Jurnal Siber Multi Disiplin (JSMD)
E-ISSN : 2987-0380
P-ISSN : 2987-0402
Siber Nusantara Research
+62 812 1046 7572 https://research.e-siber.org research.siber.info@gmail.com Jalan Marina Indah Raya No. 1 Pantai Indah Kapuk, Kamal Muara, Penjaringan, Jakarta Utara, Indonesia.
DOI: https://doi.org/10.38035/jsmd.v2i2
Received: 02 Juni 2024, **Revised:** 11 Juni 2024, **Publish:** 18 Juli 2024
https://creativecommons.org/licenses/by/4.0/

Menjelajahi Dunia Web: Panduan Pemula Untuk Pemrograman Web

Fried Sinlae[1], Laudza Kalmany[2], Ruly Setiaji[3], M Syahrul[4]
[1]Universitas Bhayangkara Jakarta Raya, Bekasi, Indonesia, Fried.sinlae@dsn.ubharajaya.ac.id
[2]Universitas Bhayangkara Jakarta Raya, Bekasi, Indonesia, Laudzakalmanyy@gmail.com
[3]Universitas Bhayangkara Jakarta Raya, Bekasi, Indonesia, 202210715306@mhs.ubharajaya.ac.id
[4]Universitas Bhayangkara Jakarta Raya, Bekasi, Indonesia, 202210715019@mhs.ubharajaya.ac.id

Coresponding Author: fried.sinlae@dsn.ubharajaya.ac.id[1]

Abstract: In today's digital era, the ability to understand and harness the web is a highly valuable skill. This guide is aimed at beginners who wish to embark on their journey into web programming. We will cover the fundamentals of web programming, including HTML, CSS, and JavaScript. Additionally, we will explain key concepts such as responsive design and SEO optimization, crucial for website success. By grasping these fundamentals, readers will be able to confidently build and manage simple websites. The guide also includes supplementary resources and practical tips to deepen understanding and expand skills in navigating this dynamic web world.

Keyword: web programming, beginners, HTML, CSS, JavaScript, guide.

Abstract: Dalam era digital ini, kemampuan untuk memahami dan memanfaatkan web menjadi keterampilan yang sangat berharga. Panduan ini ditujukan untuk pemula yang ingin memulai perjalanan mereka dalam pemrograman web. Kami akan membahas dasar-dasar pemrograman web termasuk HTML, CSS, dan JavaScript. Selain itu, kami akan menjelaskan konsep-konsep penting seperti responsif design dan pengoptimalan SEO yang penting untuk kesuksesan website. Dengan memahami dasar-dasar ini, pembaca akan dapat membangun dan mengelola website sederhana dengan percaya diri. Panduan ini juga mencakup sumber daya tambahan dan tips praktis untuk memperdalam pemahaman dan memperluas keterampilan dalam menjelajahi dunia web yang dinamis ini.

Kata Kunci: pemrograman web, pemula, HTML, CSS, JavaScript, panduan.

PENDAHULUAN

Di era komputer dan internet yang semakin berkembang pesat seperti sekarang ini, keahlian dalam teknologi web telah menjadi hal yang sangat penting. Baik untuk ekspresi pribadi, usaha bisnis, maupun keterlibatan dalam komunitas, kemampuan untuk menguasai dan

memanfaatkan web secara efektif membuka banyak peluang.(Sinlae,2024) Panduan ini dirancang khusus untuk pemula yang ingin menjelajahi dunia pemrograman web. Dengan mempelajari elemen dasar seperti HTML, CSS, dan JavaScript, kami bertujuan untuk memberdayakan pembaca untuk membangun dan mengelola kehadiran mereka di web sendiri.

Memahami dasar-dasar pemrograman web bukan hanya praktis tetapi juga sangat penting bagi seseorang yang ingin menempatkan keberadaan mereka di dunia digital. Konsep-konsep seperti desain responsif, yang memastikan situs web dapat beradaptasi dengan baik di berbagai perangkat, dan optimasi SEO, yang meningkatkan visibilitas di mesin pencari, menjadi kunci untuk mencapai kesuksesan online. Melalui panduan ini, pembaca akan mendapatkan kepercayaan diri untuk membuat situs web sederhana namun fungsional, sebagai langkah awal untuk eksplorasi lebih lanjut dan pengembangan keterampilan(S.Sukri,2017).

Baik Anda bermimpi untuk membuat blog pribadi, menampilkan portofolio, atau mengembangkan aplikasi interaktif, memulai perjalanan ini dalam pemrograman web menjanjikan pencapaian kreatif dan pertumbuhan profesional.

Mari bergabung bersama kami dalam menjelajahi dunia pemrograman web yang dinamis dan selalu berkembang, di mana kreativitas bertemu dengan fungsionalitas dan inovasi tidak mengenal batas.

