MENERAPKAN PERINTAH JAVASCRIPT

POKOK BAHASAN

- Pengenalan javascript
- Cara menulis javascript
- Variabel dan tipe data di javascript
- Operator dalam Javascript
- Operator percabangan
- Operator perulangan
- Struktur data javascript

PENDAHULUAN

Definisi JavaScript

JavaScript adalah bahasa pemrograman yang berbentuk kumpulan skrip yang biasanya digunakan untuk menambahkan interaksi antara halaman web dengan pengunjung halaman web.

JavaScript dijalankan pada sisi klien yang akan memberikan kemampuan fitur-fitur tambahan halaman web yang lebih baik dibandingkan fitur-fitur yang terdapat pada HTML.

Saat ini javascript tidak hanya digunakan di sisi client (browser) saja. Javascript juga digunakan pada server, console, program desktop, mobile, IoT, game, dan lain-lain

Definisi JavaScript

Javascript, seperti namanya, merupakan bahasa pemrograman scripting. Dan seperti Bahasa scripting lainnya, Javascript umumnya digunakan hanya untuk program yang tidak terlalu besar, biasanya hanya beberapa ratus baris. Javascript pada umumnya mengontrol program yang berbasis Java. Jadi memang pada dasarnya Javascript tidak dirancang untuk digunakan dalam aplikasi skala besar

JavaScript

Javascript adalah bahasa pemrograman yang awalnya dirancang untuk berjalan di atas browser. Namun, seiring perkembangan zaman, javascript tidak hanya berjalan di atas browser saja. Javascript juga dapat digunakan pada sisi Server, Game, IoT, Desktop, dsb.

JavaScript saat ini sudah banyak dikembangkan menjadi beberapa pengetahuan lain, seperti AJAX. Asynchronous JavaScript dan XML (AJAX) adalah teknik yang digunakan pada halaman website dengan menggunakan bahasa pemrograman JavaScript Metode yang diterapkan adalah menerima dan mengirimkan data dengan server tanpa harus menyegarkan (refresh) halaman itu.

Javascript awalnya bernama Mocha, lalu berubah menjadi LiveScript saat browser Netscape Navigator 2.0 rilis versi beta (September 1995). Namun, setelah itu dinamai ulang menjadi Javascript.

Terinspirasi dari kesuksesan Javascript, Microsoft mengadopsi teknologi serupa. Microsoft membuat 'Javascript' versi mereka sendiri bernama JScript. Lalu di tanam pada Internet Explorer 3.0.

Prinsip Dasar JavaScript

- Javascript mendukung paradigma pemrograman imparatif (Javascript dapat menjalankan perintah program baris demi baris, dengan masing-masing baris berisi satu atau lebih perintah), fungsional (struktur dan elemen-elemen dalam program sebagai fungsi matematis yang tidak memiliki keadaan (state) dan data yang dapat berubah (mutable data), dan orientasi objek (segala sesuatu yang terlibat dalam program dapat disebut sebagai "objek")
- Javascript memiliki model pemrograman fungsional yang sangat ekspresif
- Pemrograman berorientasi objek (PBO) pada Javascript memiliki perbedaan dari PBO pada umumnya
- Program kompleks pada javascript umumnya dipandang sebagai program-program kecil yang saling berinteraksi

CARA MENULIS JAVASCRIPT

BAGAIMANA CARA MENULIS KODE JAVASCRIPT DI HTML?

