

Pengenalan Javascript

A. Apa Itu Javascript

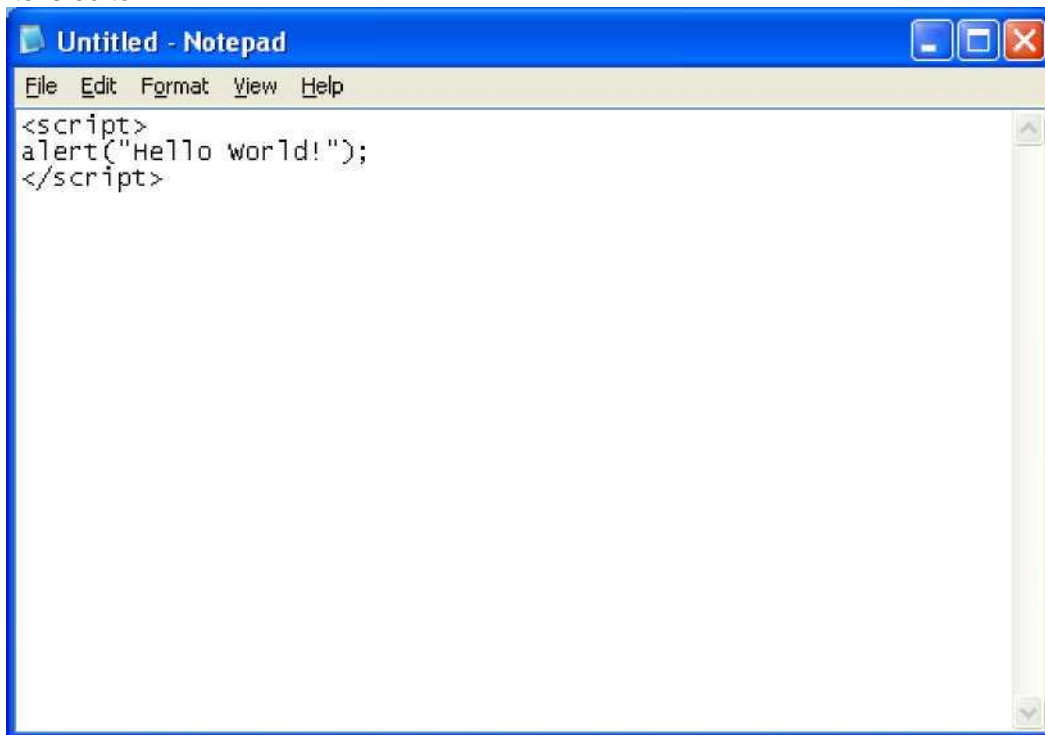
Javascript adalah bahasa script yang biasa jalan di browser, orang-orang biasa bilang client side programming. Client di sini adalah browser, seperti: Internet Explorer, Firefox, Safari dan sebagainya. Kode javascript biasanya disisipkan diantara kode-kode HTML.

Di mana saya bisa menulis kode javascript? Anda bisa menulis kode javascript di teks editor seperti notepad dan sebagainya.

Apakah saya butuh compiler untuk menjalankan javascript? Tidak perlu, anda cukup menjalankan javascript menggunakan browser. Semua browser mempunyai engine yang menginterpretasikan kode javascript kita.

