

MODUL 3

CSS (CASCADING STYLE SHEETS)

3.1 Tujuan Praktikum

Setelah mengikuti praktikum ini praktikan diharapkan dapat:

1. Praktikan dapat memahami pengertian, sejarah, dan konsep dasar CSS
2. Praktikan dapat mengetahui framework CSS
3. Praktikan dapat mengetahui dan memahami struktur CSS
4. Praktikan dapat mensimulasikan CSS untuk membangun atau mempercantik website

3.2 Alat dan Bahan

Alat dan bahan yang digunakan pada praktikum kali ini:

1. Laptop/PC
2. Visual Studio Code

3.3 Dasar Teori

3.3.1 Pengenalan CSS

CSS atau Cascading Style Sheets adalah bahasa yang digunakan untuk mendesain dan mengatur tampilan halaman web. Dengan CSS, pengembang dapat memisahkan konten (yang biasanya ditulis dalam HTML) dari aspek visualnya, sehingga memudahkan pengelolaan dan perubahan desain tanpa harus mengubah struktur konten. CSS mengumpulkan kode program untuk mempercantik atau mendesain tampilan HTML. Kita bisa mengubah desain dari teks, warna, gambar dan latar belakang dari semua kode tag HTML. CSS bekerja didalam tag-tag <html>. CSS biasanya digunakan untuk memformat tampilan web yang dibuat dengan bahasa HTML dan XHTML.

CSS bermula dari SGML (Standard Generalized Markup Language) pada tahun 1970-an dan mengalami perkembangan pesat. Ide dasar CSS dikemukakan oleh Hakon Wium Lie dalam proposalnya tentang Cascading HTML Style Sheets (CHSS) pada Oktober 1994 di konferensi W3C. Bersama Bert Bos, mereka mengembangkan standar CSS, yang resmi dipublikasikan pada tahun 1996 dengan dukungan Thomas Reardon dari Microsoft.

Nama "Cascading Style Sheets" mencerminkan cara deklarasi gaya dapat disusun secara berurutan, membentuk hubungan ayah-anak (parent-child). CSS direkomendasikan oleh World Wide Web Consortium atau W3C

sebagai teknologi internet, dan setelah distandarisasi browser seperti Internet Explorer dan Netscape mulai mendukungnya. CSS level 1, yang dirilis pada Desember 1996, bertujuan untuk mengurangi penggunaan tag baru dalam pengembangan web dan mendukung pengaturan elemen seperti font, warna, teks, latar belakang, dan atribut lainnya.

Tiga versi atau format utama Cascading Style Sheets (CSS) yang telah dibuat yaitu:

1. CSS level 1 muncul pada tahun 1996 dan menawarkan dasar untuk pengaturan gaya halaman web, seperti kontrol font, warna, dan layout dasar.
2. CSS level 2 muncul pada tahun 1998 dan menambahkan fitur seperti dukungan media dan positioning, yang memungkinkan desain yang lebih kompleks.
3. CSS level 3 dikembangkan pada awal 2000-an dan mencakup fitur animasi warna, 3D, dan kontemporer lain seperti border, radius, flexbox, layout, dan grid. Ini menjadikan CSS level 3 yang paling terbaru

3.3.2 Fungsi, Kelebihan, dan Kekurangan CSS

Fungsi CSS dapat diibaratkan sebagai hubungan antara HTML dan CSS yang mirip dengan tubuh dan pakaiannya. CSS berfungsi seperti pakaian yang dikenakan oleh tubuh. CSS memungkinkan pengelolaan yang lebih efisien, mendukung animasi dan transisi, meningkatkan interaktivitas dan pengalaman pengguna. Secara garis besar, tujuan CSS adalah untuk memberikan kontrol yang lebih besar terhadap tampilan dan estetika situs web sehingga menciptakan pengalaman pengguna yang lebih baik.

Adapun kelebihan dan kekurangan yang terdapat pada CSS diantaranya :

a. Kelebihan CSS

1. Desain responsif.
2. Penulisan kode dalam CSS bisa beberapa kali.
3. Menghemat waktu dalam proses pembuatan atau modifikasi halaman web.
4. Mudah dipelajari untuk pemula.
5. Pemisahan antara konten dan presentasi. HTML dan CSS disimpan dalam dua file terpisah, sehingga ukuran file HTML menjadi lebih kecil.

b. Kekurangan CSS

1. Kompatibilitas browser, tidak semua browser dapat menginterpretasikan perintah atau sintaks CSS dengan cara yang sama.
2. Tampilan yang tidak konsisten. Desain yang terlihat rapi di satu browser bisa tampak berantakan di browser lain.
3. Kesulitan pemeliharaan CSS. File CSS yang tidak terorganisir dalam proyek besar dapat sulit dikelola.
4. Penggunaan CSS Hack. Masalah kompatibilitas dapat diatasi dengan menggunakan script khusus atau CSS hack, meskipun ini dapat menambah kompleksitas.

