

LAPORAN PRAKTIKUM  
PERANCANGAN DAN PEMROGRAMAN WEB

MODUL 2  
(HTML DAN CSS)



Oleh:

(Rifqi M Ramdani)

(2311104044)

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK  
DIREKTORAT KAMPUS PURWOKERTO  
UNIVERSITAS TELKOM  
2025

# BAB I

## PENDAHULUAN

### 1.1 Dasar Teori

## HTML

**HTML (HyperText Markup Language)** adalah bahasa markup standar yang digunakan untuk membuat dan menyusun halaman web. HTML bekerja dengan menggunakan serangkaian *tag* (penanda) untuk menentukan elemen-elemen seperti teks, gambar, tautan, tabel, formulir, dan struktur layout pada sebuah halaman web.

**Contoh sederhana HTML:**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Judul Halaman</title>
</head>
<body>
  <h1>Hello Word</h1>
```

```
    <p>Ini adalah paragraf pertama saya di HTML.</p>
  </body>
</html>
```

**Keterangan:**

- HTML bukan bahasa pemrograman, melainkan bahasa markup untuk menyusun struktur halaman web.
- HTML biasanya digunakan bersama CSS (untuk desain/tampilan) dan JavaScript (untuk interaktivitas).

Tag pada HTML adalah elemen dasar yang digunakan untuk membangun dan menyusun struktur halaman web. Setiap *tag* biasanya ditulis dalam tanda kurung sudut < > dan memiliki pasangan pembuka dan penutup.

Berikut beberapa **tag dasar HTML yang sering digunakan**:

Tag	Fungsi
<!DOCTYPE html>	Menyatakan jenis dokumen HTML
<html>	Pembungkus seluruh isi halaman HTML
<head>	Berisi informasi metadata (title, CSS, script, dll)
<title>	Menentukan judul halaman (muncul di tab browser)
<link>	Menyisipkan file eksternal (misalnya CSS)
<script>	Menyisipkan atau memanggil kode JavaScript
<body>	Tempat seluruh konten yang ditampilkan di layar
<h1> - <h6>	Judul/heading, dari terbesar (h1) sampai terkecil (h6)
<p>	Paragraf
 	Baris baru (line break)
<hr>	Membuat garis horizontal (pemisah)
<a>	Membuat hyperlink
<img>	Menampilkan gambar
<ul>	Daftar tak berurut (bullet list)
<ol>	Daftar berurut (number list)
<li>	Item dalam daftar
<div>	Membuat blok untuk layout

<span>	Membuat inline text (penekanan kecil)
<table>	Membuat tabel
<tr>	Baris pada tabel (Table Row)
<td>	Sel pada tabel (Table Data)
<th>	Sel header tabel (tebal, rata tengah) (Table Header)
<form>	Membuat formulir untuk input data
<input>	Elemen input (teks, email, password, checkbox, radio, dll)
<textarea>	Kotak teks untuk input panjang
<select>	Dropdown pilihan
<option>	Opsi dalam dropdown
<button>	Tombol aksi
<label>	Label untuk elemen input
<header>	Bagian header dari halaman (HTML5 semantic)
<footer>	Bagian footer dari halaman
<nav>	Bagian navigasi menu
<section>	Bagian konten (section)
<aside>	Konten samping (sidebar)

```

<body>
  <h1>Selamat Datang</h1>
  <p>Ini adalah paragraf pertama saya.</p>
  <a href="https://www.google.com">Kunjungi Google</a>
  <br>
  
</body>

```

## Form pada HTML

Tag <form> pada HTML digunakan untuk membuat formulir input yang memungkinkan pengguna mengirimkan data ke server, seperti pada halaman login, pendaftaran, pencarian, atau pemesanan.

### a. Struktur Dasar <form>

```
<form action="proses.php" method="post">
  <label>Nama:</label>
  <input type="text" name="nama">