B. Memulai Javascript

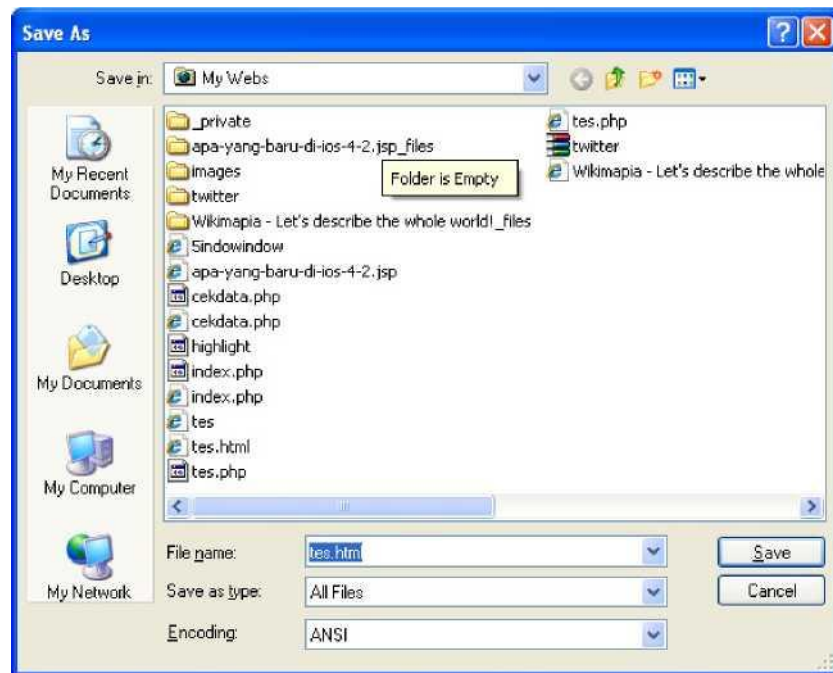
Oke sekarang mari kita coba mulai menulis kode javascript kita. Di sini kita menggunakan notepad sebagai teks editor.



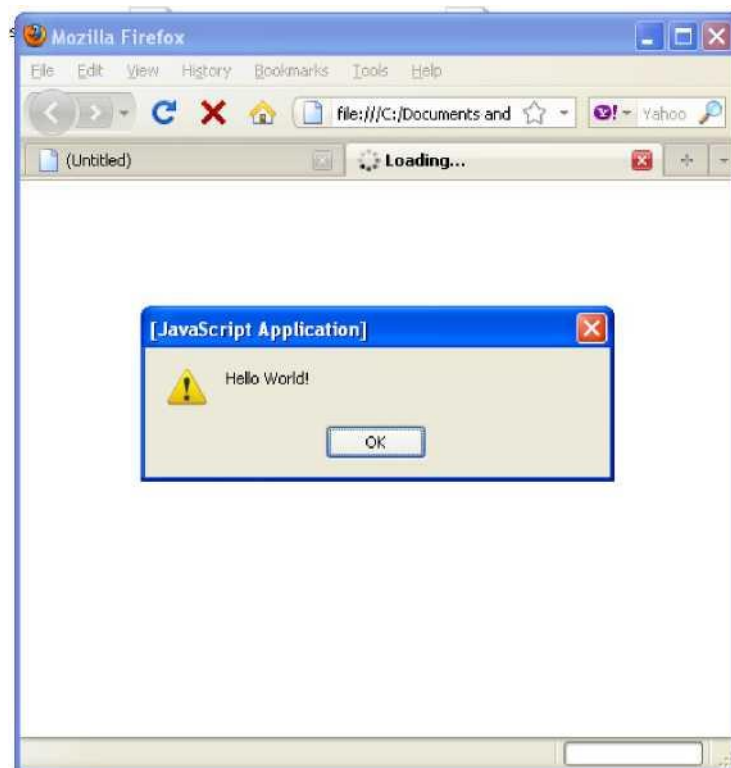
Kode javascript ditulis diantara tag `<script>` dan `</script>`, bisa kita sisipkan di kode-kode HTML kita.

Sekarang mari kita simpan kode tersebut dengan **File Name:** tes.html, jangan lupa **Save as type:** All

Files, seperti gambar di bawah.



Untuk mengetesnya, anda tinggal double klik file tes.html yang kita buat tadi, hasilnya akan seperti berikut:



Javascript bisa juga ditulis terpisah, filenya diberi ekstension .js Contoh: **namafile.js**, nanti cara menyisipkannya di file html adalah seperti berikut:

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

C. Sintaks Javascript

Sintaks javascript mirip-mirip bahasa C atau java. Javascript **bersifat case sensitive**, artinya huruf kecil dan huruf besar adalah berbeda.

Setiap baris kode javascript dipisahkan baris baru atau bisa juga titik koma (;)

Komentar dalam javascript diawali dengan // atau ditulis diantara /* dan */

D. Variabel

Variabel di javascript bisa ditulis dengan diawali huruf atau underscore (_) atau tanda dollar (\$).