Embed

Inline

Eksternal

Embed

01.PenulisanEmbed.html

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Penulisan Embed - JS</title>
    <script>
      console.log('Hello Js dari element Head');
    </script>
  </head>
  <body>
    <p>Belajar Java Script</p>
    <script>
      console.log('Hello Js dari element Body');
    </script>
  </body>
</html>
```

Belajar Java Script

Inline

02.PenulisanInline.html

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Penulisan Inline - JS</title>
    <script>
      console.log('Hello Js dari element Head');
    </script>
  </head>
  <body>
    <p>Belajar Java Script</p>
    <a href="#" onclick="alert('Yeeeeeeee.')">Klik Aku</a>
    <script>
      console.log('Hello Js dari element Body');
    </script>
  </body>
</html>
```

Penulisan Inline - JS

127.0.0.1:5500/Praktikum/02.PenulisanInline.html

Belajar Java Script

Klik Aku

Elements Console Sources Network Performance

top Filter Default levels No Issues

```
Hello Js dari element Head        01.PenulisanEmbed.html:6
Hello Js dari element Body        01.PenulisanEmbed.html:12
>
```

External

03.PenulisanExternal.html | external.js

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Penulisan External - JS</title>
  </head>
  <body>
    <p>Belajar Java Script</p>
    <script src="Js/external.js"></script>
  </body>
</html>
```

03.PenulisanExternal.html | external.js

```js
alert('Hallo, ini kode program Js External');
```

MENAMPILKAN OUTPUT PADA JAVASCRIPT

Menggunakan Fungsi console.log();

Menggunakan Fungsi alert();

Menggunakan Fungsi document.write();

Menggunakan innerHTML.

console.log();

Fungsi console.log() adalah fungsi untuk menampilkan teks ke console Javascript.

01.PenulisanEmbed.html

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Penulisan Embed - JS</title>
    <script>
      console.log('Hello Js dari element Head');
    </script>
  </head>
  <body>
    <p>Belajar Java Script</p>
    <script>
      console.log('Hello Js dari element Body');
    </script>
  </body>
</html>
```

Elements | Console | Sources | Network | Performance

top | Filter | Default levels | No Issues

```
Hello Js dari element Head        01.PenulisanEmbed.html:6
Hello Js dari element Body        01.PenulisanEmbed.html:12
>
```

alert();

Fungsi alert() adalah fungsi untuk menampilkan jendela dialog. Fungsi sebenarnya berada pada objek window. Secara lengkap bisa ditulis seperti ini: window.alert("Hello World");

Bisa juga ditulis alert() saja seperti ini: alert("Hello World");

05.Alert.html

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Alert - JS</title>
    <script>
      alert('selamat datang, Sekarang kita belajar Javascript');
      function sayHello() {
        alert('Hello');
      }
    </script>
  </head>
  <body>
    <button onclick="sayHello()">Klik saya.!</button>
  </body>
</html>
```

document.write()

Objek document adalah objek yang mewakili dokumen HTML di dalam Javascript.
Dalam objek document, terdapat fungsi write() untuk menulis sesuatu ke dokumen HTML.

06.DocumentWrite.html

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>DocumentWrite</title>
    <script>
      document.write('<h1> Hello World </h1>');
      document.write('<hr>');
      document.write('<p> saya sedang belajar Javascript');
    </script>
  </head>
  <body></body>
</html>
```

InnerHTML

innerHTML adalah sebuah atribut di dalam (objek) elemen HTML yang berisi string HTML.

Dengan innerHTML, kita dapat menampilkan output ke elemen yang lebih spesifik.

VARIABEL DAN TIPE DATA DALAM JAVASCRIPT

Variabel

Variabel adalah sebuah nama yang mewakili sebuah nilai. Variabel bisa diisi dengan **berbagai macam nilai** seperti string (teks), number (angka), objek, array, dan sebagainya.

Kita bisa ibaratkan, variabel itu seperti **wadah** untuk menyimpan sesuatu.