  <label>Email:</label>
  <input type="email" name="email">
  <button type="submit">Kirim</button>
</form>
```

### b. Atribut Penting

Atribut	Fungsi
action	Alamat tujuan data dikirim (misalnya file PHP)
method	Metode pengiriman data: GET atau POST
enctype	Format pengiriman data (biasanya untuk upload file)

### c. Jenis Input yang Sering Digunakan

Tag / Type	Fungsi
<input type="text">	Input teks biasa
<input type="email">	Input email
<input type="password">	Input password
<input type="radio">	Pilihan satu opsi

## CSS

CSS (Cascading Style Sheets) adalah bahasa yang digunakan untuk mengatur tampilan dan gaya dari elemen-elemen HTML. Jika HTML berfungsi untuk menyusun struktur sebuah halaman web, maka CSS berfungsi untuk menghias halaman tersebut agar lebih menarik, rapi, dan nyaman dilihat.

### 1.6. Jenis Penulisan CSS

- a. **Inline CSS** → ditulis langsung dalam elemen HTML menggunakan atribut `style`.

```
<p style="color:red;">Teks ini berwarna merah</p>
```

- b. **Internal CSS** → ditulis dalam tag `<style>` di bagian `<head>`.

```
<style>
  h1 {
    color: green;
  }
</style>
```

- c. **External CSS** → ditulis di file `.css` terpisah, lalu dihubungkan dengan `<link>`.

```
<link rel="stylesheet" href="style.css">
```

Kata *cascading* berarti “mengalir ke bawah”. Artinya jika ada beberapa aturan CSS yang berlaku pada elemen yang sama, browser akan memilih aturan berdasarkan **prioritas**:

1. Inline CSS (paling kuat)

## Contoh Properti CSS

### Properti CSS yang Sering Digunakan

#### 1. Teks

```
p {  
  color: black;  
  /* warna teks */  
  font-family: Arial;  
  /* jenis huruf */  
  font-size: 16px;  
  /* ukuran huruf */  
  font-weight: bold;  
  /* tebal */  
  text-align: justify;  
  /* rata kiri-kanan */  
  line-height: 1.5;  
  /* jarak antar baris */  
}
```

#### 2. Background

```
body {  
  background-color: #f4f4f4;  
  /* warna latar */  
}  
  
header {  
  background-color: navy;  
  /* latar belakang */  
  color: white;  
  /* warna teks */  
}
```

#### 3. Box Model

```
div {  
  width: 300px;  
  /* lebar */  
  height: 150px;  
  /* tinggi */  
  margin: 20px;  
  /* jarak luar */  
  padding: 15px;  
  /* jarak dalam */  
  border: 2px solid black;  
  /* garis tepi */  
  border-radius: 8px;  
  /* sudut melengkung */  
}
```

```
}
```

#### 4. Layout

```
nav ul li {  
    display: inline-block;  
    /* menu sejajar */  
    margin-right: 15px;  
    /* jarak antar menu */  
}
```

#### 5. Efek Hover (interaktif)

```
a {  
    color: blue;  
    text-decoration: none;  
    /* hilangkan garis bawah */  
}  
  