Contoh : jumlah_hits , _nama

Deklarasi variable

- Anda bisa mendeklarasi dengan menggunakan var, contoh: `var x = 5`, ini bersifat local dan global (bisa di akses oleh semua fungsi)
- Atau langsung deklarasikan tanpa var, `x = 5`

Contoh:

```
<script> var x = 5;  
var nama = "Desrizal"; document.write(nama);  
</script>
```

E. Operator

Operator Aritmatika

Yaitu operator untuk operasi matematika

Operator	Definisi	Contoh
+	Penambahan	5 + 5 = 10
-	Pengurangan	6 - 2 = 4
*	Pengalian	4 * 4 = 16
/	Pembagian	8 / 2 = 4
%	Modulus (sisa hasil pembagian)	5 % 2 = 1 10 % 4 = 2

Contoh:

```
<script> var x =  
4; var y = 2;  
  
z = x + y;  
  
alert (z);  
</script>
```

Operator Assignment

Seperti kebanyakan bahasa pemrograman lainnya, untuk memberikan nilai kepada suatu variable menggunakan tanda sama dengan =

Berikut adalah penyingkatan penulisan operator

Shorthand Operator	Artinya
x += y	x = x + y
x -= y	x = x - y
x *= y	x = x * y
x /= y	x = x / y

Contoh:

```
<script> var x = 4; var y = 2; x -= y alert(x); </script>
```

Operator Pembandingan

Berguna untuk membanding nilai suatu variable

Operator	Definisi	Contoh
==	Sama dengan	var1 == "Desrizal"
!=	Tidak sama dengan	x != y
>	Lebih besar dari	x > y
<	Lebih kecil dari	x < 6
>=	Lebih besar sama dengan	x >= y
<=	Lebih kecil sama dengan	x < 5

Contoh:

```
<html>  
<head>  
<script> var  
x = 4; var y  
= 2;  
  
if(x > y){  
    alert("x lebih besar dari y");  
}
```

```

</script>
</head>
<body>
</body>
</html>

```

Operator Logika

Operator	Definisi	Contoh
&&	DAN	<code>x>=5 && x<10</code>
	ATAU	<code>x == 6 x == 12</code>
!	BUKAN	<code>!Expression</code>

Contoh:

```

<script> var
x = 76;
if(x >= 80){
    alert("Nilai = A");
}else if(x >= 70 && x < 80) {
    alert("Nilai = B");
}else if(x >= 60 && x < 70) {
    alert("Nilai = C");
}else{
    alert("Nilai = D");
}
</script>

```

F. Statement

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

Jika anda membutuhkan kondisi yang banyak

Sintaks :

```
if(kondisi 1) {  
    kode yang dijalankan j      kondisi 1 benar  
}else if(kondisi 2){      ika  
    kode yang dijalankan j      kondisi 2 benar  
}else if(kondisi 3){      ika  
    kode yang dijalankan j      kondisi 3 benar  
}else{  
    ika  
    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        yangdijalankan j    kondisi1 benar;  
    break;                ika  
    case kondisi2  
    kode        yangdijalankan j    kondisi2 benar;  
    break;                ika  
    case kondisi3  
        kode yangdijalankan j    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>
```

Pengulangan

for

Berguna untuk pengulangan yang sudah ditentukan terlebih dahulu awalnya berapa, dijalankan sampai kondisi bagaimana.

Sintaks :

```
for(awal; kondisi; penambahan){ 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>
```

G. Fungsi

Apa itu fungsi? Fungsi adalah kumpulan blok kode yang membentuk fungsi tersendiri. Kita bisa membuat sendiri fungsi tersebut

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>
```

Penanganan Event

A. Apa itu 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

B. Contoh-contoh Penanganan Event

1. onclick

Pada contoh dibawah kita membuat tombol, yang kalau diklik akan muncul window alert <script>

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

```
<form>
```

```
<html>
```

```
<body>
```

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

```
<input type="radio" name="C1"
```

```
<input type="radio" name="C1"
```

```
<input type="radio" name="C1"
```

```
</form>
```

```
</body>
```

```
</html>
```

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

Pada contoh berikut jika kita mengklik radio button akan mengubah warna latar halaman web

```
onclick="document.bgColor='lightblue'">
onclick="document.bgColor='lightyellow'">
onclick="document.bgColor='lightgreen'">
```

2. onload

Event onload akan dieksekusi jika suatu objek telah diloat, pada contoh berikut kita menaruh event onload di tag <body>, artinya jika halaman web sudah diloat 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>
```

Objek String

A. Memformat teks dengan Javascript

Untuk membantu melakukan formatting terhadap teks secara programming, javascript menyediakan beberapa metode formatting menggunakan javascript.

