**CSS (*Cascading Style Sheet*)**

**Pengenalan CSS**

Cascading Style Sheet (CSS) adalah suatu Bahasa yang bekerja sama dengan dokumen HTML untuk mendefinisikan cara bagaimana suatu isi halaman web ditampilkan atau dipresentasikan. Presentasi ini meliputi style atau gaya teks, link dan tata letak (layout) halaman.

**Aturan Penulisan CSS**

Kode CSS tersusun atas dua bagian yaitu selector dan deklarasi. Selector adalah tag HTML yang akan diberi atau dikenal CSS, sedangkan deklarasi adalah property dan nilai yang akan ditentukan untuk tag yang bersangkutan. Berikut penulisan dari CSS yaitu :

Selector

Body {

Background : black;

Color : white;

}

Value

property

Pada penulisan script CSS diatas, dapat di lihat terdapat dua bagian yaitu body disebut sebagai selector dan background : black, color : white; disebut sebagai deklarasi. Background dan color merupakan property dari tag <body> yang akan di set nilainya, sedangkan black dan white merupakan nilai yang digunakan atau akan diisikan ke dalam atribut background dan color.

**Bagian dari CSS :**

1. Selector adalah elemen / tag HTML yang ingin diberi style. Selector dapat langsung di tuliskan nama tag yang ingin diberi style tanpa perlu menambahkan tanda < >. Pada kode CSS di atas, dapat memberikan style pada tag body yang terdapat di dalam file HTML. Jika tag HTML ingin diberi style memiliki ID, maka dapat dituliskan nama ID tersebut dengan diawali tanda #. Seperti contoh di bawah ini :

#header {

Background : black;

Color : white;

}

Setiap tag yang ada pada HTML bisa dijadikan selector dan terdapat dua jenis selector yaitu

1. **CLASS**

Class digunakan untuk membuat berbagai macam variasi bagi satu tag HTML.

dan jika tag yang diberi style memiliki class, dapat di tulis dengan menambahkan tanda ( . ) dan diikuti dengan nama class. Format penulisan class :

.artikel {

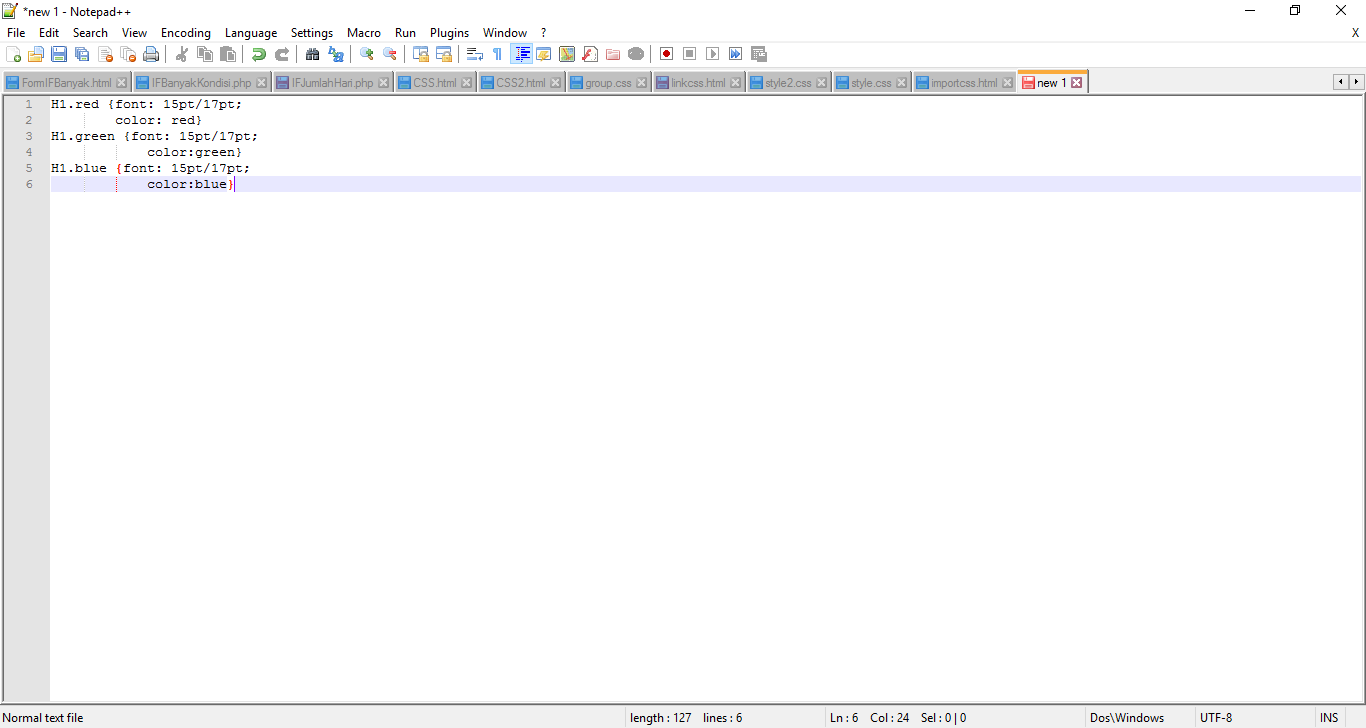
Background : black;

