# DAY 6 : Pengenalan CSS

Pada pembelajaran sebelumnya kita sudah sempat menyinggung peran CSS pada website. Website nampak begitu membosankan dan mengerikan tanpa adanya CSS. Cascading Style Sheet atau biasa disingkat CSS merupakan W3C standar yang digunakan untuk mengatur visualisasi berkas yang ditulis pada HTML. Pada pembelajaran kali ini kita akan mengenal dasar-dasar dari CSS. Tepatnya mulai dari pembuatan berkas CSS, pengenalan struktur sintaksnya, hingga menerapkan dasar styling seperti memberikan warna pada sebuah teks.

CSS syntax berbeda dengan HTML, begitu pula dengan JavaScript. CSS bukanlah sebuah bahasa pemrograman karena di dalamnya tidak terdapat logika, tidak dapat membuat sebuah variabel, tidak adanya proses iterasi, dsb. CSS hanya sebuah declarative language yang digunakan untuk mendeklarasikan suatu nilai yang nantinya digunakan untuk mengatur seperti apa sebuah elemen HTML ditampilkan pada browser.

## 1. Alasan Menggunakan CSS

Kita tentu telah mengetahui bahwa CSS digunakan untuk mempercantik halaman dari sebuah website. Berikut beberapa alasan lain mengapa kita menggunakan CSS:

**a. Mempercepat Proses Desain**

Ketika menggunakan CSS, Anda cukup mengetikkan satu kali fungsi CSS kemudian menggunakannya di berbagai halaman HTML. Fungsi CSS yang Anda buat dalam satu file dapat Anda panggil ke berbagai halaman web tanpa harus menyalin baris kode fungsi berkali-kali.

**b. Proses Pemeliharaan Mudah**

CSS memudahkan Anda untuk mengubah tampilan di berbagai halaman. Hanya dengan mengubah fungsi style di file CSS maka seluruh tampilan yang menggunakan fungsi tersebut akan berubah secara otomatis.

**c. Didukung banyak browser.**

Seluruh browser saat ini minimal sudah mendukung CSS versi 2. Untuk browser yang populer seperti Chrome dan Firefox sudah mendukung CSS versi 3.

**d. Style Lebih Beragam Dibanding HTML**

CSS mempunyai atribut lebih beragam dibandingkan dengan HTML. Apa keuntungannya? Anda mempunyai lebih banyak pilihan tampilan halaman website.

## 2. Cara Kerja CSS

Untuk dapat menerapkan CSS, ada beberapa langkah yang perlu kita lakukan, diantaranya:

1. Dimulai dari sebuah dokumen yang telah ditandai dengan tag elemen CSS.

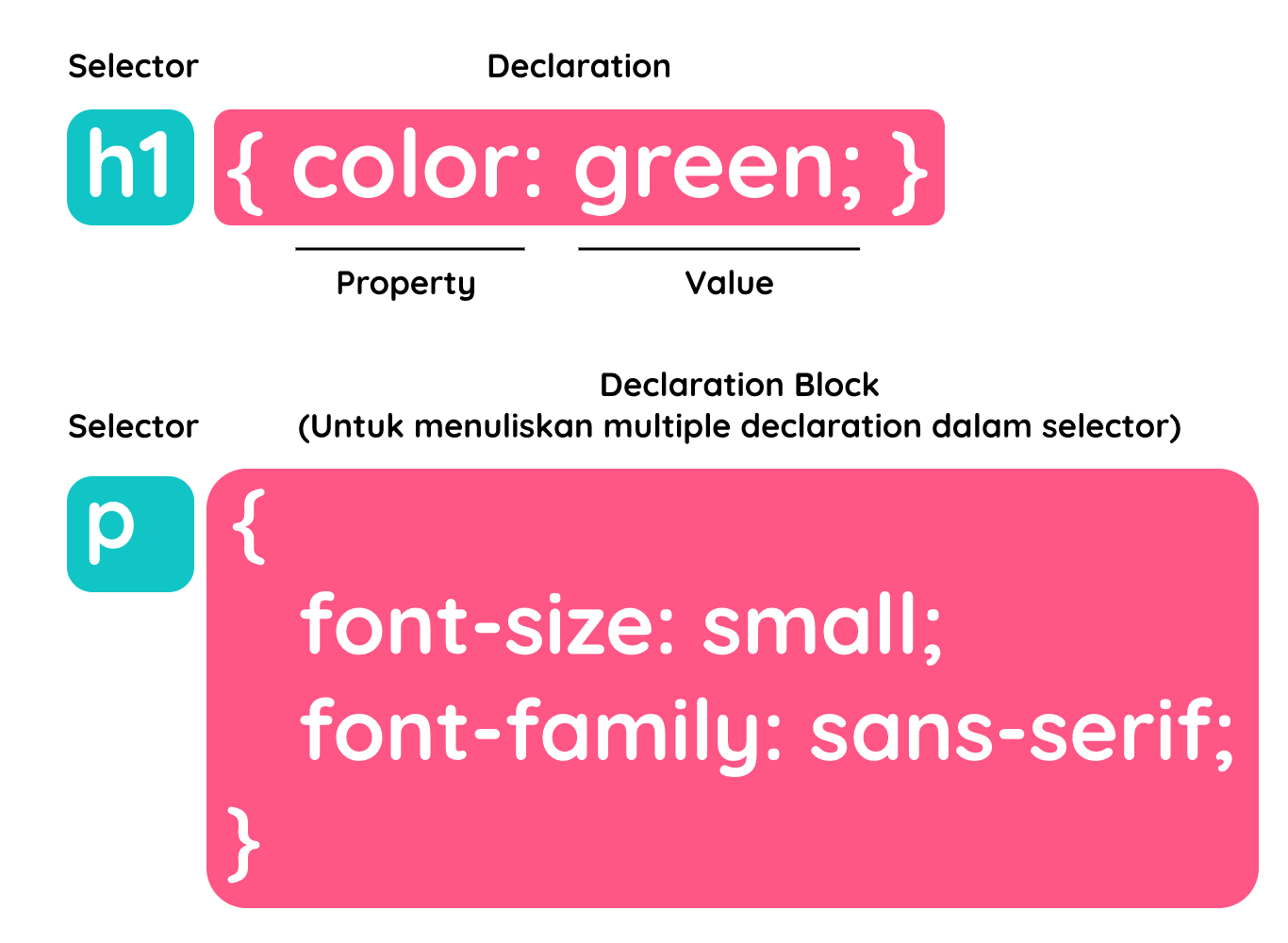
2. Menuliskan aturan styling untuk menentukan bagaimana elemen HTML akan ditampilkan.

3. Melampirkan aturan styling yang sudah dibuat pada dokumen HTML. Ketika browser memuat dokumen, tampilan elemen yang ditampilkan akan menyesuaikan dengan aturan styling yang sudah ditetapkan.

## 3. Menentukan Aturan Styling

Sebuah style sheet dibuat terdiri dari satu atau lebih aturan styling (biasa disebut dengan rules atau rule-sets) yang mendeskripsikan bagaimana sebuah elemen atau sebuah kelompok elemen ditampilkan dalam jendela browser.

Dalam penggunaan CSS, terdapat dua bagian dalam sebuah rule. Yang pertama adalah identitas elemen atau elemen yang akan menerapkan rule (singkatnya kita akan sebut selector) dan yang kedua adalah deklarasi atau instruksi yang akan diterapkan pada sebuah selector. Perhatikan ilustrasi berikut ini:



### 3.1 Selector

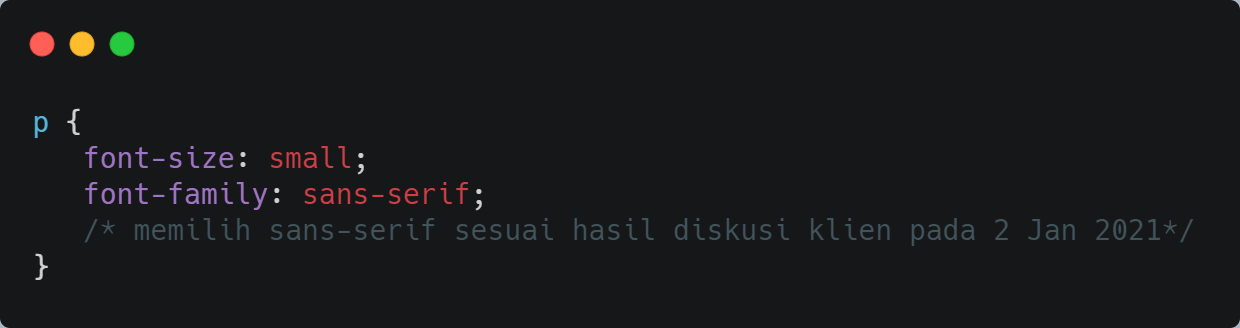
Pada contoh di atas, h1 dan p digunakan sebagai selector. Selector ini dipanggil melalui tipe elemennya, dan ini merupakan teknik dasar dari pemanggilan selector. Properti dan nilainya yang terdapat pada declaration/declaration block akan diterapkan pada seluruh elemen <h1> dan <p> yang ada pada dokumen HTML.