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> surrounding it.
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

Contoh 1:

```
<script>
var teks = "Desrizal"

panjang = teks.length;

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

Hasilnya:



Contoh 2: <script>

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

Output:

Welcome to our site!

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

Output:

WELCOME TO OUR SITE!

Contoh 3: `<script>`

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

Output:

wELC oME TO oUR sITE!

B. Penanganan Objek String

Setelah kita mempelajari metode string yang berkaitan dengan format teks. Sekarang kita lanjut metode objek string lainnya yang bertugas menangani dan memanipulasi objek string dan contohnya.

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

Contoh-contoh:

1. Validasi email, pada contoh berikut kita akan mengecek apakah email berikut valid atau tidak.

Cara mengeceknya adalah apakah string yang diinput mengandung karakter @ atau titik (.) <form

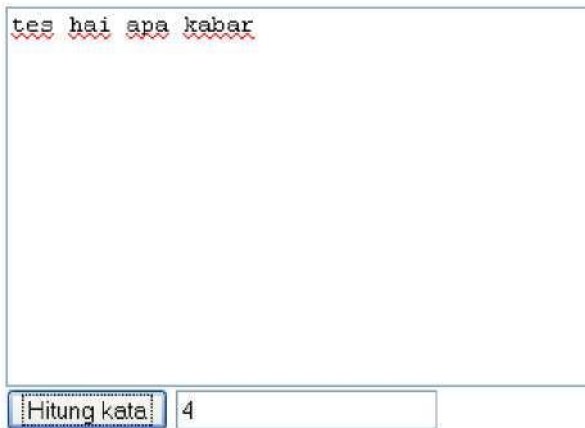
```
name="test" onSubmit="checkemail(this.test2.value);return false">
<input type="text" size=20 name="test2"> <input type="submit" value="Submit">
</form>
```

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

2. Menghitung kata, Pada contoh berikut kita akan menghitung kata, caranya adalah mengambil isi suatu textarea, lalu memotong-motong isinya dengan metode split.

```
<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/j avascript"> function hitung(){
    var formcontent=document.wordcount.wordcount2.value
    isi=formcontent.split(" ")
    document.wordcount.wordcount3.value=isi.length
}
</script>
```



C. Parse String ke Integer dan Float

parseInt, untuk mengubah string menjadi Integer

parseFloat, untuk mengubah string menjadi Float

```
<script type="text/j avascript">
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>
```

Objek Window

A. 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=3 0 0,height=2 0 0");
```

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	Set apakah bisa window diubah-ubah ukurannya	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>
```

B. Mereload, Menutup, Meloding Halaman Bam, 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: `urlkamu`

Untuk mengeprint halaman web `window.print()`

C. Komunikasi Antar Window

Adakalanya ketika kita membuka window baru, kita bisa 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>

```

D. Alert, Confirm dan Prompt

alert digunakan untuk menampilkan window alert

Contoh:

```

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

```

Hasil:



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

```

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

```



prompt, digunakan untuk meminta inputan melalui window

Contoh: `<script type="text/j avascript">
var y=window.prompt("Masukkan nama anda")
window.alert(y)
</script>`

Hasilnya



Array

A. Pengenalan Array

Apa sih array? Array adalah semacam variabel tunggal yang terindex secara teratur, jadi cukup satu nama variabel tapi bisa punya banyak nilai. Karena terindex secara teratur kita jadi gampang mengolah dan menampilkan nilai-nilai dari variabel tersebut.

Contoh bentuk array:

```
Buah[1] = "Rambutan"  
Buah[2] = "Durian"  
Buah[3] = "Manggis"
```

Bagaimana cara buat array di javascript?