Color : white;

}

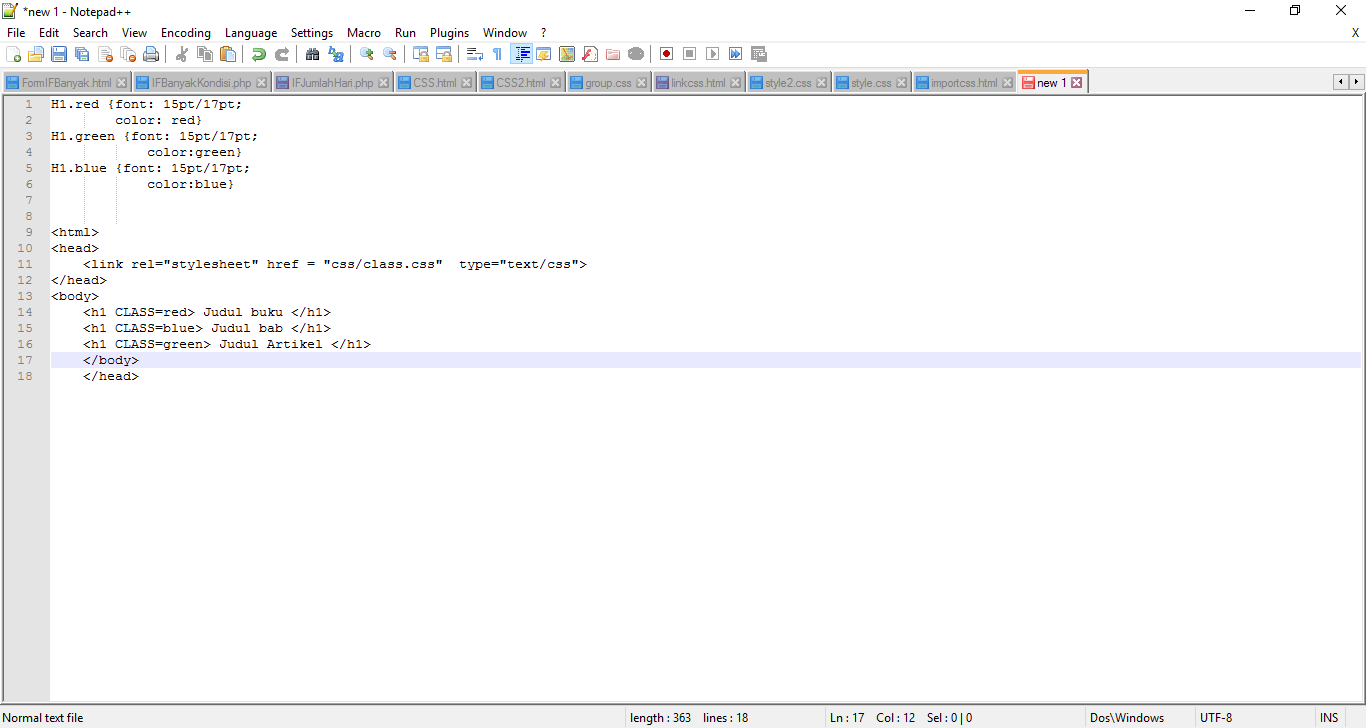
Contoh1 penerapan selector class:

File .CSS



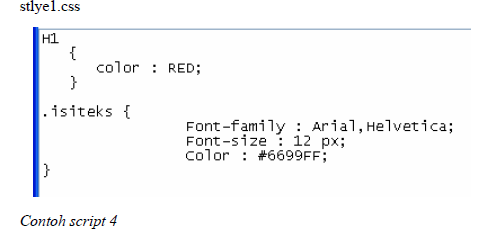
CLASS

File .HTML

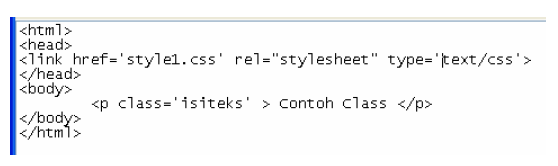


Contoh 2 penerapan selector class :

File .CSS



File .HTML



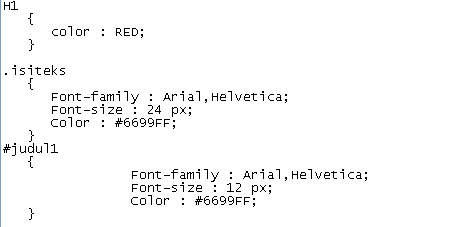
Tampilan pada Halaman Web :



1. **ID**

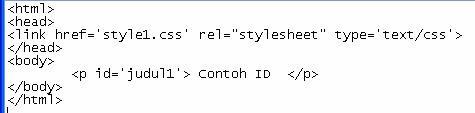
**ID** digunakan untuk memformat sebuah elemen yang untuk dan muncul satu kali dalam satu halaman web

Diawali dengan tanda #, dapat digunakan untuk mendefinisikan header,content, dan footer dalam desain web krn didefiniskan denga ID berbeda. Penulisan selector ID .



Contoh penerapan Selector ID :

File .HTML



Tampilan di Halaman Web :



1. Property & Value

Untuk membuat file CSS, hal yang diketahui adalah property dan nilai yang akan dijadikan sebagai deklarasi. Untuk memudahkan dalam pembuatan file CSS ada beberapa property dan value yang digunakan dan di klasifikasikan berdasarkan fungsinya yaitu diantaranya :

1. Teks dan font

Property dan value berikut apat diterapkan untuk penentuan jenis, ukuran, dan warna huruf maupun penempatan posisi dari suatu teks.

**Tabel Daftar Teks dan Font**

|  |  |  |
| --- | --- | --- |
| **Properti** | **Fungsi** | **Value** |
| Font – Family | Menentukan jenis huruf yang akan digunakan | Nilai yang digunakan pada font – family terdiri dari : ***arial, comic sans, times new roman, dll*** |
| Fonr – size | Menentukan ukuran huruf pada teks | Ukuran bisa dalam satuan point (pt), pixel (px), sentimeter (cm), maupun nilai absolut yaitu : ***medium, small, large*** |
| Font – style | Menentukan style pada font yang digunakan | Nilai yang digunakan adalah normal, italic, inherit |
| Font - weight | Menentukan ketebalan pada font yang digunakan | Nilai yang digunakan adalah ***light, medium, normal, bold, dll*** |
| Text - decoration | Menentukan format elemen teks | Nilai yang digunakan adalah ***underline, overline, line- through, blink, inherit*** |
| Text - transform | Menentukan huruf besar atau kecil pada elemen teks | Nilai yang digunakan ***capitalize, uppercase, lowercase*** |
| Text – align | Menentukan alignment elemen teks | Nilai yang digunakan adalah ***left, right, center, justify, inherit*** |
| Text - indent | Menentukan spasi sebelum elemen teks | Nilai yang digunakan adalah Angka dalam satuan point (pt), pixel (px), inche (in), sentimeter (cm) |

1. Warna dan background

Property dan value berikut digunakan dalam penentuan style untuk latar (*background*) dari teks, halaman, tabel, atau elemen lain.