a:hover {  
    color: red;  
    /* berubah saat diarahkan */  
}
```

### 1.2 Tujuan

- A. Menyusun dan menampilkan konten web seperti teks, gambar, video, tabel, atau formulir.
- B. Mendefinisikan struktur halaman menggunakan elemen seperti header, footer, paragraf, daftar, dan lainnya.
- C. Menghubungkan halaman satu dengan lainnya melalui hyperlink.
- D. Menjadi fondasi bagi teknologi web lainnya, seperti CSS (untuk tampilan) dan JavaScript (untuk interaksi).
- E. Memungkinkan mesin pencari (seperti Google) memahami konten halaman melalui struktur tag yang terorganisir.



Jadi, tanpa HTML, sebuah halaman web tidak akan memiliki bentuk atau isi yang bisa ditampilkan dan dipahami oleh browser maupun pengguna.

### 1.3 Manfaat

Manfaat dari praktikum dan penguasaan HTML dan CSS adalah:

- Membangun Fondasi Web: Memungkinkan praktikan untuk membangun dasar dari setiap situs web atau aplikasi online.
- Keterampilan Frontend Dasar: Memberikan keterampilan dasar yang penting untuk menjadi developer frontend atau web designer.
- Pemahaman Struktur Konten: Memahami bagaimana konten (teks, gambar, video) distrukturkan dan ditampilkan oleh browser kepada pengguna.
- Desain Responsif dan Rapi: Mampu menerapkan gaya menggunakan CSS, termasuk teknik layout modern seperti Flexbox dan Grid, untuk membuat tampilan yang profesional dan adaptif.
- Optimasi Mesin Pencari (SEO): Memahami cara menyusun tag HTML secara terorganisir, yang membantu mesin pencari (seperti Google) memahami konten halaman.

## BAB II

### HASIL PRAKTIKUM

#### 2.1 SOAL

- a) Buat layout html seperti pada Gambar 1 dan terdapat 4 menu utama (Beranda, Produk, Member, Login)

Header		
Beranda Produk Member Login		
Populer	Konten Dinamis	Iklan
Footer		

Gambar 1 Layout HTML halaman profile

## Source Code

profile.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1"
/>
  <title>Pertemuan 2</title>
</head>
<body>
  <div class="Haha"></div>
  <h1>Pertemuan 2</h1>
  <p>Halo, ini adalah pertemuan kedua saya.</p>
  <p>55555</p>
  <p><a href="https://www.google.com">Google</a></p>

  
  <h1>Menu Makanan</h1>
  <ol>
    <li>Ayam Filet</li>
    <li>Seblak</li>
    <li>Nasi Goreng</li>
    <li>Jamur Crispy</li>
  </ol>

  <form action="">
    <label for="nama">No Telp:</label>
    <input type="number" id="nama" name="nama"><br><br>

    <label for="pwd">Password:</label>
    <input type="password" id="pwd" name="pwd"><br><br>

    <input type="submit" value="Kirim">
  </form>
</body>
</html>
```