Untuk membuat array di javascript, kita perlu mendeklarasikan variabel array dengan cara **new Array()**

Contoh:

```
<html>
```

```
<body>

<script type="text/j avascript">
var mobil = new Array()
mobil[0] = "Saab"
mobil[1] = "Volvo"
mobil[2] = "BMW"

for (i=0;i<mobil.length;i++){
    document.write(mobil[i] + "<br />")
}
</script>

</body>
</html>
```

Selain di atas kita juga bisa membuat array dengan cara menulis langsung di dalam argument Array(). Index atau Key array dimulai dari nol (0)

Contoh:

```
<html>
<body>
<script type="text/j avascript">
var buah = new Array("Mangga", "Rambutan", "Durian");

document.write(buah[1]);

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


Hasilnya adalah **Rambutan**

B. Metode untuk Manipulasi Array

Javascript menyediakan metode-metode untuk melakukan manipulasi terhadap objek array

Properti	Deskripsi
length	Menghasilkan jumlah elemen dalam suatu array

Metode Array

concat()

fungsi:

Menggabungkan dua atau lebih array

Sintaks:

```
arrayObject.concat(arrayX,arrayX,.....,arrayX)
```

Contoh:

```
<script type="text/j avascript">
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Tove"
arr[2] = "Hege"

var arr2 = new
Array(3) arr2[0] =
"John" arr2[1] =
"Andy" arr2[2] =
"Wendy"

var arr3 = new
Array(2) arr3[0] =
"Stale" arr3[1] =
"Borge"

document.write(arr.concat(arr2,arr3))

</script>
```

Hasilnya:

Jani,Tove,Hege,John,Andy,Wendy,Stale,Borge

join()

Fungsi:

Menuliskan elemen-elemen array ke dalam satu string dipisahkan oleh karakter yang kita

set Sintaks: `arrayObject.join(pemisah)`

Contoh:

```
<script type="text/j avascript">
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"

document.write(arr.join() + "<br />")

document.write(arr.join(".")) </script>
```

Hasilnya:

Jani,Hege,Stale
Jani.Hege.Stale

Pop()

Fungsi:

Untuk mendelete elemen terakhir dari suatu array Sintaks: `arrayObject.pop()`

Contoh:

```
<script type="text/j avascript">
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"

document.write(arr + "<br />") document.write(arr.pop() + "<br />")
document.write(arr)

</script>
```

Hasilnya:

Jani,Hege,Stale
Stale

Jani,Hege

push()

Fungsi:

Menambah satu atau lebih elemen ke bagian akhir suatu array

Sintaks:

```
arrayObject.push(newelement1,newelement2,....,newelementX)
```

Contoh:

```
<script type="text/j avascript">
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"

document.write(arr + "<br />")
document.write(arr.push("Kai Jim") + "<br />")
document.write(arr)

</script>
```

Hasilnya:

Jani,Hege,Stale

4

Jani,Hege,Stale,Kai Jim

reverse()

Fungsi:

Membalikkan urutan elemen dalam suatu array Sintak:

```
arrayObject.reverse()
```

Contoh: <script type="text/j avascript">

```
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"
```

```
document.write(arr + "<br />")
document.write(arr.reverse())
```

```
</script>
```

Hasilnya:

Jani,Hege,Stale
Stale,Hege,Jani

shift()

Fungsi:

Untuk mendelete elemen pertama dari suatu array

Sintaks:

```
arrayObject.shift()
```

Contoh:

```
<script type="text/j avascript">
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"

document.write(arr + "<br />")
document.write(arr.shift() + "<br />")
document.write(arr)

</script>
```

Hasilnya:

Jani,Hege,Stale
Jani
Hege,Stale

unshift()

Fungsi:

Untuk menambahkan satu atau lebih elemen pada awal array Sintaks:

```
arrayObject.unshift(newelement1,newelement2,newelementX)
```

Contoh:

```
<script type="text/j avascript">
var arr = new Array()
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"

document.write(arr + "<br />") arr.unshift("Kai Jim")
document.write(arr)

</script>
```

Hasilnya:

Jani,Hege,Stale

Kai Jim,Jani,Hege,Stale

sort()

Fungsi:

Untuk mengurutkan elemen array Sintaks:

```
arrayObject.sort(sortby)
```

Contoh:

```
<script type="text/j avascript"> = new Array(6)
var arr "Jani"
arr[0] : "Hege"
arr [ "Stale"
1] "Kai Jim"
arr[2] ^ "Borge"
arr[3]
arr[4]
```

```
arr[5] = "Tove"

document.write(arr + "<br
/>")
document.write(arr.sort())

