



Variabel dan Fungsi JavaScript

Ikhwan Alfath Nurul Fathony

© Hak Cipta 2025 – Prodi Teknologi Informasi - Fakultas Teknik - Universitas Tidar



Variabel & Fungsi

Mahasiswa memahami logika
pemrograman dasar dengan JavaScript

JavaScript



JavaScript

- JavaScript adalah bahasa pemrograman yang digunakan untuk pengembangan website agar lebih dinamis.
- Website dinamis yang dimaksud adalah konten di dalamnya dapat bergerak atau mengubah apapun di layar tanpa harus dimuat ulang secara manual.
- Sebelumnya JavaScript hanya bekerja pada sisi client/frontend saja. Dengan begitu, proses pengolahan kode-kodenya hanya berjalan di sisi browser.
- Seiring perkembangannya, JavaScript juga bisa digunakan di sisi server dengan bantuan berbagai macam framework seperti Node.js, dan React.js.
- Selain itu JavaScript juga dapat digunakan untuk mengedit DOM (Document Object Model).

JavaScript and Java are completely different languages, both in concept and design.

JS

Penulisan Tag JavaScript

JavaScript



```
<script>  
document.getElementById("demo").innerHTML = 5 + 6;  
</script>
```

JavaScript



Type Data Variable

JavaScript memiliki berbagai tipe data.

- Number adalah tipe data angka, contoh: `let age = 25`
- String adalah tipe data teks, contoh: `let name = "John"`
- Boolean adalah tipe data nilai benar atau salah, contoh: `let isAdult = true`
- Array adalah tipe data berupa daftar nilai, contoh: `let fruits = ["Apple", "Banana", "Cherry"];`
- Object adalah tipe data berupa Koleksi nilai, contoh: `let person = {firstName: "John", lastName: "Doe"}`

OPERATOR

ARITMATIKA

Operator (=) memberikan nilai ke variabel.

❖ `let x = 10;`

Operator penjumlahan (+) menambahkan angka

❖ `let x = 5; let y = 2;`
`let z = x + y;`

Operator perkalian (*) mengalikan angka.

❖ `let x = 5; let y = 2;`
❖ `let z = x * y;`

Operator Aritmatika adalah operator yang biasanya temukan dalam **Operasi Matematika**

Operator	Deskripsi
+	Tambahan
-	Pengurangan
*	Perkalian
**	Eksponen (ES2016)
/	Pembagian
%	Modulus
++	Increment
--	Decrement

OPERATOR ASSIGNMENT

```
let x = 10;  
x *= 5;
```

Operator Assignment adalah suatu operator penugasan yang digunakan untuk memasukkan nilai ke **dalam suatu variable**.

Operator	Contoh
=	x = y
+=	x += y
-=	x -= y
*=	x *= y
/=	x /= y
%=	x %= y
**=	x **= y

Perulangan

Looping atau perulangan adalah suatu metode dalam pemrograman untuk mengeksekusi suatu perintah yang sama terus-menerus hingga kondisi tertentu terpenuhi.

```
text += cars[0] + "<br>";  
text += cars[1] + "<br>";  
text += cars[2] + "<br>";  
text += cars[3] + "<br>";  
text += cars[4] + "<br>";  
text += cars[5] + "<br>";
```



```
const cars = ["BMW", "Volvo", "Saab", "Ford", "Fiat", "Audi"];  
let text = "";  
for (let i = 0; i < cars.length; i++) {  
  text += cars[i] + "<br>";  
}
```


Perulangan FOR

for - loop blok kode beberapa kali

```
for (expression 1; expression 2; expression 3) {  
    // code block to be executed  
}
```

```
for (let i = 0; i < 5; i++) {  
    console.log(i);  
}
```

Perulangan FOR

for - loop blok kode beberapa kali

```
for (key in object) {  
    // code block to be executed  
}
```

```
const person = {fname:"John", lname:"Doe", age:25};  
  
let text = "";  
for (let x in person) {  
    text += person[x];  
}
```