3.3.3 Framework CSS

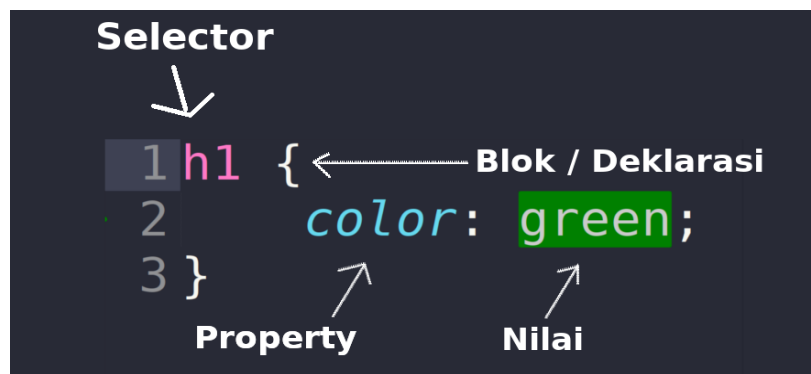
Framework CSS, memudahkan desain website atau aplikasi, khususnya layout dan tema. Framework CSS menyediakan elemen desain seperti sistem grid,

UI interaktif, tipografi, tombol, dan ikon. Karena itu, Anda tidak perlu membuat desain dari nol. Anda dapat mempercepat proses pengembangan front-end dengan menyalin dan menyesuaikan elemen yang ada untuk mencapai desain yang diinginkan. Contoh Framework CSS:

1. Bootstrap
2. Foundation
3. Bulma
4. Semantic UI
5. UIkit
6. Materialize CSS
7. Milligram
8. PureCSS
9. Skeleton
10. Tailwind

3.3.4 Struktur CSS

Berikut adalah struktur CSS yang perlu diketahui sebelum kita mulai untuk mempercantik tampilan website kita:



Gambar 3.3.4 Struktur CSS

3.3.5 Bootstrap

Bootstrap adalah framework CSS open source yang membantu pengembang web membuat tata letak dan komponen web dengan cepat dengan menyediakan template berbasis HTML dan CSS untuk elemen seperti tipografi, formulir, tombol, dan navigasi. Selain itu, Bootstrap menyediakan ekstensi yang dapat dipilih yang berbasis JavaScript untuk menambah interaktivitas.



Gambar 3.3.5 Bootstrap

LANGKAH – LANGKAH PRAKTIKUM:

1. Selektor

Selektor adalah kata kunci untuk memilih elemen pada HTML yang akan kita atur atau kita sesuaikan. Tag <h1> adalah selektor.

```
1  h1 {  
2      color : green;  
3  }
```

Elemen <h1> akan memberikan warna teks green pada HTML.

2. Blok Deklarasi

Blok deklarasi adalah tempat kita menuliskan atribut – atribut CSS yang akan diberikan kepada selektor.

```
1  p {  
2      font-size : 50px;  
3  }
```

Tag <p> mengatur ukuran font text sebesar 50 pixel. Blok deklarasi dimulai atau dibuka dengan tanda kurung { lalu ditutup dengan }.

- Properti dan Nilainya

Properti adalah sifat atau aturan yang diberikan kepada elemen tertentu. Setiap properti harus diakhiri dengan ";" Titik koma mungkin tidak digunakan jika hanya ada satu properti. Properti harus ditulis di bagian deklarasi.

```
1  p {  
2      font-size : 50px;  
3  }
```

Properti pada gambar di atas adalah font-size dan nilai atau ukurannya adalah 50px.

3. Penulisan CSS

Untuk penulisan Inline Style Sheets dan Internal Style Sheets dilakukan dalam file HTML nya langsung tanpa membuat file CSS terpisah.

- Inline Style Sheets

Cara penulisan CSS menggunakan Inline Style Sheets adalah dengan menambahkan atribut "style" pada tag HTML yang akan ditambahkan pemformatan CSS. Misal, kita akan membuat tampilan elemen h1 menjadi

berwarna blueviolet, maka penulisan elemen h1-nya adalah sebagai berikut:

```
<h1 style="color: blueviolet">Hello World!</h1>
```

- Internal Style Sheets

Cara penulisan CSS menggunakan Internal Style Sheets adalah dengan cara menuliskan baris kode CSS pada elemen head menggunakan tag <style> seperti pada gambar dibawah ini:

```
1 <html>
2 <head>
3   <title>Hello World</title>
4   <style>
5     body {background: lavender;color: greenyellow;}
6     h1 {color: gold;}
7   </style>
8 </head>
9
10 <body>
11   <h1>Hello World!</h1>
12 </body>
13 </html>
```