METODE

Pada penulisan jurnal ini, Jurnal yang berjudul "MENJELAJAHI DUNIA WEB: PANDUAN PEMULA UNTUK PEMROGRAMAN WEB" menyajikan panduan komprehensif bagi pemula untuk memahami dasar-dasar pemrograman web menggunakan bahasa pemrograman JavaScript, HTML, dan CSS melalui metode penelitian studi literatur. Penelitian ini mengumpulkan dan menganalisis berbagai referensi dari jurnal-jurnal ilmiah yang tersedia, memberikan gambaran mendalam tentang konsep, teknik, dan praktik terbaik dalam pengembangan web. Hasil dari penelitian ini diharapkan dapat menjadi sumber yang berguna bagi individu yang baru memulai perjalanan mereka dalam bidang pemrograman web.

HASIL DAN PEMBAHASAN

Pengenalan Pemrograman Web

Kata “Program” adalah asal dari istilah pemrograman. Program merupakan set perintah tertulis yang dibuat oleh programmer (pembuat program) menggunakan bahasa pemrograman tertentu; bagian yang dapat digunakan dari software. Pembuatan program disebut pemrograman. Jadi pemrograman ialah suatu proses yang mencakup Penulisan kode , Pengujian (debug),Pemeliharaan (maintenance) sebuah arahan yang ditulis dalam bentuk kode yang dapat dimengerti komputer.

Pemrograman web adalah proses membuat dan mengembangkan aplikasi atau situs web yang dapat diakses melalui internet. Proses ini melibatkan penggunaan berbagai bahasa pemrograman dan dan teknologi untuk merancang, membangun, dan memelihara komponen yang membentuk suatu situs web atau aplikasi web. Berikut ini adalah bagian utama dari pemrograman web:

1. Bahasa Pemrograman: Seperti HTML, CSS, JavaScript, PHP, Python, dan lain-lain. Setiap bahasa memiliki peran dan kegunaan masing-masing dalam membangun situs web atau aplikasi web.
2. Markah dan Gaya: HTML atau Hypertext Markup Language digunakan untuk membuat struktur dasar halaman web, sementara CSS atau Cascading Style Sheets digunakan untuk mengatur tampilan dan tata letak halaman.
3. Skrip Klien: JavaScript adalah bahasa yang dibuat untuk membuat interaksi dinamis di dalam halaman web, seperti validasi formulir, animasi, manipulasi DOM (Document Object Model), dan komunikasi dengan server.

4. Skrip Server: Bahasa pemrograman yaitu PHP, Python, Ruby, digunakan untuk mengembangkan bagian-bagian dari situs web yang berjalan di server. Mereka dapat memproses formulir, mengelola data, berinteraksi dengan basis data, dan menghasilkan halaman dinamis.
5. Basis Data: Untuk aplikasi web yang memerlukan penyimpanan dan pengelolaan data, basis data MySQL, PostgreSQL, MongoDB, digunakan untuk menyimpan informasi yang diolah dan diakses oleh situs web.
6. Kerangka Kerja (Framework): Ada juga kerangka kerja (framework) seperti Laravel (untuk PHP), Django (untuk Python), atau React (untuk JavaScript) yang menyediakan struktur dan alat untuk mempermudah pengembangan aplikasi web dengan cara tertentu.
7. Keamanan: Aspek penting dari pemrograman web adalah keamanan, termasuk perlindungan dari serangan seperti SQL injection, cross site scripting (XSS) serta kerentanan lainnya dapat mempengaruhi situs web atau aplikasi web.

Pemrograman web melibatkan penggunaan kombinasi teknologi ini untuk menciptakan pengalaman pengguna yang interaktif, intuitif, dan efisien saat menggunakan aplikasi atau mengakses situs web melalui browser web mereka.

Pengenalan Html

HTML dikenal juga HyperText Markup Language adalah bahasa yang umum dipakai untuk membuat halaman web. Ini adalah kombinasi dari Hypertext, yang mendefinisikan tautan setiap halaman web, dan bahasa markup, yang digunakan untuk mendefinisikan dokumen teks dalam tag untuk menyusun halaman web. Bahasa ini digunakan untuk memberi anotasi teks sehingga mesin dapat memahami dan memanipulasinya sesuai dengan itu. HTML memiliki tag yang dapat dibaca oleh orang dan memungkinkan manipulasi teks.

Secara sederhana, HTML adalah fondasi atau tulang punggung dari sebuah halaman web. Bayangkan HTML sebagai kerangka dasar yang menentukan struktur dan isi dari halaman web, sedangkan CSS dan JavaScript yang membuat halaman web tersebut menjadi lebih menarik dan interaktif.

Pembahasan dasar-dasar HTML:

1. Struktur Dasar Halaman Web
   - Penjelasan tentang struktur dasar sebuah halaman web HTML, yang dimulai dengan elemen `<html>` sebagai elemen utama yang mengelilingi seluruh konten halaman.
   - Elemen `<head>` yang berisi informasi meta-data tentang halaman seperti judul (`<title>`), meta tag, dan berkas eksternal yang dihubungkan seperti CSS dan JavaScript.
   - Elemen `<body>` yang berisi konten utama yang akan ditampilkan kepada pengguna, seperti teks, gambar, video, formulir, dan elemen-elemen interaktif lainnya.

