VERSION 1.3 5 Oktober 2024



[PEMROGRAMAN WEB]

Modul 2 - Javascript

DISUSUN OLEH:
OGYA ADYATMA PUTRA
GERALDI NATHAN TOMMY SAPUTRA

DIAUDIT OLEH: AMINUDIN, S.KOM., M.CS.

LAB. INFORMATIKA
UNIVERSITAS MUHAMMADIYAH MALANG

[PEMROGRAMAN WEBSITE]

PERSIAPAN MATERI

https://www.petanikode.com/tutorial/javascript/

https://www.w3schools.com/js/default.asp

https://www.geeksforgeeks.org/javascript/

TUJUAN

Mahasiswa mampu memahami konsep dasar dari pemrograman website

TARGET MODUL

- 1. Mahasiswa dapat memahami dasar-dasar javascript dalam perancangan website
- 2. Mahasiswa mampu menerapkan javascript ke dalam perancangan website

PERSIAPAN SOFTWARE/APLIKASI

Hardware dan Infrastruktur

- Laptop/PC
- Koneksi Internet

Software

• Text Editor Visual Studio Code (Recommended)

Extension (VS Code)

- Live Server | Ritwick Dey
- Prettier Code Formatter | Prettier

Programming Language

Javascript

MATERI

PENGENALAN JAVASCRIPT

Javascript merupakan sebuah bahasa pemrograman populer yang digunakan untuk membuat website interaktif dan dinamis, dimana konten didalamnya dapat bergerak dan tampil secara otomatis tanpa harus dimuat ulang manual oleh pengguna. Javascript juga merupakan salah satu fundamental yang harus dipahami untuk menjadi seorang web developer. Javascript sendiri tidak luput dari apa yang sudah yang sudah dipelajari di modul 1, dimana pada modul 1 mencoba tentang tentang HTML dan CSS. Nah sekarang akan menerapkan HTML CSS beserta dengan javascript.

Dalam penulisan javascript terdapat beberapa perbedaan dibanding bahasa pemrograman lainnya seperti yang sudah dipelajari. Karena pada penulisan javascript itu akan berfokus pada hasil yang akan ditampilkan di website.

Syntax Java

```
class HelloWorld {
   public static void main(String[] args) {
      System.out.println("Hello World");
   }
}
```

Syntax C

```
#include <stdio.h>
int main(){
printf("Hello World");

return 0;
}
```

Syntax Javascript

dapat dilihat beberapa perbedaan dari penulisan javascript dimana ketika menampilkan "hello world" pada bahasa c atau java dilakukan pembuatan class terlebih dahulu untuk menampilkan output "hello world" sedangkan dalam penggunaan javascript dapat dilakukan di dalam tag html menggunakan "<script></script>" untuk menuliskan logika pemrograman.

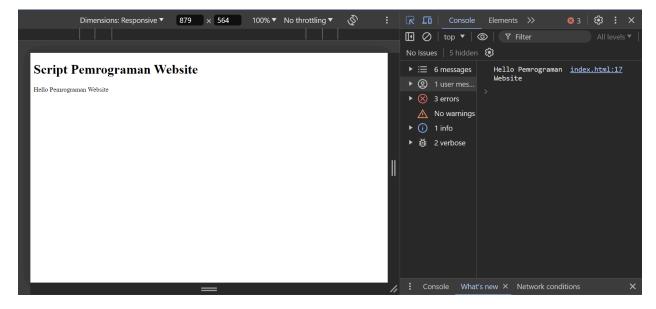
Setelah melihat perbedaan syntax para beberapa bahasa pemrograman diatas, pada javascript sendiri dapat menerapkan dengan cara yang berbeda. Jika melihat kembali contoh syntax javascript diatas, syntax tersebut dinamakan embed script. Embed Script sendiri merupakan penulisan javascript dalam file html menggunakan tag element "<script>". Kita dapat juga menulisnya dengan file yang berbeda agar lebih rapi seperti menerapkan styling css pada file yang berbeda dengan file html.