Perulangan WHILE

while - loop melalui blok kode saat kondisi yang ditentukan benar

```
while (condition) {  
    // code block to be executed  
}
```

```
while (i < 10) {  
    text += "The number is " + i;  
    i++;  
}
```

Perulangan DO-WHILE

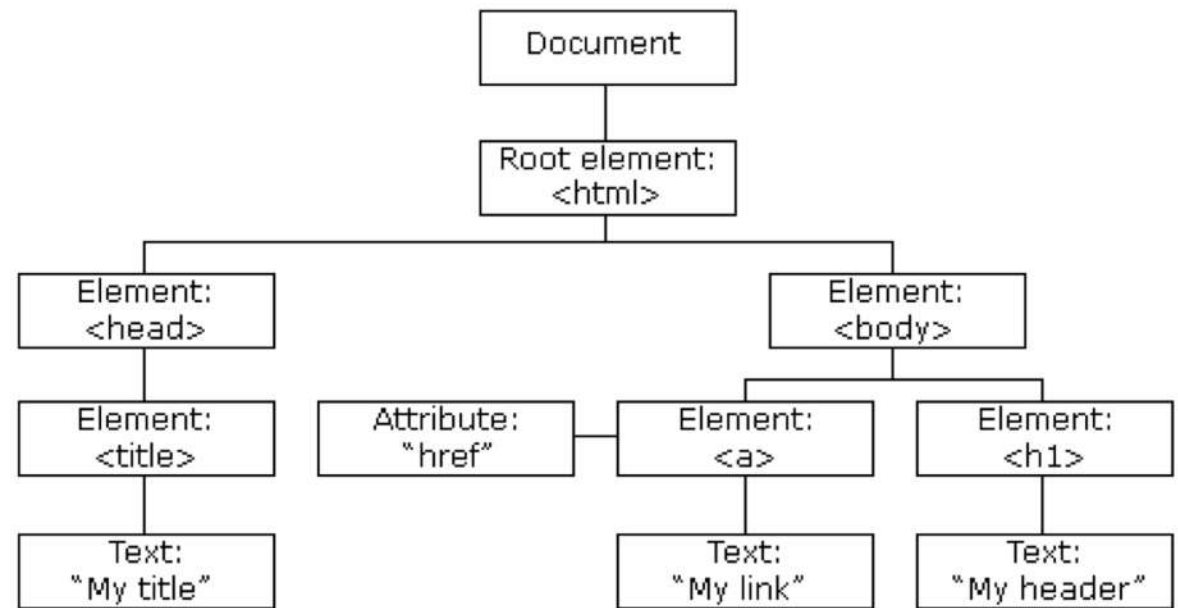
do/while - loop dalam pemrograman yang memastikan blok kode di dalamnya dieksekusi setidaknya sekali, sebelum memeriksa kondisi untuk menentukan apakah loop harus dilanjutkan atau dihentikan.

```
do {  
    // code block to be executed  
}  
while (condition);
```

```
do {  
    text += "The number is " + i;  
    i++;  
}  
while (i < 10);
```

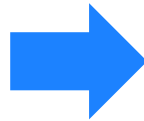
Pengenalan DOM

- DOM atau Document Object Model, adalah struktur data yang digunakan oleh browser untuk mewakili dokumen HTML atau XML yang sedang diakses.
- DOM adalah representasi dari halaman web dalam bentuk struktur pohon yang terdiri dari node.
- Setiap bagian dari halaman web, seperti elemen HTML, atribut, dan teks, diwakili sebagai node di dalam pohon ini.
- DOM memungkinkan JavaScript untuk berinteraksi dengan dan memanipulasi halaman web secara dinamis.
- Pengembang dapat mengakses dan mengubah konten, struktur, dan gaya halaman web secara dinamis tanpa perlu memuat ulang halaman.