## Style.css

```
/* Atur ulang gaya dasar */
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f4f4f4; /* Warna latar belakang umum */
  color: #333;
}

/* Gaya Header (Bagian atas) */
header {
  background-color: #004d99; /* Biru tua */
  color: white;
  padding: 15px 30px;
  display: flex; /* Mengatur item secara horizontal */
  justify-content: space-between; /* Memisahkan logo dan navigasi */
  align-items: center;
}

.logo {
  font-size: 24px;
  font-weight: bold;
}

nav a {
  color: white;
  text-decoration: none;
  margin-left: 20px;
  padding: 5px 10px;
  border-radius: 4px;
  transition: background-color 0.3s;
}

nav a:hover {
  background-color: #003366;
}

/* Kontainer Utama untuk Layout 3 Kolom */
.container {
  display: flex; /* Mengaktifkan Flexbox untuk 3 kolom */
  max-width: 1200px;
  margin: 20px auto; /* Pusatkan konten */
  padding: 0 20px;
```

```
        gap: 20px; /* Jarak antar kolom */
    }

    /* Kolom Kiri: Artikel Populer (Sidebar) */
    .sidebar-left {
        flex: 1; /* Ambil 1 bagian proporsi lebar */
        padding: 20px;
        background-color: #fff;
        border-radius: 8px;
        box-shadow: 0 2px 4px rgba(0,0,0,0.1);
    }

    .sidebar-left h2 {
        color: #004d99;
        border-bottom: 2px solid #ccc;
        padding-bottom: 10px;
        margin-top: 0;
    }

    .sidebar-left ul {
        list-style: none;
        padding: 0;
    }

    .sidebar-left li {
        padding: 8px 0;
        border-bottom: 1px dotted #eee;
    }

    .sidebar-left li a {
        text-decoration: none;
        color: #333;
    }

    .sidebar-left li a:hover {
        color: #ff6600;
    }

    /* Kolom Tengah: Konten Utama */
    .main-content {
        flex: 3; /* Ambil 3 bagian proporsi lebar (Lebih lebar) */
        padding: 30px;
        background-color: #fff;
        border-radius: 8px;
```

```
        box-shadow: 0 2px 4px rgba(0,0,0,0.1);
    }

    .main-content h1 {
        color: #004d99;
        margin-top: 0;
        border-bottom: 3px solid #f0f0f0;
        padding-bottom: 10px;
    }

    .section-kami {
        display: flex;
        gap: 20px;
        margin-top: 20px;
    }

    .visi-kami, .layanan-kami {
        flex: 1;
        color: white;
        padding: 20px;
        border-radius: 8px;
    }

    .visi-kami {
        background-color: #c00; /* Merah */
    }

    .layanan-kami {
        background-color: #007bff; /* Biru terang */
    }

    .section-kami h3 {
        margin-top: 0;
        border-bottom: 1px solid rgba(255, 255, 255, 0.5);
        padding-bottom: 5px;
    }

    /* Kolom Kanan: Promosi Spesial (Sidebar) */
    .sidebar-right {
        flex: 1; /* Ambil 1 bagian proporsi lebar */
        padding: 20px;
        background-color: #fff;
        border-radius: 8px;
        box-shadow: 0 2px 4px rgba(0,0,0,0.1);
    }
```

```
}

.sidebar-right h2 {
    color: #ff6600; /* Oranye */
    border-bottom: 2px solid #ccc;
    padding-bottom: 10px;
    margin-top: 0;
}

.promo-box {
    color: white;
    padding: 15px;
    margin-bottom: 15px;
    border-radius: 6px;
    text-align: center;
}

.promo-box h4 {
    margin: 0;
    font-size: 18px;
}

.promo-box p {
    margin: 5px 0 0;
    font-size: 14px;
}

.blue {
    background-color: #007bff;
}

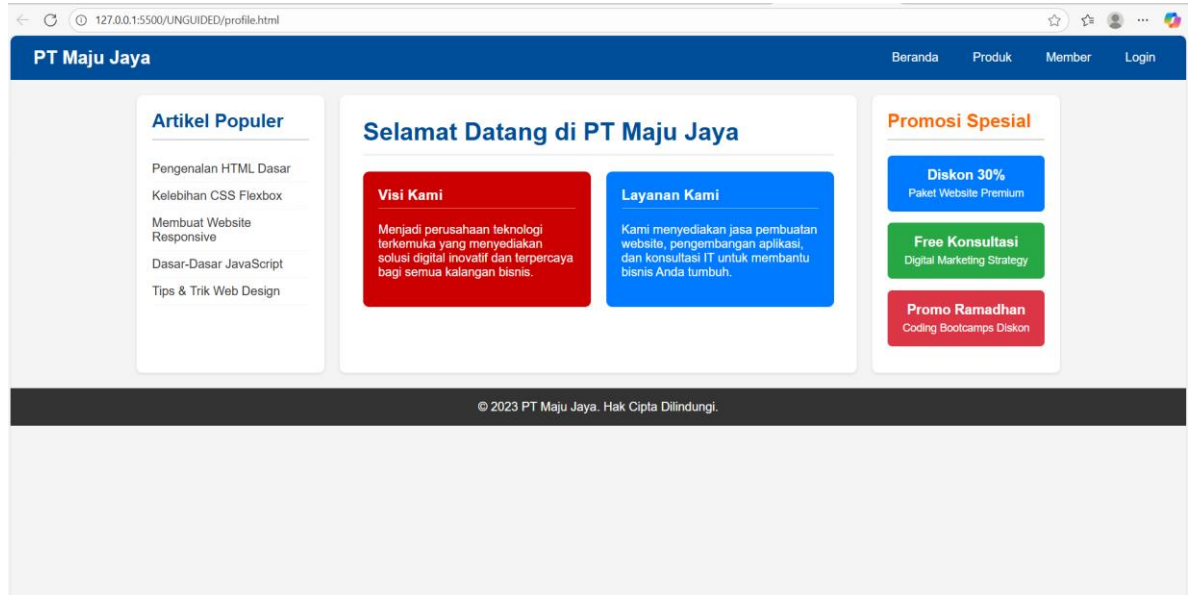
.green {
    background-color: #28a745;
}

.red {
    background-color: #dc3545;
}

/* Gaya Footer (Bagian bawah) */
footer {
    background-color: #333;
    color: white;
    text-align: center;
    padding: 15px 0;
    margin-top: 20px;
}
```