Penulisan Javascript secara terpisah

```
1 // Script.js
2
3 document.writeln("Hello World");
4
```

Kalau kita perhatikan atau mungkin sudah pernah belajar sebelumnya mengenai javascript, contoh penulisan syntax diatas berbeda dari yang biasanya kita tahu, karena pada dasarnya kita biasa menggunakan console.log bukan document.writeln. Perbedaan yang dapat dilihat adalah document.writeln akan menampilkan sebuah script yang sudah dibuat pada halaman website, berbeda dengan console.log dimana kita akan melihat pesan tersebut dengan melakukan inspect element terlebih dahulu seperti contoh dibawah ini.

```
1 <!DOCTYPE html>
    <html Lang="en">
      <head>
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>Document</title>
      </head>
      <body>
        <h1>Script Pemrograman Website</h1>
       <script>
          document.writeln("Hello Pemrograman Website");
          console.log("Hello Pemrograman Website");
        </script>
      </body>
    </html>
```



SYNTAX JAVASCRIPT

Syntax Javascript adalah aturan penulisan kode yang benar dalam bahasa pemrograman Javascript agar dapat dimengerti dan dijalankan oleh browser atau runtime environment seperti Node.JS.

1. Variabel

```
1  var x = 5;
2  Let y = 10;
3  const z = 15;
4  console.log(x, y, z);
```

2. Tipe Data

```
1  // TipeData
2  Let number = 100; // Number
3  Let text = "Hello"; // String
4  Let isTrue = true; // Boolean
5  Let obj = { name: "John", age: 30 }; // Object
6  Let arr = [1, 2, 3, 4, 5]; // Array
```

3. Function

```
(!DOCTYPE html>
(html lang="en")
(head>
(meta charset="UTF-8" />
(meta name="viewport" content="width=device-width, initial-scale=1.0" />
(title>Conton Fungsi JavaScript</title>
(head>
(head>
(body>
(!-- Tombol untuk memicu fungsi JavaScript -->
(button onclick="greet('Mahasiswa')">Klik untuk Menyapa</button>

(!-- Script JavaScript -->
(script>
// Definisi fungsi greet
function greet(name) {
    alert("Hello, " + name + "! Selamat belajar JavaScript!");
    }

    </script>
(/body>
(/ntml>)
```

4. Operator

```
1  // Operator
2  Let sum = 5 + 10; // Aritmatika
3  console.log(sum);
4  Let isEqual = 5 == 5; // Perbandingan
5  console.log(isEqual);
6  Let isBothTrue = true && false; // Logika
7  console.log(isBothTrue);
8  Let assignValue = 10; // Assignment
9  console.log(assignValue);
```

5. Array

```
1 Let car = ["BMW", "VW", "Mercy"];
2 console.log(car);
3 console.log(car[1]);
```

6. Objek

Pembuatan objek pada javascript dapat dilakukan seperti ini dimana person memiliki tiga properti yaitu menyimpan "nama", "umur" dan method "greed" dimana dalam method ini akan menampilkan sebuah pesan yang akan mengambil "name" dari objek person.

```
1 // Objek
2 Let person = {
3    name: "Jody",
4    age: 21,
5    greet: function () {
6       console.log("Hello, nama saya" + this.name + ", umur saya " + this.age);
7    },
8 };
9 person.greet();
```

7. Perulangan

Perulangan for pada kode di bawah dimulai dengan inisialisasi variabel "i" ke 0 dan terus berjalan selama nilai "i" kurang dari 5, dengan menambahkan 1 ke "i" setiap kali iterasi selesai. Pada setiap iterasi, nilai "i" akan dicetak ke konsol menggunakan console.log(i), menghasilkan output dari 0 hingga 4.

```
1 for (i = 0; i < 5; i++) {
2  console.log(i);
3 }
4</pre>
```