Membuat Variabel

Cara membuat variabel yang umum digunakan di javascript adalah menggunakan kata kunci var lalu diikuti dengan nama variabel dan nilainya. Contoh :

- var title = "Belajar Pemrograman Javascript";
- var siteName = "Blog Saya";
- var url = "https://www.blogku.com";
- var visitorCount = 90;

Perlu kamu ketahui juga, selain kata kunci var kita juga bisa membuat variabel dengan kata kunci let atau tanpa awalan apapun

Membuat Variabel

Cara membuat variabel yang umum digunakan di javascript adalah menggunakan kata kunci var lalu diikuti dengan nama variabel dan nilainya. Contoh :

- var title = "Belajar Pemrograman Javascript";
- var siteName = "Blog Saya";
- var url = "https://www.blogku.com";
- var visitorCount = 90;

Perlu kamu ketahui juga, selain kata kunci var kita juga bisa membuat variabel dengan kata kunci let atau tanpa awalan apapun

Membuat variabel

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Belajar Variabel dalam Javascript</title>
    <script>
        // membuat variabel
        var name = "Detik.com";
        var visitorCount = 34567;
        var isActive = true;
        var url = "https://www.detik.com";

        // menampilkan variabel ke jendela dialog (alert)
        alert("Selamat datang di " + name);

        // menampilkan variabel ke dalam HTML
        document.write("Nama Situs: " + name + "<br>");
        document.write("Jumlah Pengunjung: " + visitorCount + "<br>");
        document.write("Status Aktif: " + isActive + "<br>");
        document.write("Alamat URL: " + url + "<br>");
    </script>
</head>
<body>


</body>
</html>
```

Mengisi Ulang Variabel

Variabel bersifat mutable, artinya nilai yang tersimpan di dalamnya dapat kita isi ulang (berubah).

```
// mula-mula kita buat variabel dengan isi seperti ini
var age = 18;

// lalu kita isi ulang
age = 21;
```

Karena kata kunci var dibutuhkan saat membuat variabel saja. Sedangkan untuk mengisi ulang, kita cukup tulis seperti di atas.

Apabila kita menggunakan kata kunci var, berarti jadinya kita membuat variabel baru donk, bukan mengisi ulang

Menghapus Variabel

Penghapusan variabel dalam Javascript memang jarang dilakukan. Namun, untuk program yang membutuhkan ketelitian dalam alokasi memori, penghapusan variabel perlu dilakukan agar penggunaan memori lebih optimal.

Penghapusan variabel dapat dilakukan dengan katakunci delete.

```
bookTitle = "Belajar Pemrograman Javascript";
delete bookTitle;
```

Mengenal Tipe data pada Javascript

Tipe data adalah jenis-jenis data yang bisa kita simpan di dalam variabel. Ada beberapa tipe data dalam pemrograman Javascript:

- String (teks)
- Float
- Object
- Integer atau Number
- Boolean

Mengenal Tipe Data JS

Javascript adalah bahasa yang bersifat dynamic typing, artinya kita tidak harus menuliskan tipe data pada saat pembuatan variabel seperti pada bahasa C, C++, Java, dsb. yang bersifat static typing. Javascript akan otomatis mengenali tipe data yang kita berikan pada variabel.

Aturan Penulisan Nama Variabel di Javascript

- Penamaan variabel tidak boleh menggunakan angka di depannya contoh var 123name = “hasan”;
- Penamaan variabel boleh menggunakan awal underscore. Contoh var _name = “hasan” ;
- Penamaan variabel dianjurkan menggunakan camelCase apabila tediri dari dua suku kata. Contoh var fullName = “hasan”;
- Penamaan variabel dianjurkan menggunakan bahasa inggris. Contoh var postTitle = "Belajar Javascript";

OPERATOR DALAM JAVASCRIPT

Mengenal Operator pada Javascript

Operator adalah simbol yang digunakan untuk melakukan operasi pada suatu nilai dan variabel. Operator dalam pemrograman terbagi dalam 6 jenis:

- Operator aritmatika
- Opeartor relasi atau perbandingan
- Operator Penugasan
- Operator Logika
- Operator Bitwise
- Operator Ternary

OPERATOR ARITMATIKA

Nama Operator	Simbol
Penjumlahan	+
Pengurangan	-
Perkalian	*
Pemangkatan	**
Pembagian	/
Sisa Bagi	%

OPERATOR ARITMATIKA

aritmatika.html

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Operator Aritmatika</title>
</head>
<body>
    <script>
        var a = 20;
        var b = 4;
        var c = 0;
        // pengurangan
        c = a - b;
        document.write(`${a} - ${b} = ${c}<br/>`);
        // Perkalian
        c = a * b;
        document.write(`${a} * ${b} = ${c}<br/>`);
        // pemangkatan
        c = a ** b;
        document.write(`${a} ** ${b} = ${c}<br/>`);
        // Pembagian
        c = a / b;
        document.write(`${a} / ${b} = ${c}<br/>`);
        // Modulo
        c = a % b;
        document.write(`${a} % ${b} = ${c}<br/>`);
    </script>
</body>

</html>
```