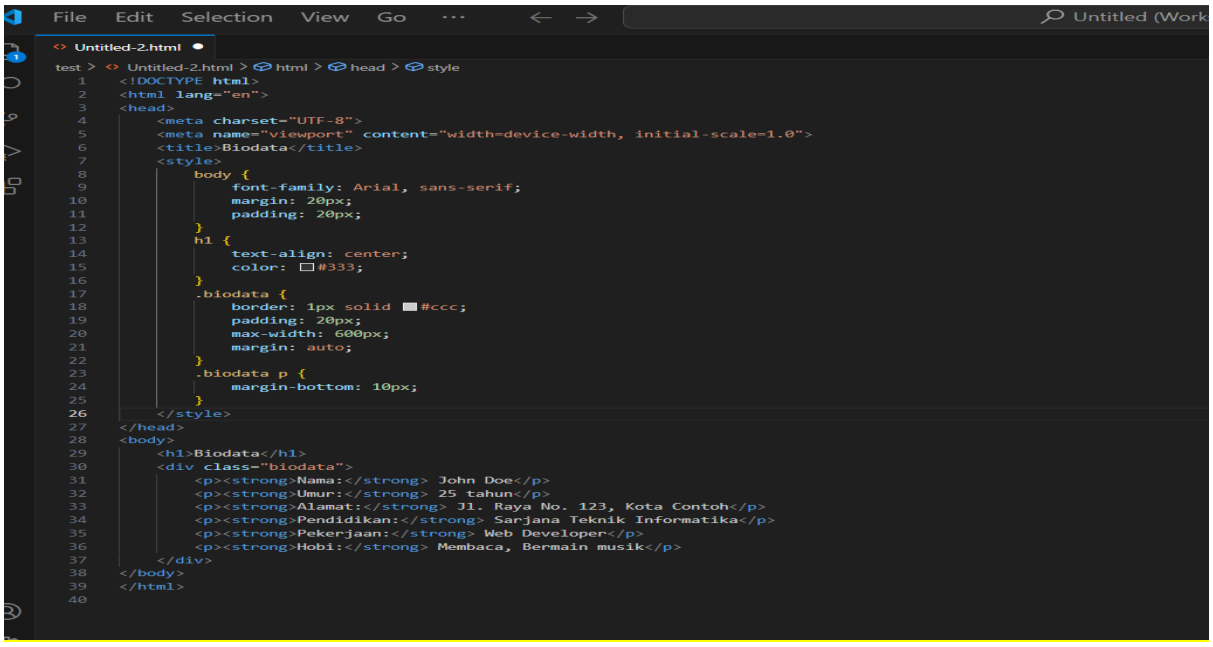
2. Elemen HTML
   - Pengenalan terhadap berbagai elemen HTML yang dibuat untuk menentukan struktur dan konten halaman web. Ini termasuk elemen teks seperti `<p>` untuk paragraf, `<h1>`-`<h6>` untuk judul-judul, `<a>` untuk tautan (hyperlink), `<img>` untuk menampilkan gambar, `<ul>`, `<ol>`, dan `<li>` untuk membuat daftar, serta banyak elemen lainnya.
   - Penjelasan tentang bagaimana elemen-elemen ini dapat disusun secara hierarkis dalam dokumen HTML untuk menciptakan struktur yang terorganisir dan mudah dimengerti.

3. Atribut HTML

- Atribut adalah komponen penting dari salah satu elemen HTML yang memberikan beberapa informasi tambahan tentang elemen tersebut atau mengatur perilaku dan tampilannya.
- Contoh atribut termasuk `src` untuk menentukan sumber gambar atau video dalam elemen `<img>` atau `<video>`, `href` untuk menentukan URL tautan dalam elemen `<a>`, `class` dan `id` untuk memberikan nama atau identitas khusus untuk pengaturan gaya CSS atau fungsi JavaScript.

4. Hubungan antara Elemen dan Atribut
- Penjelasan tentang bagaimana elemen dan atribut bekerja bersama untuk membentuk konten halaman web yang terstruktur dan berfungsi dengan baik. Misalnya, bagaimana sebuah gambar ditampilkan menggunakan elemen `<img>` dengan atribut `src` untuk menunjukkan lokasi gambar yang harus ditampilkan.

Materi ini akan memberikan dasar yang kuat bagi pembaca untuk memahami bagaimana HTML digunakan untuk membangun halaman web, baik untuk tujuan informasi, pendidikan, atau aplikasi web lainnya. Pemahaman yang kuat tentang struktur dasar HTML, elemen, dan atribut akan menjadi dasar yang sangat penting untuk mempelajari tahap selanjutnya dalam pengembangan web, termasuk CSS untuk styling dan JavaScript untuk interaktivitas. Dibawah ini kode HTML untuk membuat biodata:



Pengenalan CSS

CSS juga dikenal sebagai Cascading Style Sheets adalah bahasa yang dibuat untuk mengelola tampilan dan tata letak pada beberapa elemen di halaman web. Dengan dibantu CSS, Anda dapat mengendalikan warna teks, jenis font, jarak antar elemen, tata letak halaman, dan berbagai jenis lainnya. Berikut adalah beberapa konsep dasar dalam CSS:

- Selektor CSS: Selector CSS digunakan untuk memilih elemen-elemen HTML yang ingin Anda gayakan. Ada beberapa jenis selector, di antaranya:
- Selektor Elemen: Memilih elemen berdasarkan nama elemennya, misalnya `p` untuk paragraf.

```css
p {
    /* aturan gaya untuk paragraf */
}
```

- Selektor Kelas: Memilih elemen berdasarkan atribut kelasnya, ditandai dengan titik (`.`), misalnya `.kelas-saya`.