Struktur DOM

```
<!DOCTYPE html>
<html>
  <head>
    <title>Halaman
Contoh</title>
  </head>
  <body>
    <h1>Selamat Datang</h1>
    <p>Ini adalah paragraf
contoh.</p>
  </body>
</html>
```



```
Document
├── html
│   ├── head
│   │   └── title
│   │       └── "Halaman Contoh"
│   └── body
│       ├── h1
│       │   └── "Selamat Datang"
│       └── p
│           └── "Ini adalah paragraf contoh."
```


Interaksi dengan DOM menggunakan JavaScript

Dengan DOM objek, JavaScript mendapatkan semua yang dibutuhkan untuk membuat HTML dinamis:

- JavaScript dapat mengubah semua elemen HTML
- JavaScript dapat mengubah semua atribut HTML
- JavaScript dapat mengubah semua styles CSS
- JavaScript dapat menghapus elemen dan atribut HTML yang ada
- JavaScript dapat menambahkan elemen dan atribut HTML baru
- JavaScript dapat bereaksi terhadap semua peristiwa HTML yang ada di halaman
- JavaScript dapat membuat events baru di HTML

Contoh penggunaan DOM

Mengubah konten

```
let header = document.getElementById("header");  
header.innerHTML = "Judul Baru";
```

Mengubah gaya

```
let header = document.getElementById("header");  
header.style.color = "blue";
```

Menambahkan atau menghapus elemen

```
let newParagraph =  
document.createElement("p"); newParagraph.textContent = "Paragraf  
baru"; document.body.appendChild(newParagraph);
```

Contoh penggunaan DOM

Manipulasi DOM dengan event

```
<button id="myButton">Klik Saya</button>
<p id="myParagraph">Ini adalah paragraf.</p>

<script>
  let button = document.getElementById("myButton");
  let paragraph = document.getElementById("myParagraph");

  button.addEventListener("click", function() {
    paragraph.textContent = "Tombol telah diklik!";
  });
</script>
```