Operator Penugasan

Operator penugasan adalah operator yang digunakan untuk memberikan tugas kepada variabel. Biasanya digunakan untuk mengisi variabel.

Nama Operator	Sombol
Pengisian Nilai	=
Pengisian dan Penambahan	+=
Pengisian dan Pengurangan	-=
Pengisian dan Perkalian	*=
Pengisian dan Pemangkatan	**=
Pengisian dan Pembagian	/=
Pengisian dan Sisa bagi	%=

Operator Penugasan

penugasan.html

```html
<!DOCTYPE html>
<html lang="en">
<head><title>Operator Penugasan</title></head>
<body>
    <script>
        document.write("Mula-mula nilai score...<br>");
        // pengisian nilai
        var score = 100;
        document.write("score = "+ score + "<br/>");
        // pengisian dan menjumlahan dengan 5
        score += 5;
        document.write("score = "+ score + "<br/>");
        // pengisian dan pengurangan dengan 2
        score -= 2;
        document.write("score = "+ score + "<br/>");
        // pengisian dan perkalian dengan 2
        score *= 2;
        document.write("score = "+ score + "<br/>");
        // pengisian dan pembagian dengan 4
        score /= 4;
        document.write("score = "+ score + "<br/>");
        // pengisian dan pemangkatan dengan 2
        score **= 2;
        document.write("score = "+ score + "<br/>");
        // pengisian dan modulo dengan 3;
        score %= 3;
        document.write("score = "+ score + "<br/>");
    </script>
</body>
</html>
```

Operator Perbandingan

Operator relasi atau perbandingan adalah operator yang digunakan untuk membandingkan dua nilai. Operator perbandingan akan menghasilkan sebuah nilai boolean true dan false.

Nama Operator	Simbol
Lebih Besar	>
Lebih Kecil	<
Sama Dengan	== atau ===
Tidak Sama dengan	!= atau !==
Lebih Besar Sama dengan	>=
Lebih Kecil Sama dengan	<=
Nama Operator	Simbol

Operator Perbandingan

```
<body>
    <script>
        var aku = 25;
        var kamu = 15;

        // sama dengan
        var hasil = aku == kamu;
        document.write(`${aku} == ${kamu} = ${hasil}<br/>`);

        // lebih besar
        var hasil = aku > kamu;
        document.write(`${aku} > ${kamu} = ${hasil}<br/>`);

        // lebih besar samadengan
        var hasil = aku >= kamu;
        document.write(`${aku} >= ${kamu} = ${hasil}<br/>`);

        // lebih kecil
        var hasil = aku < kamu;
        document.write(`${aku} < ${kamu} = ${hasil}<br/>`);

        // lebih kecil samadengan
        var hasil = aku <= kamu;
        document.write(`${aku} <= ${kamu} = ${hasil}<br/>`);

        // tidak samadengan
        var hasil = aku != kamu;
        document.write(`${aku} != ${kamu} = ${hasil}<br/>`);
    </script>
</body>
```

Operator Logika

Operator logika digunakan untuk melakukan operasi terhadap dua nilai Boolean, true dan false

Nama Operator	Simbol
Logika AND	&&
Logika OR	||
Negasi/kebalikan	!

