

LAPORAN TEORI

PEMROGRAMAN WEB



**SOFTWARE ENGINEERING
LABORATORY**

NAMA : Claudia Sherly Febriana Dewi
NIM : 202431160
KELAS : F
DOSEN : M Yoga Distra Sudirman, ST., MTI
NO.PC : 11
ASISTEN : 1. Abdur Rasyid Ridho
 2. Karina Distyara Putri Maulana
 3.

INSTITUT TEKNOLOGI PLN
TEKNIK INFORMATIKA

2025

A. CSS Fundamental

CSS (Cascading Style Sheets) adalah bahasa yang digunakan untuk mengatur tampilan visual dari halaman web. Dengan CSS, kita bisa mengubah warna, ukuran, layout, jarak antar elemen, dan banyak aspek estetika halaman.

1. Selector

Selector digunakan untuk memilih elemen HTML yang ingin diberikan style. Tipe selector:

- Tag selector: memilih berdasarkan nama tag
- Class selector: memilih berdasarkan class (.)
- ID selector: memilih berdasarkan id (#)
- Universal selector: memilih semua elemen (*)

2. Property

Property adalah atribut yang ingin diubah, misalnya:

- color: warna teks
- background-color: warna latar belakang
- font-size: ukuran teks
- margin, padding: jarak antar elemen
- border: garis tepi elemen

3. Value

Value adalah nilai dari property, misal:

- Warna: red, #f0f0f0, rgb(255,0,0)
- Ukuran: 16px, 1.5em
- Tipe border: solid, dashed

4. Inline CSS

Ditulis langsung di tag HTML. Kelebihannya cepat, mudah untuk percobaan kecil.

Kekurangannya sulit dikelola untuk banyak halaman

5. Internal CSS

Ditulis dalam tag <style> di <head>. Kelebihannya semua style ada di satu file HTML.

Kekurangannya sulit dipakai untuk banyak halaman

6. External CSS

Ditulis di file .css terpisah, di-link ke HTML. Kelebihannya mudah dikelola, style konsisten di banyak halaman. Kekurangannya perlu file tambahan dan koneksi agar diload

B. Bootstrap Components

Bootstrap adalah framework CSS dan JS yang memudahkan pembuatan halaman web responsive dan modern tanpa menulis banyak kode dari awal.

1. Navbar: menu navigasi horizontal dan responsive secara otomatis.

Contoh:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>contoh navbar</h1>
10  <nav class="navbar navbar-expand-lg navbar-light bg-light">
11    <a class="navbar-brand" href="#">Brand</a>
12    <div class="collapse navbar-collapse">
13      <ul class="navbar-nav">
14        <li class="nav-item"><a class="nav-link" href="#">Home</a></li>
15        <li class="nav-item"><a class="nav-link" href="#">About</a></li>
16      </ul>
17    </div>
18  </nav>
19
20 </body>
21 </html>
22

```

← ⏪ ⓘ File C:/Users/claud/Documents/PEMWEB/Laporan2/contoh%20s.html

contoh navbar

Brand

- [Home](#)
- [About](#)

2. Card: kotak konten yang rapi dengan header, body, footer.

Contoh:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>contoh card</h1>
10  <div class="card" style="width: 18rem;">
11    <div class="card-body">
12      <h5 class="card-title">Judul Card</h5>
13      <p class="card-text">Isi konten card di sini.</p>
14      <a href="#" class="btn btn-primary">Klik</a>
15    </div>
16  </div>
17
18 </body>
19 </html>

```

← ⏪ ⓘ File C:/Users/claud/Documents/PEMWEB/Laporan2/contoh%20s.html

contoh card

Judul Card

Isi konten card di sini.

[Klik](#)

3. Container, Row & Col (Grid): layout responsif.

Contoh:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>contoh Container, Row & Col</h1>
10  <div class="container">
11    <div class="row">
12      <div class="col-md-6">Kolom 1</div>
13      <div class="col-md-6">Kolom 2</div>
14    </div>
15  </div>
16 </body>
17 </html>

```

← ⏪ ⓘ File C:/Users/claud/Documents/PEMWEB/Laporan2/contoh%20s.html

Contoh Container, Row & Col

Kolom 1
Kolom 2

4. Button: tombol interaktif.

Contoh:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>Contoh Button</h1>
10  <button class="btn btn-success">Sukses</button>
11  <button class="btn btn-danger">Danger</button>
12
13 </body>
14 </html>
15

```

← ⏪ ⓘ File C:/Users/claud/Documents/PEMWEB/Laporan2/contoh%20s.html

Contoh Button

[Sukses](#) [Danger](#)

5. Alert: pesan peringatan atau informasi.

Contoh:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>Contoh Button</h1>
10  <div class="alert alert-warning" role="alert">
11    Ini alert peringatan!
12  </div>
13 </body>
14 </html>
15

```

← ⏪ ⓘ File C:/Users/claud/Documents/PEMWEB/Laporan2/contoh%20s.html

Contoh Alert

Ini alert peringatan!

C. JavaScript Dasar

JavaScript (JS) adalah bahasa pemrograman yang dijalankan di browser untuk membuat halaman web interaktif. JS memungkinkan kita mengubah isi halaman, menangkap aksi pengguna, dan menjalankan logika dinamis.

1. Variabel

- Variabel adalah wadah untuk menyimpan data sementara selama program berjalan.
- Let: variabel yang bisa diubah nilainya
 - Const: variabel konstan, nilainya tidak bisa diubah
 - Var: variabel lama, bisa diubah, punya scope berbeda

2. Operator

Operator digunakan untuk memanipulasi nilai dalam variable.

- Aritmatika → + - * / %
- Perbandingan → == === != !== > < >= <=
- Logika → && (dan), || (atau), ! (not)

3. Kondisi

Kondisi digunakan untuk mengambil keputusan berdasarkan nilai tertentu.

- if, else if, else: blok kode dijalankan jika kondisi terpenuhi
- switch: memilih dari beberapa opsi nilai

4. Looping/Perulangan

Digunakan untuk menjalankan kode berulang kali.

- For: perulangan dengan jumlah tertentu
- While: perulangan selama kondisi terpenuhi
- do...while: hamper sama dengan while, tapi dijalankan minimal 1 kali

5. Fungsi

Fungsi adalah sekumpulan kode yang bisa dipanggil berkali-kali, bisa menerima input (parameter) dan mengembalikan output.

D. DOM Manipulation

DOM (Document Object Model) adalah struktur pohon elemen HTML yang bisa dimanipulasi dengan JavaScript.

- getElementById: mengambil elemen berdasarkan id
- querySelector / querySelectorAll: mengambil elemen menggunakan selector CSS
- Event Listener: menangkap aksi pengguna
- innerHTML: mengubah isi elemen
- Style Manipulation: mengubah style elemen secara dinamis