### 3.2 Declarations

Bagian deklarasi terdiri dari pasangan properti dengan nilainya. Kita bisa menetapkan lebih dari satu deklarasi pada satu rule, contohnya seperti pada selector p di atas. Kita menetapkan lebih dari satu deklarasi pada declaration block. Setiap deklarasinya harus diakhiri dengan semicolon (;) sebagai tanda diakhirinya sebuah deklarasi.

### 3.3 Comments

CSS comments digunakan untuk menjelaskan kode, dan dapat membantu ketika Anda mengedit kode sumber di lain waktu. Apa yang tertulis akan diabaikan oleh browser dan tidak ditampilkan di browser, sehingga dapat menjadi opsi baik untuk memberi catatan atau informasi dokumentasi pada kode. Cara penulisan dimulai dengan /\* dan diakhiri \*/ seperti contoh berikut:



## 4. Melampirkan CSS pada HTML

Setelah kita menuliskan rules, maka tahapan selanjutnya adalah melampirkan atau menerapkan aturan tersebut pada berkas HTML. Sebenarnya, terdapat tiga cara untuk menerapkan styling pada elemen HTML:

### 4.1 External Style Sheet

External Style Sheet merupakan berkas terpisah yang di dalamnya hanya terdapat sebuah rules. Berkas ini harus berekstensi .css, dan berkas ini nantinya dihubungkan pada dokumen HTML. Cara ini merupakan yang paling powerful dalam menerapkan styling. Karena dengan cara ini, satu berkas styling (.css) dapat digunakan oleh banyak berkas HTML.

Untuk menyambungkan berkas .css dengan dokumen HTML, kita dapat menggunakan elemen <link> pada <head> berkas HTML. Contohnya:



Pada elemen <link> tersebut, kita tetapkan berkas CSS yang digunakan dengan menggunakan atribut href dan beri nilai *“stylesheet”* pada atribut rel sebagai relationship (hubungan) antara berkas style.css dengan dokumen HTML.

Pada contoh di atas kita tahu bahwa berkas css yang digunakan merupakan berkas lokal (berkas yang berada pada komputer/server kita sendiri). Nilai atribut href juga dapat berupa berkas .css yang tersedia melalui sebuah URL. Contohnya, banyak pengembang menggunakan bootstraps.min.css untuk membantu penyusunan layout website-nya. Kita bisa menggunakannya pada berkas HTML dengan langsung menuliskan URL untuk berkas tersebut. Contoh penggunaan url pada stribut href adalah sebagai berikut:



### 4.2 Embedded Style Sheet

Embedded Style Sheet merupakan kumpulan rules yang dituliskan dalam berkas HTML dengan menggunakan elemen <style>. Dengan begitu rules yang dituliskan hanya dapat dicakup oleh satu berkas HTML. Penulisan rules harus dituliskan dalam elemen <style> dan ditempatkan di dalam <head> dari berkas HTML. Perhatikan contoh penerapan berikut ini:



### 4.3 Inline Style

Inline Style merupakan styling yang diterapkan pada elemen HTML dengan menggunakan atribut style. Contohnya seperti berikut:

A picture containing text

Description automatically generated

Untuk menambahkan styling properties lainnya (multiple properties), kita tuliskan dengan menggunakan semicolon (;) sebagai pemisah antar styling properties-nya. Contohnya sebagai berikut:



Inline styles hanya diterapkan pada elemen di mana atribut style diterapkan. Teknik ini seharusnya dihindari terkecuali benar-benar diperlukan untuk menggantikan sebuah styling yang ditetapkan pada Embedded Style Sheet atau External Style Sheet.