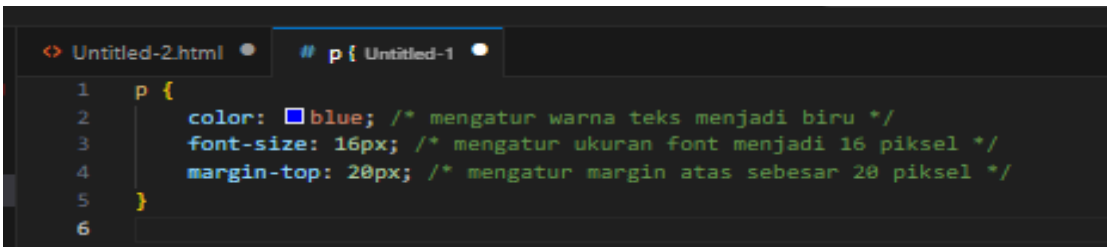
```css
.nama-kelas {
    /* aturan gaya untuk elemen dengan kelas 'nama-kelas' */
}
```

- Selektor ID: Memilih elemen berdasarkan ID uniknya, ditandai dengan pagar (`#`), misalnya `#header`.

```css
#nama-id {
    /* aturan gaya untuk elemen dengan ID 'nama-id' */
}
```

- Properti CSS

Properti CSS mengontrol aspek-aspek tertentu dari elemen yang dipilih. Setiap properti memiliki nama yang menggambarkan sifat yang diubah dan diikuti oleh nilai. Contoh properti dan nilai termasuk:



- Nilai CSS

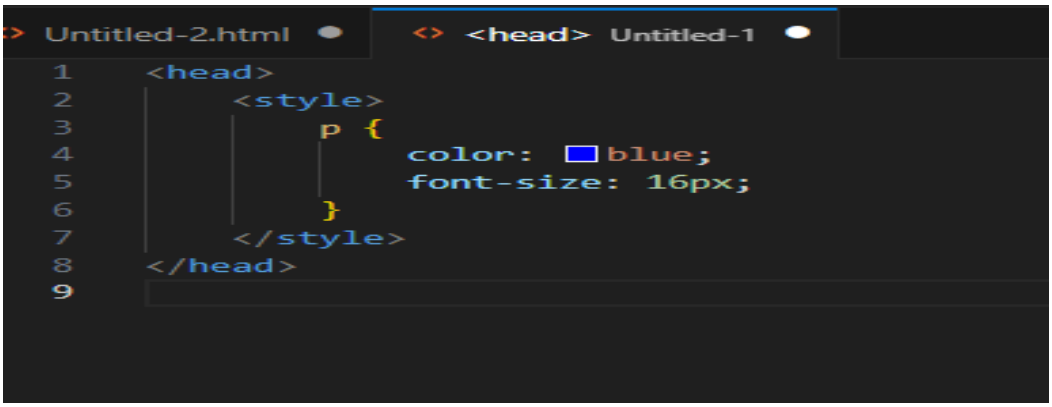
Nilai CSS diberikan kepada properti dan menentukan hasil akhir dari properti tersebut. Nilai dapat berupa angka (seperti ukuran font dalam piksel), warna (seperti nama warna atau kode warna), atau kata kunci tertentu (seperti `left`, `center`, atau `right` untuk tata letak).

Cascading dan Spesifisitas:

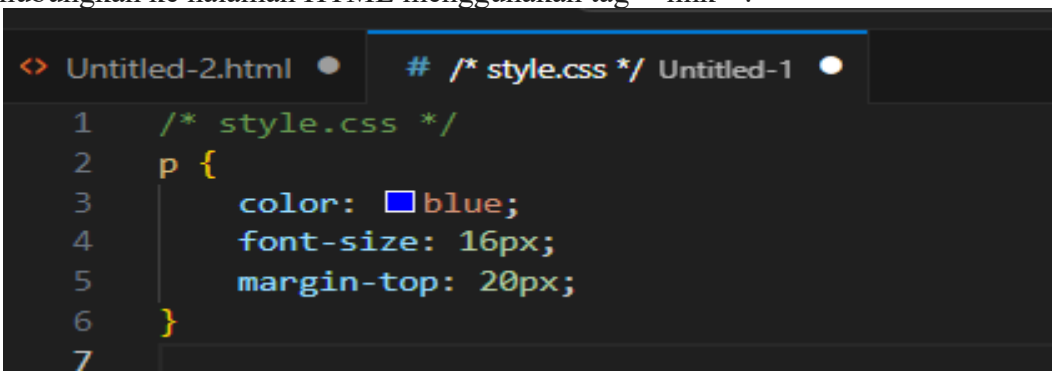
- **Cascading**: Istilah ini merujuk pada cara CSS menangani beberapa aturan gaya yang berlaku untuk elemen yang sama. Aturan gaya yang lebih spesifik atau yang berada lebih dekat dengan elemen tersebut dalam struktur dokumen akan mengambil keutamaan.
- **Spesifisitas**: Setiap selector memiliki tingkat spesifisitas yang menentukan aturan gaya mana yang akan diterapkan jika ada konflik. Spesifisitas dihitung berdasarkan kombinasi dari jenis selector, kelas, ID, dan pseudo-class yang digunakan.
- Cara Menerapkan Gaya ke Elemen HTML