Operator Logika

logika.html

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Operator Logika</title>
</head>
<body>
    <script>
        var aku = 20;
        var kamu = 10;

        var benar = aku > kamu;
        var salah = aku < kamu;

        // operator && (and)
        var hasil = benar && salah;
        document.write(`${benar} && ${salah} = ${hasil}<br/>`);

        // operator || (or)
        var hasil = benar || salah;
        document.write(`${benar} || ${salah} = ${hasil}<br/>`);

        // operator ! (not)
        var hasil = !benar
        document.write(`!${benar} = ${hasil}<br/>`);

    </script>
</body>

</html>
```

Operator Bitwise

Operator bitwise merupkan operator yang digunakan untuk operasi berdasarkan bit (biner).

Nama	Simbol di Java
AND	&
OR	|
XOR	^
Negasi/kebalikan	~
Left Shift	<<
Right Shift	>>
Left Shift (unsigned)	<<<
Right Shift (unsigned)	>>>

Operator Bitwise

```html
<head>
    <title>Operator Bitwise</title>
</head>
<body>
    <script>
        var x = 4;
        var y = 3;
        // operator bitwise and
        var hasil = x & y;
        document.write(`${x} & ${y} = ${hasil}<br/>`);
        // operator bitwise or
        var hasil = x | y;
        document.write(`${x} | ${y} = ${hasil}<br/>`);
        // operator bitwise xor
        var hasil = x ^ y;
        document.write(`${x} ^ ${y} = ${hasil}<br/>`);
        // operator negasi
        var document: Document
        document.write(`~${x} = ${hasil}<br/>`);
        // operator bitwise right shift >>
        var hasil = x >> y;
        document.write(`${x} >> ${y} = ${hasil}<br/>`);
        // operator bitwise right shift <<
        var hasil = x << y;
        document.write(`${x} << ${y} = ${hasil}<br/>`);
        // operator bitwise right shift (unsigned) >>>
        var hasil = x >>> y;
        document.write(`${x} >>> ${y} = ${hasil}<br/>`);
    </script>
</body>
```

Operator Ternary

Jika Operator-operator sebelumnya hanya dua bagian saja, yaitu: bagian kiri dan kanan. Atau biasa disebut operator binary.

Sedangkan Operator ternary merupakan operator yang teridiri dari tiga bagian. Operator trinary terdiri dari bagian kiri, tengah, dan kanan

```
bagian kiri <operator> bagian tengah <operator> bagian kanan
```

```
<kodisi> ? "benar" : "salah"
```

Operator Ternary

kamu suka aku ? ya : tidak;

jawaban benar

jawaban salah

Operator Ternary

ternary.html

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <title>Operator Ternary</title>
</head>

<body>

    <script>
        var pertanyaan = confirm("Apakah kamu berumur diatas 18 tahun?")

        var hasil = pertanyaan ? "Selamat datang" : "Kamu tidak boleh di sini";
        document.write(hasil);
    </script>
</body>

</html>
```

PERCABANGAN DALAM JAVASCRIPT

Percabangan If

```
if (condition) {
    block of code to be executed if the condition is true
}
```

Percabangan If

banganIF.html ×

percabangan ▸ percabanganIF.html ▸ html

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Percabangan if</title>
</head>
<body>
    <script>
        var totalBelanja = prompt("Total belanja?", 0);

        if(totalBelanja > 100000){
            document.write("<h2>Selamat Anda dapat hadiah</h2>");
        }

        document.write("<p>Terimakasih sudah berbelanja di toko kami</p>");
    </script>
</body>
</html>
```

Terimakasih sudah berbelanja di toko kami

Selamat Anda dapat hadiah

Terimakasih sudah berbelanja di toko kami

Percabangan If/Else