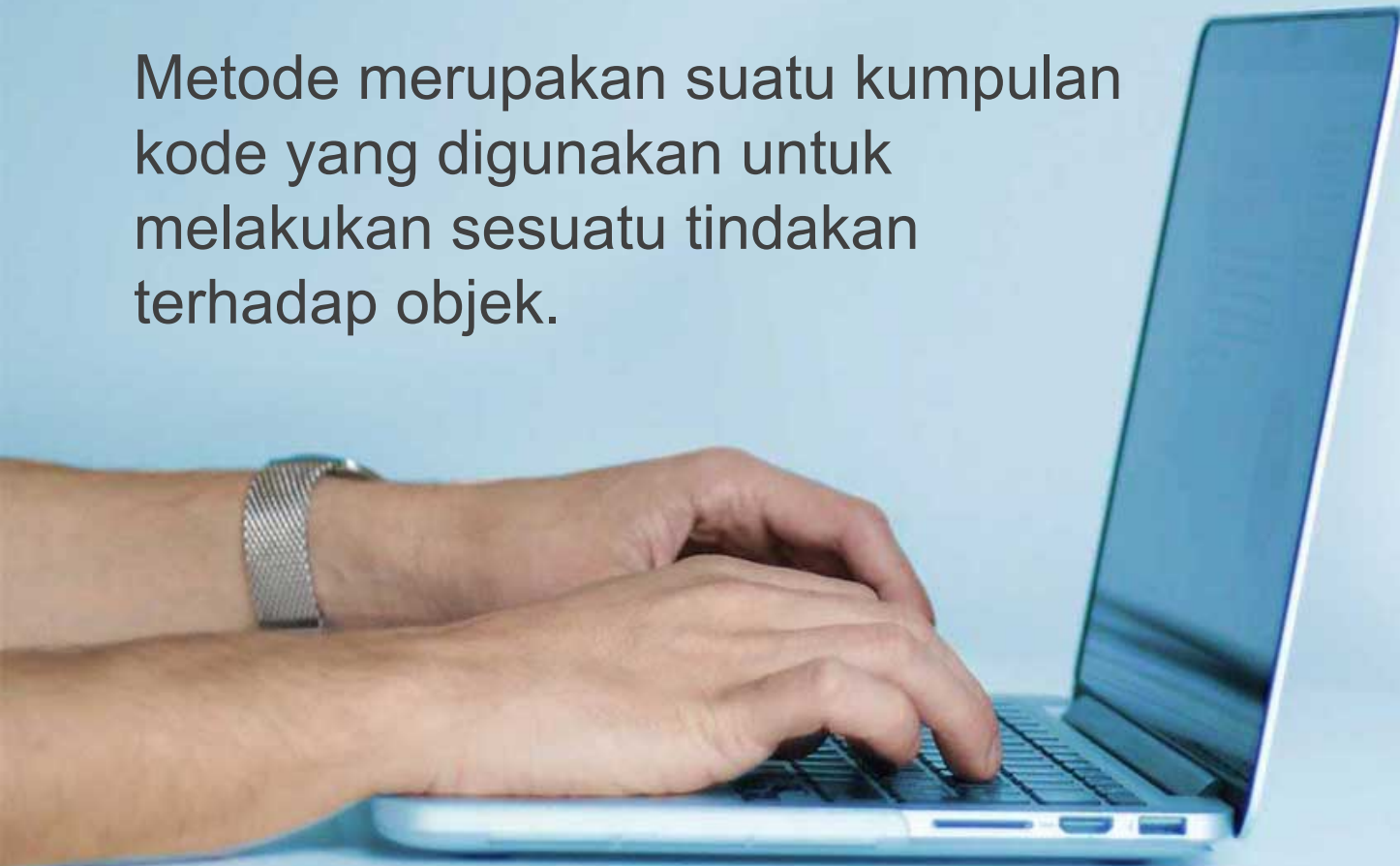
Modern Portfolio Designed

METODE JS

Metode merupakan suatu kumpulan kode yang digunakan untuk melakukan sesuatu tindakan terhadap objek.

PENULISANNYA
dipisahkan
dengan tanda titik

nama_objek.nama_metode
("parameter")



Mengakses elemen

Method	Keterangan
<code>document.getElementById(id)</code>	Menemukan elemen demi elemen id
<code>document.getElementsByTagName(name)</code>	Menemukan elemen berdasarkan nama tag
<code>document.getElementsByClassName(name)</code>	Menemukan elemen berdasarkan nama kelas

Kotak Dialog – alert()

Contoh:

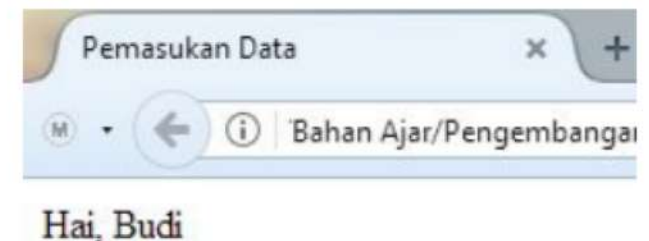
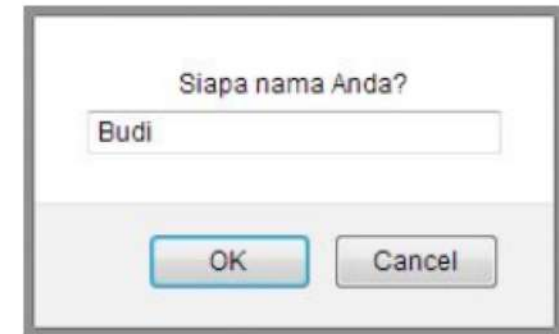
```
<HTML>
<HEAD>
<TITLE>Alert Box</TITLE>
</HEAD>
<BODY>
<SCRIPT LANGUAGE = "JavaScript">
<!--
window.alert("Ini merupakan pesan
untuk Anda");
//-->
</SCRIPT>
</BODY>
</HTML>
```



Kotak Dialog – confirm()

Contoh:

```
<HTML>
<HEAD>
<TITLE>Konfirmasi</TITLE>
</HEAD>
<BODY>
<SCRIPT LANGUAGE = "JavaScript">
<!--
var jawaban = window.confirm("Anda ingin
meneruskan?");
document.write("Jawaban Anda: " +
jawaban);
//-->
</SCRIPT>
</BODY>
</HTML>
```