Ada beberapa cara untuk menerapkan CSS ke halaman web:

1. CSS Internal : Gaya CSS ditulis di dalam tag `<style>` di dalam elemen `<head>` dari halaman HTML.



2. CSS Eksternal : Gaya CSS disimpan dalam file terpisah dengan ekstensi `.css` dan dihubungkan ke halaman HTML menggunakan tag `<link>`.



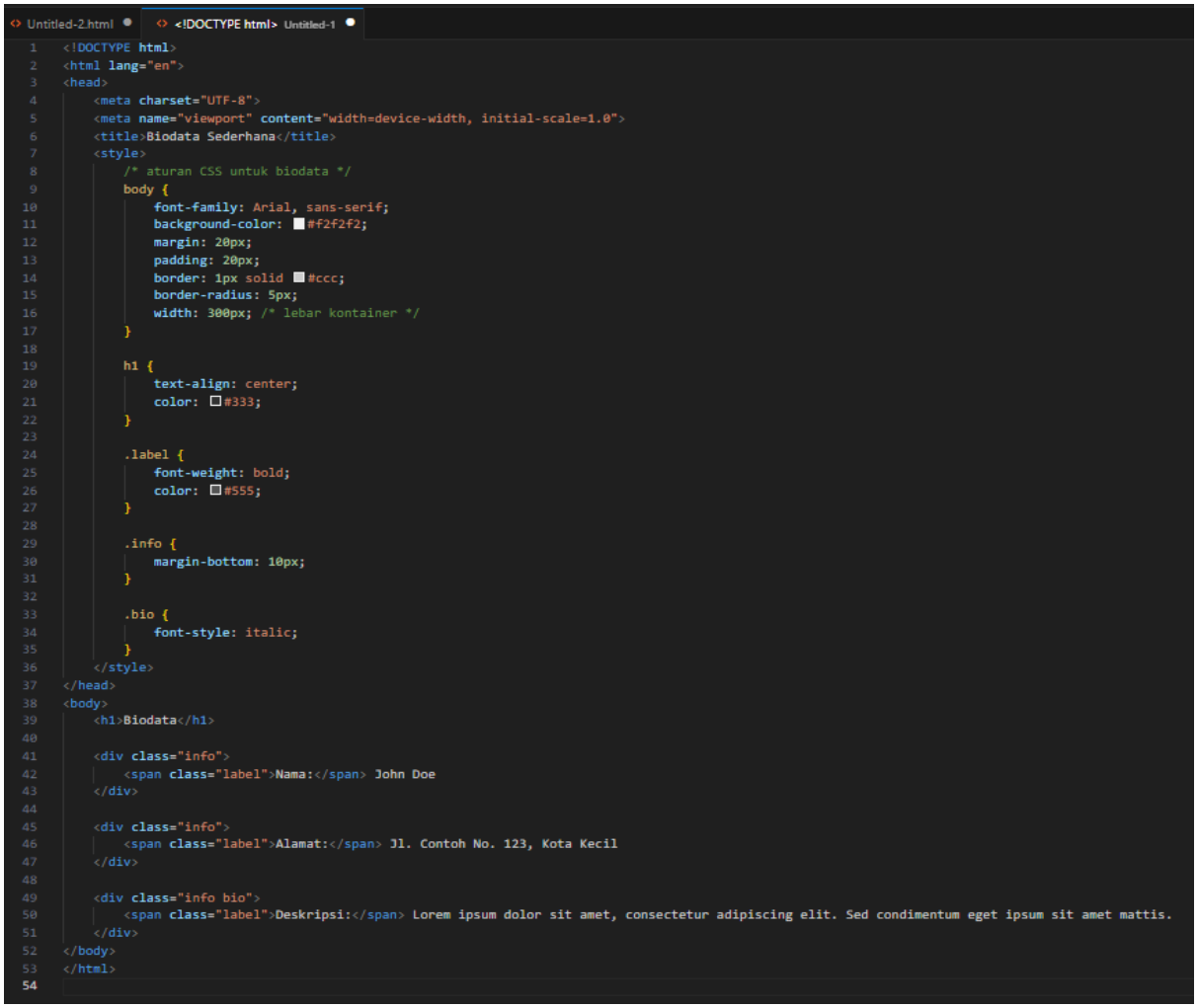
3. Inline CSS : Gaya CSS ditambahkan langsung ke dalam atribut `style` pada elemen HTML.