</script>
```

Hasilnya:

Jani,Hege,Stale,Kai Jim,Borge,Tove
Borge,Hege,Jani,Kai Jim,Stale,Tove

Tanggal dan Waktu

A. Metode Tanggal dan Waktu

Untuk membuat objek tanggal dan waktu terlebih dahulu anda mendeklarasikan variabel/objek tanggal dan waktu tersebut

```
var mydate= new Date()
```

Berikut adalah beberapa metode tanggal dan waktu di javascript

Metode	Deskripsi
getDate()	Menghasilkan tanggal dalam suatu bulan
getDay()	Meghasilkan nama hari
getHours()	Menghasilkan jam (dimulai dari 0-23)!
getMinutes()	Menghasilkan menit
getSeconds()	Menghasilkan detik
getMonth()	Menghasilkan bulan. (dimulai dari 0-11)!
getFullYear()	Menghasilkan tahun
getTime()	Menghasilkan waktu yang lengkap

Contoh:

```
<script type="text/j avascript"> var hari_ini= new Date() var tahun =  
hari_ini.getFullYear() var bulan = hari_ini.getMonth()+1 var tanggal =  
hari_ini.getDate()  
  
document.write("Tanggal hari ini: ")  
document.write(tanggal+"/"+bulan+"/"+tahun)  
  
</script>
```

Hasilnya:

Tanggal hari ini: 4/12/110

B. Javascript Timer

Pada javascript terdapat fungsi timer, yaitu `setTimeout()`, yang berfungsi untuk mengatur timer untuk mengeksekusi suatu fungsi atau kode tertentu

Sintaks:

```
setTimeout("expression", delaytime)
```


expression bisa berupa kode javascript atau suatu fungsi delaytime, adalah setiap berapa milidetik suatu expression dieksekusi

Contoh:

```
<html>
<body>
<input type="text" id="timer" size="12">
<input type="button" value="Start" onClick="count()">

<script type="text/j avascript"> var c=0
document.getElementById("timer").value = "" function
count () {
    document.getElementById("timer").value=
    c c=c+1
    setTimeout("count()",1000)
}
</script>
</body>
</html>
```

C. Membuat Jam Digital

```
<form name="Tick">
<input type="text" size="12" name="Clock"> </form>

<script type="text/j avascript"> function show()
{
var Digital=new Date() var hours=Digital.getHours() var
minutes=Digital.getMinutes() var
seconds=Digital.getSeconds() var dn="AM" if (hours>12){
dn="PM"
    hours=hours-12
}
if (hours==0)
hours=12
if (minutes<=9)
minutes="0"+minutes
if (seconds<=9)
seconds="0"+seconds
document.Tick.Clock.value=
hours+":"+minutes+":"+seconds+" "+dn
setTimeout("show()",1000)
}
show ()
</script>
```

Dynamic HTML

A. Mengakses dan Manipulasi Objek HTML

Untuk mengakses objek HTML, kita bisa 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

Fungsi: untuk mendapatkan isi teks atau html suatu objek atau tag HTML Contoh:

```
<html>
<head>
<script type="text/j avascript"> 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

Fungsi: untuk mengambil atau memanipulasi nilai suatu input form Contoh:

```
<html>
<head>
<script type="text/j avascript"> 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>
```

B. Memanipulasi Style atau CSS Objek HTML

Untuk memanipulasi style atau CSS suatu objek, adalah 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

```
<html>  
<head>  
</head>  
<body>  
<table border=1 width=500 bgcolor="yellow">  
<tr>  
<td onmouseover="this.bgColor='red'"  
onmouseout="this.bgColor='yellow'">Desrizal</td> </tr>  
<tr> <td onmouseover="this.bgColor='red'"  
onmouseout="this.bgColor='yellow'">Nurmi Yulita rahmi</td>  
</tr>  
</body>  
</html>
```

Mengubah warna latar halaman web

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

Mengubah posisi objek

```
<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>
```

Penanganan Form

A. Penanganan CheckBox di javascript

Deteksi apakah checkbox sudah dicek `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

```
<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>

```

B. Penanganan Input Radio di Javascript

```

<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>

```

C. Penanganan Select Box di Javascript

Mengambil nilai select box

```

<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>
```

Mendelete option dari dropdown list

```
<html>
<head>
<script type="text/j avascript"> 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>
```