```
if (condition) {
    block of code to be executed if the condition is true
} else {
    block of code to be executed if the condition is false
}
```

Percabangan If/Else

percabanganIFELSE.html

latihan4percabangan ▸ percabanganIFELSE.html ▸ html ▸ head

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Percabangan if/else</title>
</head>
<body>
    <script>
        var password = prompt("Password:");

        if(password == "kopi"){
            document.write("<h2>Selamat datang bos!</h2>");
        } else {
            document.write("<p>Password salah, coba lagi!</p>");
        }

        document.write("<p>Terima kasih sudah menggunakan aplikasi ini!</p>");

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

Percabangan If/Else

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Percabangan if/else/if</title>
</head>
<body>
    <script>
        var nilai = prompt("Inputkan nilai akhir:");
        var grade = "";

        if(nilai >= 90) grade = "A"
        else if(nilai >= 80) grade = "B+"
        else if(nilai >= 70) grade = "B"
        else if(nilai >= 60) grade = "C+"
        else if(nilai >= 50) grade = "C"
        else if(nilai >= 40) grade = "D"
        else if(nilai >= 30) grade = "E"
        else grade = "F";

        document.write(`<p>Grade anda: ${grade}</p>`);
    </script>
</body>
</html>
```

Percabangan switch

```
switch(expression) {
    case x:
        code block
        break;
    case y:
        code block
        break;
    default:
        code block
}
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Percabangan switch/case</title>
</head>
<body>
    <script>

        var jawab = prompt("Kamu beruntung! Silahakn pilih hadiahmu dengan memasukan angka 1 sampai 5"
        var hadiah = "";

        switch(jawab){
            case "1":
                hadiah = "Tisu";
                break;
            case "2":
                hadiah = "1 Kotak Kopi";
                break;
            case "3":
                hadiah = "Sticker";
                break;
            case "4":
                hadiah = "Minyak Goreng";
                break;
```

Percabangan switch

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Percabangan switch/case</title>
</head>
<body>
    <script>

        var jawab = prompt("Kamu beruntung! Silahakn pilih hadiahmu dengan memasukan angka 1 sampai 5"
        var hadiah = "";

        switch(jawab){
            case "1":
                hadiah = "Tisu";
                break;
            case "2":
                hadiah = "1 Kotak Kopi";
                break;
            case "3":
                hadiah = "Sticker";
                break;
            case "4":
                hadiah = "Minyak Goreng";
                break;
```

```html
            case "5":
                hadiah = "Uang Rp 50.000";
                break;
            default:
                document.write("<p>Opps! anda salah pilih</p>");
        }

        if(hadiah === ""){
            document.write("<p>Kamu gagal mendapat hadiah</p>");
        } else {
            document.write("<h2>Selamat kamu mendapatkan " + hadiah + "</h2>");
        }
    </script>
</body>
</html>
```

Percabangan Bersarang (Nested)

Percabangan menggunakan operator ternary yang merupakan bentuk lain dari percabangan if/else.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Percabangan NESTED</title>
</head>
<body>
    <script>
        var username = prompt("Username:");
        var password = prompt("Password:");

        if(username == "rahmatfauzi"){
            if(password == "kopi"){
                document.write("<h2>Selamat datang pak bos!</h2>");
            } else {
                document.write("<p>Password salah, coba lagi!</p>");
            }
        } else {
            document.write("<p>Bos, Anda tidak terdaftar!</p>");
        }
    </script>
</body>
</html>
```

PERULANGAN / LOOPING DALAM JAVASCRIPT

Percabangan Bersarang (Nested)

Perulangan akan membantu kita mengeksekusi kode yang berulang-ulang, berapapun yang kita mau. Ada lima macam bentuk perulangan di Javascript. Secara umum, perulangan ini dibagi dua. Yaitu: counted loop dan uncounted loop