds())
</SCRIPT>
</body> </html>
```

### Contoh Date (2)

```
<html>
<!-- COMP519 is11.html 09.09.2005 -->
<head> <title>Time page</title> </head>
<body>
 Time when page was loaded:
 <script type="text/javascript">
   now = new Date();
   document.write("" + now + "");
   time = "AM";
   hours = now.getHours();
   if (hours > 12) {
       hours -= 12;
       time = "PM"
   else if (hours == 0) {
       hours = 12i
   document.write("" + hours + ":" +
                  now.getMinutes() + ":" +
                  now.getSeconds() + " " +
                  time + "");
 </script>
</body> </html>
```

Secara default, date akan ditampilkan secara penuh,

```
Sun Feb 03 22:55:20 GMT-0600 (Central Standard Time) 2002
```

Dapat pula hanya menampilkan sebagian dari date yang diinginkan menggunakan methods

here, determine if "AM" or "PM" and adjust so hour between 1-12

10:55:20 PM

### Contoh Class

```
<html>
             js15.html 09.09.2005 -->
<!-- COMP519
<head>
 <title>Dice page</title>
 <script type="text/javascript"</pre>
        src="Die.is">
 </script> </head>
<body>
<script type="text/javascript">
   die6 = new Die(6);
   die8 = new Die(8);
   roll6 = -1; // dummy value to start loop
   roll8 = -2; // dummy value to start loop
   while (roll6 != roll8) {
     roll6 = die6.Roll();
     roll8 = die8.Roll();
     document.write("6-sided: " + roll6 +
                    "   " +
                    "8-sided: " + roll8 + "<br />"); }
   document.write("<br />Number of rolls: " +
                  die6.numRolls);
 </script>
</body> </html>
```

create a Die object using new (similar to String and Array)

here, the argument to Die initializes numSides for that particular object

each Die object has its own properties (numSides & numRolls)

Roll(), when called on a particular Die, accesses its numSides property and updates its NumRolls

### JavaScript untuk membuat Page Dynamic

#### Membuat Password

- Password adalah system pengamanan untuk pencegahan hal-hal yang tidak diinginkan dari pihak luar.
- Penerapan password dapat dilakukan dengan berbagai cara.
- Misalnya dari jumlah karakter password yang harus terdidi dari jumlah karakter tertentu, password dengan enkripsi atau pembatasan pemasukan password
- Password terbatas untuk membatasi proses pengisian password yang berulang kali

# JavaScript untuk membuat Page Dynamic

```
<HTML> <SCRIPT language="JavaScript">
Function InputPassword()
   coba=1;
   var Pass
                    Pass=prompt('Masukkan password !')
   do
                    if (Pass=="Hanya Aku Yang Tahu")
                              alert("Hai..Selamat datang !!");
                              window.open('coba1.htm');
                              break;
                              alert("Password nggak cocok !! Ulangi lagi !!");
          else
                              if (coba==3)
   alert("Maaf sudah 3x !!");
   history.go(-1);
   coba=coba+1; }
While (coba<=3)
</script> <form method="POST">
<input type=button value="Password" name="B3" onClick="Input Password()"> 
</form> </html>
```

### Membuat Form Interaktif (1)



### Membuat Form Interaktif (2)

Cari pakai Google:	Cari!!
odn pakar doogiot	

<INPUT TYPE="submit" VALUE="Cari .. !!"> </FORM>