- Penggunaan CSS dalam Praktik

1. CSS digunakan bersama dengan HTML dan JavaScript untuk mengembangkan halaman web modern yang menarik dan responsif.
2. Pengembang web menggunakan CSS untuk mengatur tampilan halaman secara konsisten, meningkatkan pengalaman pengguna, dan memperbaiki aksesibilitas.
3. Framework CSS seperti Bootstrap, Tailwind CSS, atau Foundation menyediakan komponen dan gaya pradesain untuk mempercepat proses pengembangan.

- Contoh Penggunaan CSS

Berikut adalah contoh sederhana penerapan CSS pada sebuah biodata :



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Biodata Sederhana</title>
    <style>
        /* aturan CSS untuk biodata */
        body {
            font-family: Arial, sans-serif;
            background-color: #f2f2f2;
            margin: 20px;
            padding: 20px;
            border: 1px solid #ccc;
            border-radius: 5px;
            width: 300px; /* lebar kontainer */
        }

        h1 {
            text-align: center;
            color: #333;
        }

        .label {
            font-weight: bold;
            color: #555;
        }

        .info {
            margin-bottom: 10px;
        }

        .bio {
            font-style: italic;
        }
    </style>
</head>
<body>
    <h1>Biodata</h1>

    <div class="info">
        <span class="label">Nama:</span> John Doe
    </div>

    <div class="info">
        <span class="label">Alamat:</span> Jl. Contoh No. 123, Kota Kecil
    </div>

    <div class="info bio">
        <span class="label">Deskripsi:</span> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed condimentum eget ipsum sit amet mattis.
    </div>
</body>
</html>
```

Dengan menggunakan CSS, Anda dapat membuat halaman web lebih menarik, mudah dibaca, dan terstruktur dengan baik dengan mengontrol tampilan dan tata letak dari elemen-elemen HTML.

Pengenalan JavaScript

JavaScript adalah bahasa pemrograman yang populer dipakai saat membuat website, game dan aplikasi .JIka memahami dan menguasainya, Anda dapat membuat halaman website yang menarik atau mengembangkan game online berbasis web.selama proses pembuatan website, JavaScript memiliki kemampuan untuk membuat website tampak lebih menarik dan interaktif, dengan konten-konten dinamisnya.

Misalnya, konten dapat bergerak dan diperbarui secara real-time tanpa harus mengganti halaman web secara berulang.Sebagai contoh, Google Maps memiliki user experience yang luar biasa.Ketika mengaktifkan fitur penunjuk rute maka Google Maps dapat terus menginformasikan lokasi Anda secara realtime tanpa perlu mengulanginya. Pastinya tanpa JavaScript,hal ini tidak dapat dilakukan.

Variabel

Dalam JavaScript, penggunaan keyword var digunakan untuk memulai pembuatan variable. Contoh penggunaan keyword var dalam JavaScript sebagai berikut :

var title = “Latihan Javascript”;

Pada contoh di atas, kami membuat variable bernama title yang memiliki nilai berupa teks “Latihan Javascript”.
Var url = “https://neagahoster.co.id”;

var visitor count = 12345;

Dalam contoh diatas, kami membuat variabel url dan visitor_count yang memiliki nilai angka dan teks

Setiap variabel dalam contoh - contoh diatas akan dipisahkan dengan tanda titik koma.Selain itu,Spasi tidak digunakan saat menulis nama variable. Anda dapat menggunakan underscore atau huruf besar saat menggunakan dua kata.

Tipe Data

Setelah mempelajari variabel dalam bahasa ini, Anda harus tahu tipe data yang akan disimpan di variabel.

Ada beberapa contoh tipe data dalam Javascript:

1. String atau teks
2. Integer atau Number berupa bilangan bulat
3. Float berupa bilangan pecahan
4. Boolean berupa (true/false)
5. Object berupa semua tipe data

Semua tipe data JavaScript sebenarnya ditulis secara dinamis. Ini berarti bahwa, tidak seperti dalam bahasa pemrograman yang lainnya,anda tidak perlu menulis tipe data saat mengisi variable.

Operator

Maksud Operator disini adalah simbol yang dapat mewujudkan nilai baru.Berikut beberapa contoh operator yang bisa Anda gunakan di JavaScript:

1. Operator Aritmatika – Digunakan untuk menghitung nilai variabel angka. Terdiri dari simbol penjumlahan (+), pengurangan (-), perkalian (*), pemangkatan (**), pembagian (/)
2. Operator Penugasan – Digunakan untuk melakukan tugas pada variabel, seperti mengisi nilai variabel. Salah satu simbol penugasan adalah (=).
3. Operator Perbandingan – Operator relasi juga disebut sebagai operator perbandingan merupakan operator yang digunakan untuk membandingkan dua nilai. Hasilnya dari perbandingan ini akan menghasilkan nilai boolean ‘true’ dan ‘false’. Simbol operator ini diantaranya, (>), (<), (>=), dan sebagainya.
4. Operator Logika – Operator logika digunakan untuk melaksanakan operasi terhadap nilai boolean. Simbol dari operator ini adalah logika AND (&&), logika OR (||), dan negasi (!)

Struktur Kontrol

Struktur kontrol dalam JavaScript digunakan untuk mengatur alur eksekusi program berdasarkan kondisi atau melakukan pengulangan.

1. **Kondisional (if-else)**: Digunakan untuk menjalankan blok kode jika kondisi tertentu terpenuhi.