}

## Hasil Output



## 2.2 SOAL



- b) Buat layout html untuk halaman administrator seperti pada Gambar 2 yang berisi 4 menu (Data User, Kelola Produk, Edit Password, Logout)

Header	
Menu: Data User Kelola Produk Edit Password Logout	Konten Dinamis
Footer	

Gambar 2 Layout HTML halaman administrator

#### Source Code

```
<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Halaman Administrator</title>
  <link rel="stylesheet" href="admin_style.css">
</head>
<body>

  <header class="admin-header">
    <h1>Administrator Dashboard</h1>
  </header>

  <div class="main-wrapper">
    <aside class="admin-menu">
      <h2>Menu:</h2>
      <nav>
        <ul>
          <li><a href="#">Data User</a></li>
          <li><a href="#">Kelola Produk</a></li>
          <li><a href="#">Edit Password</a></li>
          <li><a href="#" class="logout">Logout</a></li>
```

```

        </ul>
    </nav>
</aside>

    <main class="admin-content">
        <h2>Konten Dinamis</h2>
        <p>Selamat datang di area administrator. Silakan pilih menu di
samping kiri untuk mengelola data.</p>
    </main>
</div>

<footer class="admin-footer">
    &copy; 2025 Admin Panel. Hak Cipta Dilindungi.
</footer>

</body>
</html>

```

#### admin\_style.css

```

/* Atur ulang gaya dasar */
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f0f0f5;
    color: #333;
    /* Aktifkan Grid pada body untuk mengatur 3 baris utama (Header,
Konten+Sidebar, Footer) */
    display: grid;
    /* Mendefinisikan 3 baris: Header, Main-Wrapper, Footer */
    grid-template-rows: auto 1fr auto;
    min-height: 100vh; /* Memastikan layout mengisi seluruh tinggi
viewport */
}

/* 1. HEADER */
.admin-header {
    grid-row: 1; /* Menempati baris pertama */
    background-color: #004d99;
    color: white;
    padding: 20px;
    text-align: center;