8. If Else

```
1  Let num = 10;
2
3  if (num >= 0 && num <= 10) {
4    console.log(num);
5  } else {
6    console.log("Nilai salah");
7  }</pre>
```

Document Object Manipulation

Document Object Manipulation adalah proses memanipulasi struktur, konten dan gara dari dokumen web menggunakan Javascript melalui Document Object Model. DOM adalah antarmuka pemrograman yang mewakili dokumen HTML atau XML sebagai pohon objek yang dapat diakses dan dimodifikasi. Dengan Document Object Manipulation, pengembang dapat mengubah elemen-elemen halaman web secara dinamis, menambah atau menghapus elemen. mengubah atribut serta memperbarui konten tanpa harus memuat ulang halaman. Beberapa komponen utama dalam Document Object Manipulation

Mengakses Elemen DOM	getElementById("content")getElementsbyClassName("content")querySelector("content")
Mengubah Konten Elemen	- InnerText("content") - textContent("content") - innerHTML("content")
Mengubah Atribut Elemen	- setAttribute("content") - getAttribute("content") - removeAttribute("content")
Event Handling	- addEventListener

• getElementByld()

```
cbody>
cp id="myParagraph">Ini adalah paragraf.
cscript>
// Mengakses elemen berdasarkan ID
const paragraph = document.getElementById("myParagraph");
console.log(paragraph.innerText); // Output: "Ini adalah paragraf."
</script>
c/body>
```

• getElementsbyClassName("content")

```
cbody>
cp class="myClass">Paragraf pertama.
cp class="myClass">Paragraf kedua.
cscript>
// Mengakses elemen-elemen berdasarkan kelas
const elements = document.getElementsByClassName("myClass");
console.log(elements[0].innerText); // Output: "Paragraf pertama."
</script>
c/script>
c/body>
```

• querySelector("content")

```
cbody>
div class="container">Ini adalah kontainer.</div>
cscript>
// Mengakses elemen menggunakan selector CSS
const container = document.querySelector(".container");
console.log(container.innerText); // Output: "Ini adalah kontainer."
</script>
</body>
```

• InnerText("content")

textContent("content")

• innerHTML("content")

```
cbody>
cdiv>konten lama</div>
div id="container">Konten baru</div>
cscript>
// Mengubah HTML dalam elemen
const container = document.getElementById("container");
container.innerHTML = "<strong>Konten baru 1</strong>";
</script>
c/body>
```

• setAttribute("content")

```
cbody>
cimg id="myImage" src="image1.jpg" alt="Gambar 1" />
script>
// Mengubah atribut elemen
const image = document.getElementById("myImage");
image.setAttribute("src", "image2.jpg");
</script>
</body>
```

getAttribute("content")

```
choody>
ca id="myLink" href="https://example.com">Kunjungi situs</a>
cscript>
// Mendapatkan atribut elemen
const link = document.getElementById("myLink");
const hrefValue = link.getAttribute("href");
console.log(hrefValue); // Output: "https://example.com"
</script>
c/body>
```

• removeAttribute("content")

```
chody>
cinput id="myInput" type="text" placeholder="Isi teks di sini" />
script>
// Menghapus atribut dari elemen
const input = document.getElementById("myInput");
input.removeAttribute("placeholder");

//script>

c/script>
```

addEventListener

```
cbody>
cbutton id="myButton">Klik Saya!</button>
cscript>

// Menambahkan event listener untuk menangani klik
const button = document.getElementById("myButton");
button.addEventListener("click", function () {
    alert("Tombol diklik!");
});
</script>
c/body>
```

Tabel dan contoh diatas merupakan beberapa komponen yang sering dipakai dalam tahap pengembangan website, tetapi masih banyak komponen lainnya yang perlu dieksplorasi dalam kebutuhan dalam pengembangan sebuah website Mengenai DOM. Gambar dibawah merupakan pengimplementasian DOM secara sederhana.