```javascript
let nilai = 80;

if (nilai >= 70) {
    console.log("Lulus");
} else {
    console.log("Tidak Lulus");
}
```

2. **Perulangan (for)**: Digunakan untuk mengulangi blok kode sejumlah tertentu.

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

3. **Perulangan (while)**: Mengulangi blok kode selama kondisi tertentu terpenuhi.

```javascript
let counter = 0;
while (counter < 5) {
    console.log(counter);
    counter++;
}
```

Fungsi Dasar

Fungsi digunakan untuk mengelompokkan kode yang dapat digunakan kembali dan memproses input untuk menghasilkan output.Dengan struktur kontrol dan fungsi dasar ini, Anda dapat mengatur alur eksekusi program dan mengorganisir kode secara lebih efisien dalam pengembangan aplikasi JavaScript.

```javascript
// Mendefinisikan fungsi
function sapa(nama) {
    console.log("Halo, " + nama + "!");
}

// Memanggil fungsi dengan argumen
sapa("John"); // Output: Halo, John!
sapa("Jane"); // Output: Halo, Jane!
```

Hasil Penggabungan HTML,CSS,JavaScript

Contoh kode HTML, CSS, dan JavaScript untuk membuat sebuah form biodata sederhana. Form ini akan memiliki input untuk nama, alamat, dan deskripsi diri, serta tombol untuk mengirimkan data. Form ini akan menampilkan alert dengan informasi yang dimasukkan setelah tombol "Submit" ditekan.

HTML (index.html)

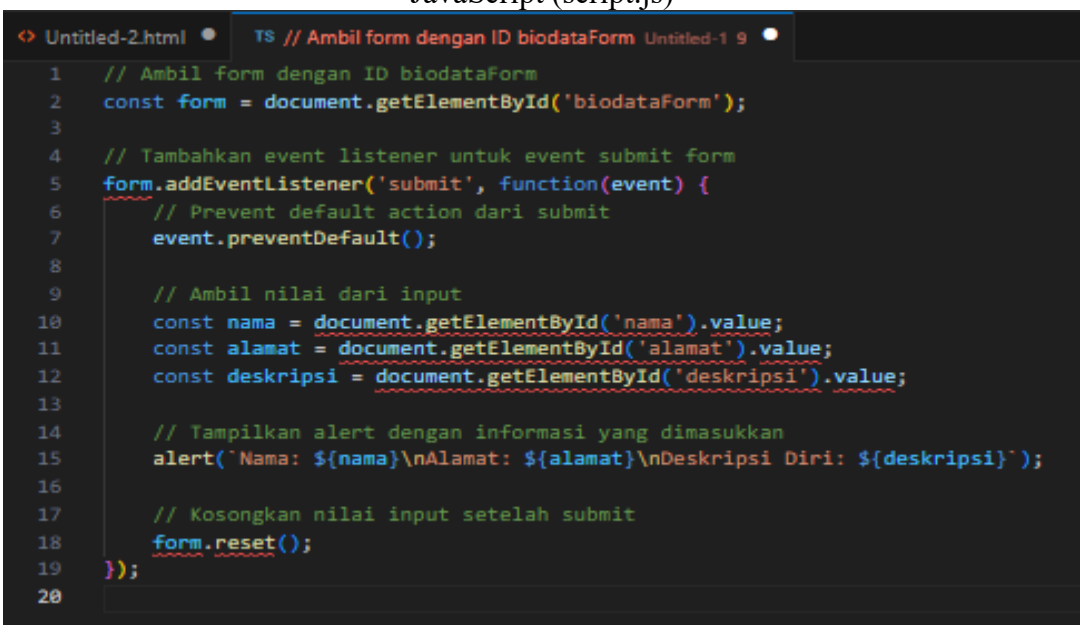
Untitled-2.html <!DOCTYPE html> Untitled-1

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Form Biodata</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
<div class="container">
    <h2>Form Biodata</h2>
    <form id="biodataForm">
        <label for="nama">Nama:</label><br>
        <input type="text" id="nama" name="nama" required><br>

        <label for="alamat">Alamat:</label><br>
        <textarea id="alamat" name="alamat" rows="4" required></textarea><br>

        <label for="deskripsi">Deskripsi Diri:</label><br>
        <textarea id="deskripsi" name="deskripsi" rows="4" required></textarea><br>

        <button type="submit">Submit</button>
    </form>
</div>
<script src="script.js"></script>
</body>
</html>
```