```

```
border-bottom: 5px solid #003366;
}

/* 2. MAIN WRAPPER (Area Konten dan Menu) */
.main-wrapper {
  grid-row: 2; /* Menempati baris kedua (area utama) */
  display: grid;
  /* Mendefinisikan 2 kolom: Menu (lebar tetap/kecil) dan Konten (sisa
lebar) */
  grid-template-columns: 250px 1fr;
  width: 90%;
  max-width: 1200px;
  margin: 20px auto;
  background-color: white;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

/* 2A. SIDEBAR/MENU */
.admin-menu {
  grid-column: 1; /* Menempati kolom pertama */
  padding: 20px;
  background-color: #e9ecef; /* Warna latar belakang menu */
  border-right: 1px solid #ccc;
}

.admin-menu h2 {
  color: #004d99;
  margin-top: 0;
  border-bottom: 2px solid #ccc;
  padding-bottom: 10px;
}

.admin-menu ul {
  list-style: none; /* Hilangkan bullet */
  padding: 0;
  margin-top: 10px;
}

.admin-menu li a {
  display: block;
  padding: 10px 15px;
  margin-bottom: 5px;
  text-decoration: none;
  color: #333;
```

```
border-radius: 4px;
transition: background-color 0.3s;
}

.admin-menu li a:hover {
background-color: #004d99;
color: white;
}

.admin-menu li a.logout {
color: #dc3545; /* Merah untuk Logout */
font-weight: bold;
}

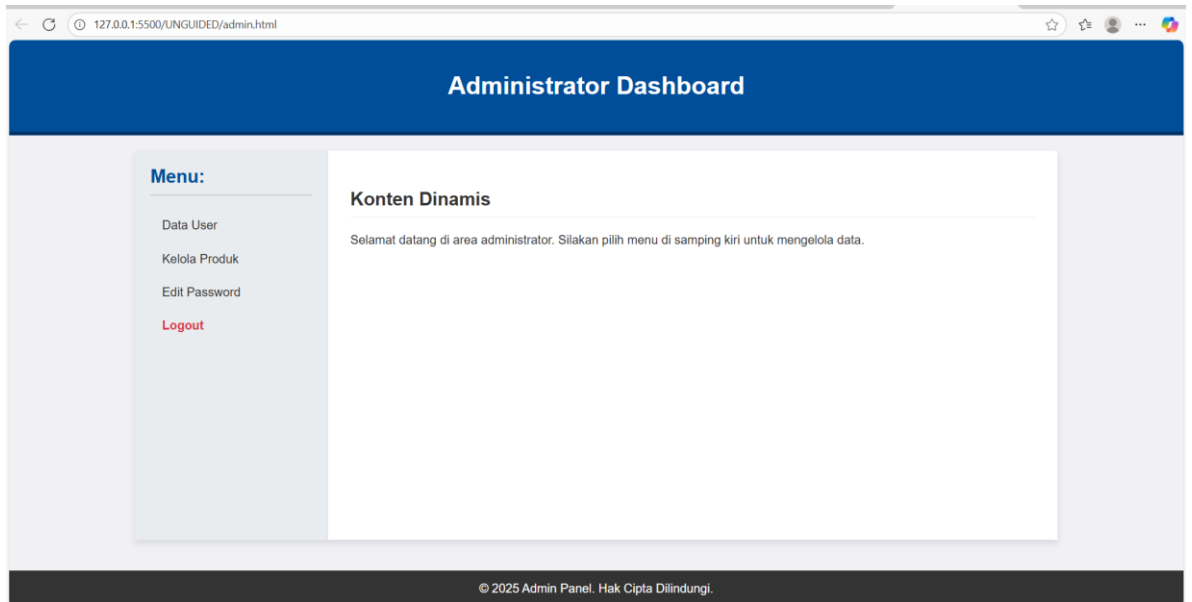
.admin-menu li a.logout:hover {
background-color: #dc3545;
color: white;
}

/* 2B. KONTEN DINAMIS */
.admin-content {
grid-column: 2; /* Menempati kolom kedua */
padding: 30px;
}

.admin-content h2 {
color: #333;
border-bottom: 1px solid #eee;
padding-bottom: 10px;
}

/* 3. FOOTER */
.admin-footer {
grid-row: 3; /* Menempati baris ketiga */
background-color: #333;
color: white;
text-align: center;
padding: 15px 0;
margin-top: 20px;
}
```

## Hasil Output



## BAB III

### KESIMPULAN & SARAN

#### 3.1 Kesimpulan

Praktikum ini menegaskan bahwa HTML menyediakan struktur dan konten halaman web, sementara CSS mengatur gaya serta tata letak. Kami berhasil mengimplementasikan layout multi-kolom menggunakan CSS Flexbox (untuk halaman profil) dan layout berbasis area menggunakan CSS Grid (untuk halaman administrator), membuktikan bahwa penguasaan kedua teknik CSS modern tersebut sangat penting untuk membangun user interface yang terstruktur dan profesional.

#### 3.2 Saran

Saran saya

- Responsifitas: Menerapkan media queries pada kedua layout agar tampilan dapat menyesuaikan diri dengan baik di berbagai ukuran layar (mobile, tablet, desktop).
- Interaktivitas: Mengintegrasikan JavaScript untuk membuat layout admin menjadi lebih interaktif, misalnya untuk menampilkan/menyembunyikan sidebar atau memvalidasi form tanpa memuat ulang halaman.
- Modularity CSS: Menggunakan penamaan kelas CSS yang lebih terstruktur (misalnya dengan metodologi BEM) untuk memastikan kode CSS mudah dipelihara pada proyek yang lebih besar.