Pernyataan IF

- **Pernyataan IF tanpa else**

```
if (kondisi) {  
    // blok pernyataan yang dijalankan  
    // kalau kondisi bernilai benar  
}
```

```
<HTML>  
<HEAD>  
<TITLE>Contoh if</TITLE>  
</HEAD>  
<BODY>  
<SCRIPT LANGUAGE = "JavaScript">  
<!--  
    var nilai = prompt("Nilai (0-100): ", 0);  
    var hasil = "Tidak Lulus";  
  
    if (nilai >= 60)  
        hasil = "Lulus";  
  
    document.write("Hasil: " + hasil);  
    //-->  
</SCRIPT>  
</BODY>  
</HTML>
```

- **Pernyataan IF dengan ELSE**

```
if (kondisi) {  
    // blok pernyataan yang dijalankan  
    // kalau kondisi bernilai benar  
} else {  
    // blok pernyataan yang dijalankan  
    // kalau kondisi bernilai salah  
}
```

```
<HTML>  
<HEAD>  
<TITLE>Contoh if-else</TITLE>  
</HEAD>  
<BODY>  
<SCRIPT LANGUAGE = "JavaScript">  
<!--  
    var nilai = prompt("Nilai (0-100): ", 0);  
    var hasil = "";  
  
    if (nilai >= 60)  
        hasil = "Lulus";  
    else  
        hasil = "Tidak Lulus";  
  
    document.write("Hasil: " + hasil);  
    //-->  
</SCRIPT>  
</BODY>  
</HTML>
```

Fungsi

- Mendefinisikan Fungsi

```
function nama(daftar_parameter) {  
    pernyataan_1;  
    pernyataan_n;  
}
```

c = jumlah (2 , 3);

Nama fungsi

Nilai balik

argumen

```
HTML>  
<HEAD>  
<TITLE>Contoh Fungsi</TITLE>  
</HEAD>  
<BODY>  
<SCRIPT LANGUAGE = "JavaScript">  
<!--  
    function jumlah(x, y) {  
        var hasil = x + y;  
        return(hasil);  
    }  
    var z = jumlah(2, 3);  
    document.write(z);  
    document.write("<BR>");  
    document.write(jumlah(4, 5));  
    //-->  
</SCRIPT>  
</BODY>  
</HTML>
```

5
9

Fungsi Rekursif

- Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri.
- Contoh : Faktorial

```
<HTML>
<HEAD>
<TITLE>Variabel Lokal</TITLE>
</HEAD>
<BODY>
<SCRIPT LANGUAGE = "JavaScript">
<!--
  function faktorial(n) {
    if (n == 0)
      return(1);
    else
      return( n * faktorial(n-1));
  }

  var z = 77;
  document.write("4! = " + faktorial(4));
//-->
</SCRIPT>
</BODY>
</HTML>
```

$$4! = 24$$

JQuery

JQuery

- JQuery adalah library JavaScript yang lightweight, "menulis lebih sedikit, melakukan lebih banyak".
- Tujuan dari jQuery adalah untuk mempermudah penggunaan JavaScript di situs web Anda.
- JQuery mengambil banyak tugas umum yang membutuhkan banyak baris kode JavaScript untuk diselesaikan, dan membungkusnya menjadi metode yang dapat Anda panggil dengan satu baris kode.
- JQuery juga menyederhanakan banyak hal rumit dari JavaScript, seperti panggilan AJAX dan manipulasi DOM.

jQuery

- Library jQuery berisi fitur-fitur berikut:
 - Manipulasi HTML/DOM
 - Manipulasi CSS
 - Metode event HTML
 - Efek dan animasi
 - AJAX
- Ada banyak library JavaScript lain, tetapi jQuery termasuk yang paling populer, dan juga yang paling dapat diperluas.
- Banyak perusahaan terbesar di Web menggunakan jQuery, seperti:
 - Google
 - Microsoft
 - IBM
 - Netflix

Cara Penggunaan JQuery

Ada beberapa cara untuk menggunakan jQuery.