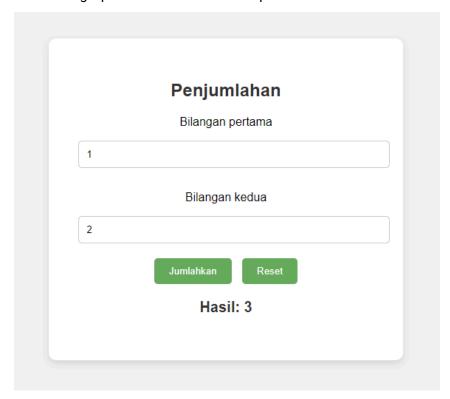
CODELAB

- 1. Lakukan upload github pada Tugas 2 Modul 1 minggu lalu dengan **nama branch** "**Modul 1**". Kemudian tulis nama, nim, dan url github yang sudah di upload pada spreadsheets dibawah ini.
 - **☐** List Pengumpulan Url Github Praktikum Web

Pastikan untuk mengisi sesuai dengan kelas praktikum.

2. Membuat operasi penjumlahan menggunakan HTML dan JAVASCRIPT

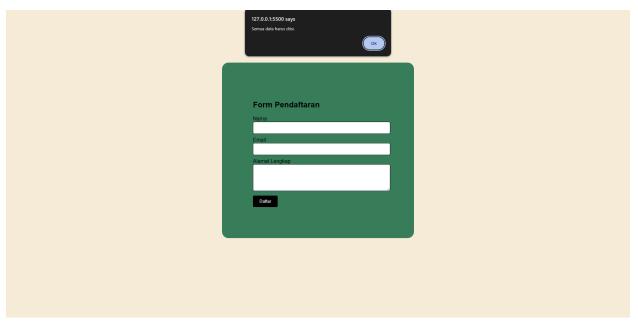
NOTE: mengimplementasikan css adalah opsional



3. Membuat sebuah form validation sederhana menggunakan HTML, CSS dan JAVASCRIPT



Note: Menampilkan alert ketika user menginputkan data kosong



TUGAS PRAKTIKUM

TUGAS 1

Buatlah sebuah calculator javascript dengan mengimplementasikan HTML, CSS dan JS dengan ketentuan sebagai berikut

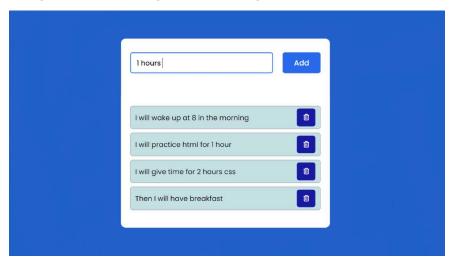
- Terdapat fungsi tambah, bagi, kurang, kali, pangkat dan modulus.
- Menerapkan fungsi Clear All apabila menekan tombol C / CE / AC.



Note: Gambar diatas hanyalah contoh salah satu contoh calculator, buatlah sesuai dengan kreativitas yang dimiliki.

Tugas 2

Membuat To Do List App dengan ketentuan dapat menambahkan data hingga menghapus dan mengedit data yang sudah ditambahkan.



Note: Gambar diatas hanyalah contoh salah satu contoh, buatlah sesuai dengan kreativitas.

Tugas 3

Upload tugas 1 dan 2 diatas kedalam github yang sudah dibuat pada codelab sebelumnya, dengan nama branch "Modul 2". Pastikan url github masih sama dengan yang tersedia pada spreadsheets.

List Pengumpulan Url Github Praktikum Web

19

KRITERIA & DETAIL PENILAIAN

Kriteria	Persentase Penilaian
Dapat membuat calculator dengan menerapkan responsive design	30%
Dapat membuat todo list app dengan menerapkan responsive design	30%
Tugas 3	10%
Pemahaman	30%