- External Style Sheets

Untuk menggunakan External Style Sheets untuk menulis CSS, Anda harus membuat file CSS terpisah dan kemudian membuat tautan dengan tag pada elemen head dokumen html yang akan menerapkan format CSS. <link> adalah pemformatan CSS. Sebagai ilustrasi, baris kode CSS ditampilkan seperti berikut:

```
1 <html>
2   <head>
3     <title>WORLD</title>
4     <link rel="stylesheet" href="style.css" />
5   </head>
6   <body>
7     <div>
8       <h1 class="h1-footer">Telnet Laboratory</h1>
9     </div>
10  </body>
11 </html>
```

```

.h1-footer{
  font-size: 40px;
  color: white;
}

.class{
  height: 20;
  width: 20;
}

```

4. Aturan Penulisan CSS

Didalam penulisan CSS terdapat dua komponen utama, yaitu:

a. Deklarator

Terdiri dari Property dan Value, untuk membuat tampilan pada selector sesuai dengan perintah yang ada pada deklarator sehingga hasil deklarator dapat ditampilkan di browser.

b. Selector

Tag HTML, seperti body, table, h1, dll., yang akan disesuaikan dengan style CSS. Selector dasar ada lima jenis, yaitu:

- Universal Selector

Universal selector hanya ada 1 saja di dalam CSS, yaitu tanda bintang (*). Selector ini bertujuan untuk diterapkan pada semua tag yang ada.

```

1  * {
2    margin: 0;
3    padding: 0;
4    background-color: skyblue;
5    color: rebeccapurple;
6  }

```

- Element Type Selector

Selector tipe element, juga dikenal sebagai selector tag, adalah selector yang nilainya adalah tag HTML itu sendiri. Setiap tag HTML dapat digunakan sebagai selector, dan selector ini akan menangkap semua tag tersebut.

```
<h1>Hello World</h1>
<p>Hello World Hello World Hello World.
    Hello World Hello World Hello World Hello World.
    Hello World Hello World Hello World Hello World.
</p>
```

```
1  h1{
2      font-family: monospace;
3      text-decoration: underline;
4      font-weight: bold;
5  }
6  p{
7      font-size: 160px;
8  }
```

- Class Selector

Merupakan salah satu pilihan yang paling banyak digunakan dan digunakan. Untuk menggunakan class selector, kita harus memiliki tag HTML yang memiliki atribut class.

```
<h1>class="judul">Hello World</h1>
<p class="isi">Hello World Hello World Hello World.
    Hello World Hello World Hello World Hello World.
    Hello World Hello World Hello World Hello World.
</p>
```

Perhatikan bahwa untuk tag di atas, kita menambahkan atribut kelas dengan nilainya adalah nama kelas. Satu nama kelas dapat memiliki lebih dari satu tag, dan satu class dapat memiliki lebih dari satu tag.

Contoh, dalam baris terakhir pada contoh diatas, tag <h1> memiliki atribut class="judul".

```
4  .judul {
5      background: blue;
6      color: brown;
7  }
8
9  .isi{
10     background: red;
11     color: black;
12 }
```

Kita menggunakan titik sebelum nama class di CSS untuk menggunakan class selector. Jika class memiliki nilai "judul", warna teks akan berwarna biru. dan teks di sekitar class "isi" akan berwarna merah.

- ID Selector

Selector class dan selector ID keduanya sangat populer dan digunakan secara bersamaan, tetapi class selector menggunakan atribut class untuk tag HTML, sedangkan ID Selector menggunakan atribut id.

```
<h1 id="judul_id">class="judul">Hello World</h1>
<p id="isi_id">Hello World Hello World Hello World.
Hello World Hello World Hello World Hello World.
Hello World Hello World Hello World Hello World.
</p>
```

Atribut id berfungsi sebagai selector CSS dan kode unik untuk setiap tag, terutama untuk kode JavaScript, jadi id yang digunakan tidak boleh sama.

Dengan kata lain, id hanya bisa digunakan satu kali dalam sebuah halaman web dan tidak boleh sama.

```
1  #judul_id{
2      font-family: sans-serif;
3      font-size: 16px;
4      color: lightseagreen;
5  }
6
7  #isi_id{
8      font-size: 20px;
9      font-weight: bold;
10     color: salmon;
11 }
```

- Attribute Selector

Dibandingkan dengan selector sebelumnya, selector ini sedikit lebih canggih. Fungsinya digunakan untuk mencari seluruh tag yang memiliki atribut yang dituliskan.

HTML:

```
<input type="text" placeholder="ketikkan sesuatu...." />
```

CSS:

```
1  input[type="text"]{
2      background: none;
3      color: cyan;
4      padding: 10px;
5      border: 1px solid cyan;
6  }
```