CSS (style.css)

```css
body {
    font-family: Arial, sans-serif;
    background-color: #f2f2f2;
    padding: 20px;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
}

.container {
    background-color: #fff;
    padding: 20px;
    border-radius: 5px;
    box-shadow: 0 0 10px rgba(0,0,0,0.1);
    width: 300px;
}

form {
    display: flex;
    flex-direction: column;
}

label {
    margin-bottom: 5px;
}

input, textarea {
    width: 100%;
    padding: 8px;
    margin-bottom: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}

button {
    background-color: #4CAF50;
    color: white;
    padding: 10px 15px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    font-size: 16px;
}

button:hover {
    background-color: #45a049;
}
```

JavaScript (script.js)



```javascript
// Ambil form dengan ID biodataForm
const form = document.getElementById('biodataForm');

// Tambahkan event listener untuk event submit form
form.addEventListener('submit', function(event) {
    // Prevent default action dari submit
    event.preventDefault();

    // Ambil nilai dari input
    const nama = document.getElementById('nama').value;
    const alamat = document.getElementById('alamat').value;
    const deskripsi = document.getElementById('deskripsi').value;

    // Tampilkan alert dengan informasi yang dimasukkan
    alert(`Nama: ${nama}\nAlamat: ${alamat}\nDeskripsi Diri: ${deskripsi}`);

    // Kosongkan nilai input setelah submit
    form.reset();
});
```

Penjelasan Singkat:

- **HTML**: Form terdiri dari input untuk nama (text), alamat (textarea), deskripsi diri (textarea), dan tombol "Submit". Setiap input memiliki atribut required untuk memastikan tidak ada yang kosong.
- **CSS**: Mengatur tata letak dan gaya visual form dan elemennya.
- **JavaScript**: Menggunakan event listener untuk menangkap event submit pada form. Ketika form disubmit, nilai dari input diambil, ditampilkan dalam sebuah alert, dan nilai input di-reset.

Dengan mengikuti kode yang dibuat di atas, Anda akan membuat dan mengatur form biodata yang sederhana namun efektif dalam hal tampilan dan fungsionalitasnya.

KESIMPULAN

Berdasarkan pembahasan diatas, terdapat kesimpulan bahwa dengan menggunakan HTML, CSS,dan JavaScript, dapat membuat sebuah Biodata berbasis web. Penggunaan HTML untuk membuat kerangka dari suatu tampilan Biodatanya, CSS, untuk mengatur tampilan, layout, dan juga gaya dari Form Biodata, JavaScript untuk menjalankan proses Menggunakan event listener untuk menangkap event submit pada form. Ketika form disubmit, nilai dari input diambil, ditampilkan dalam sebuah alert, dan nilai input di-reset.

REFERENSI

(Sinlae, 2024 PERKEMBANGAN CYBER DI ERA GLOBALISASI)

(S.Sukri,2017 Mendapatkan Peringkat Terbaik Website Pada Search Engine Dengan Metode Search Engine Optimization (SEO)

(M. R. P. Kurniadi, "Arti Pemrograman Menurut KBBI," Lectur Id, 21 Juli 2018.)

(Erawan, 2015 PEMROGRAMAN WEB)

(Andreas Bayu Putra, Arya Syahputra, Dandi Pratama, Dimas Firdaus, Fajar Rizki Nuriantoro, Ferry Andika Firmansyah, Handi Satria, Muhamad Fitrianto, Pebrialiyansyah, & Pebrianto. (2024). Pengenalan Dasar-Dasar HTML dan CSS Pada Siswa/I Yayasan Al-Qaaf Tangerang Banten Dengan Menggunakan Bahasa Pemprograman Visual Studio Code Serta Mampu Membuat Website Sederhana . *APPA : Jurnal Pengabdian Kepada Masyarakat* , *1* (6), 417–422. Diperoleh dari https://jurnalmahasiswa.com/index.php/appa/article/view/915)

(Saputra, A., & Astuti, Y. (2018). Analisis Pengaruh Struktur HTML Terhadap Rangking Search Engine Result Page. *Jurnal Mantik Penusa*, *2*(2).)

(Kaban, R., & Fajrillah. (2017). PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN DENGAN FRAMEWORK CSS BOOTSTRAP DAN WEB DEVELOPMENT LIFE CYCLE. *Jurnal Ilmiah Informatika* , *2* (1), 83-89. https://doi.org/10.35316/jimi.v2i1.454)

(Kaluvakuri, S., & Vadiyala, VR (2016). Memanfaatkan Potensi CSS: Referensi Lengkap untuk Styling Web. *Teknik Internasional* , *4* (2), 95-110.)

(Jihadi, H., & Syarabil, A. F. (2023). PERBANDINGAN REACT JS DAN VUE JS DALAM PENGEMBANGAN APLIKASI WEB INTERAKTIF: SEBUAH STUDI KOMPARATIF. *Jurnal Sistem Informasi Bisnis (JUNSIBI)*, *4*(2), 70-79.)