1. Unduh library JQuery dari jquery.com

```
<head>  
<script src="jquery-3.6.0.min.js"></script>  
</head>
```

2. Sertakan JQuery dari CDN, seperti Google

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/  
jquery.min.js"></script>  
</head>
```

Jquery Syntax

- Sintaks JQuery dibuat khusus untuk memilih elemen HTML dan melakukan beberapa tindakan pada elemen.
- Sintaks dasarnya adalah: **\$(selector).action()**
 - Tanda \$ untuk mendefinisikan/mengakses jquery
 - (selector) untuk "meminta (atau menemukan)" elemen HTML
 - action () yang akan dilakukan pada elemen
- Contoh:
- `$(this).hide()` - menyembunyikan elemen
- `$("#test").hide()` - menyembunyikan elemen dengan id="test"

JavaScript vs JQuery

Parameter	JavaScript	JQuery
Fungsi	Sebagai sebuah Bahasa pemrograman	Sebuah application programming interface (API) serta library JavaScript lintas platform yang dirancang untuk meningkatkan fitur browser web
Coding/Scripting	Mengharuskan programmer untuk menulis skrip, sehingga memakan banyak waktu	Sebagian besar skrip sudah ada di library
Panjang Kode	Membutuhkan lebih banyak baris kode	Membutuhkan lebih sedikit baris kode
Kecepatan/ Performa	JavaScript murni mengakses Document Object Models (DOM) lebih cepat karena browser langsung mengakses bahasa.	Meskipun cepat, masih perlu sedikit waktu untuk dikonversi menjadi JavaScript agar dijalankan di browser.
Event Handling/ Interaksi	Mungkin melakukan event handling/interaksi, tetapi membutuhkan banyak baris kode	Mudah ditangani, karena banyak fungsi (animasi, peristiwa) sudah ditentukan sebelumnya di library. Tidak perlu kode tambahan.

JavaScript vs JQuery

Menyembunyikan elemen yang memiliki id “textbox”

```
// JavaScript
document.getElementById('textbox').style.display = "none";
// jQuery
$('#textbox').hide();
```

Membuat tag <h1> dengan teks “my text”

```
// JavaScript
var h1 = document.createElement("h1");
h1.innerHTML = "my text";
document.getElementsByTagName('body')[0].appendChild(h1);

// jQuery
$('body').append($("<h1/>").html("my text"));
```

Latihan



Buat skrip sehingga
menghasilkan
seperti ini

Password Input

Your password ●●

Verify password ●

Reset

Submit Query

The two passwords are not the same
Please re-enter both now

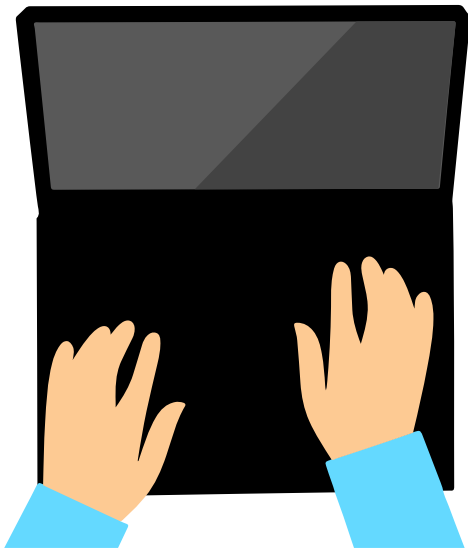
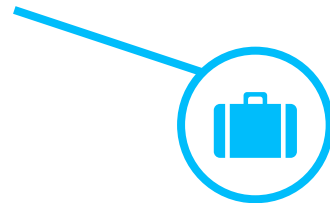
OK

TUGAS



Buatlah sebuah form yang dilayout dengan tabel dan CSS, dan ketika form tersebut disubmit, akan

menampilkan hasil input pada halaman yang sama.



THANK YOU

Universitas Tidar