**Tabel Daftar Properti dan Nilai Warna dan *Background***

|  |  |  |
| --- | --- | --- |
| **Properti** | **Fungsi** | **Value** |
| Background – color | Menentukan warna background dari suatu elemen | Nilai yang digunakan adalah angka dalam heksadesimal atau teks yang menunjukkan warna, seperti : ***black, navy, pink, red, yellow, dll*** |
| Background – image | Menentukan gambar yang akan digunakan sebagai background | Nilai yang digunakan adalah URL dari gambar yang akan digunakan, seperti contoh : url (“lokasi\_file\_gambar”) |
| Background - repeat | Menentukan tipe pengulangan yang akan diterapkan untuk gambar background | Nilai yang digunakan adalah ***no-repeat, repeat, repeat-x, repeat-y*** |
| Color | Menentukan warna teks | Nilai yang digunakan adalah angka dalam heksadesimal atau teks yang menunjukkan warna, seperti : ***black, navy, pink, red, yellow, dll*** |

1. Bingkai (*border*) dan tabel

Berikut ini adalah property dan value yang digunakan untuk menentukan style dari border pada halaman, elemen teks, tabel dan gambar

**Tabel Daftar Properti dan Nilai bingkai adan tabel**

|  |  |  |
| --- | --- | --- |
| **Properti** | **Fungsi** | **Value** |
| Border – color | Menentukan warna border | Nilai yang digunakan adalah angka dalam heksadesimal atau teks yang menunjukkan warna, seperti : ***black, navy, pink, red, yellow, dll*** |
| Border - style | Menentukan gaya garis dari border | Nilai yang digunakan adalah ***dotted, dashed, solid, double, groove, ridge, inset, outset, inherit, none*** |
| Border - width | Menentukan lebar border | Nilai yang digunakan adalah angka dalam satuan point (pt), pixel (px), inche (in), sentimeter (cm), maupun salah satu dari nilai berikut : ***thin, medium, thick, inherit*** |
| Caption – side | Menentukan posisi judul dari suatu tabel | Nilai yang digunakan adalah ***top, dan bottom*** |
| float | Menentukan spasi sebelumelemen teks, relatif ke elemen yang berada di sekitarnya | Nilai yang digunakan adalah ***left, right, none, inherit*** |

1. Daftar (*list*)

Property dan value berikut dapat diterapkan sebagai style untuk elemen daftar (list), baik yang terurut (*ordered list* ) maupun tidak terurut (*unordered list*).

**Tabel Daftar Properti dan nilai daftar (*list*)**

|  |  |  |
| --- | --- | --- |
| **Properti** | **Fungsi** | **Value** |
| List-style-image | Menentukan gambar yang akan digunakan untuk mengganti bullet dari semua item yang terdapat di dalam daftar | nilai yang digunakan adalah alamat dari gambar, **URL (“lokasi\_file\_gambar”), none, inherit** |
| List – style – type | Menentukan tipe bullet pada daftar tidak terurut (*undordered list*) atau tipe penomoran pada daftar yang terurut (*ordered list*) | Nilai yang digunakan adalah ***bullet dapat diganti dengan tipe disc, circle, square.***  Penomoran dapat diganti dengan ***angka lower-roman, upper-roman, lower-latin, upper-latin, atau karakter alfabet*** |

**Menggabungkan Kode CSS dengan dokumen HTML**

Proses penggabungan kode CSS dengan dokumen HTML dapat dilakukan melalui 3 (tiga) cara, yaitu :

1. **Embedded styles**

merupakan menulis kode CSS pada bagian header dokumen HTML. Pada cara ini, semua kode CSS ditulis di dalam tag <style> dan di akhir dengan tag </style> dan ditempatkan pada bagian header dari dokumen HTML. Kode CSS yang ditulis dengan cara seperti hanya berlaku untuk satu dokumen saja.

**Manfaat Embedded Style :**

1. Perubahan hanya terjadi pada 1 halaman
2. Class dan ID bisa digunakan oleh embedded style
3. Tidak perlu mengupload beberapa file karena HTML dan CSS bisa digunakn di file yang sama

**Kekurangan menggunakan Embedded Style :**

1. Meningkatkan waktu akses website
2. Perubahan hanya terjadi pada 1 halaman dan tidak efisien bila ingin menggunakan CSS yang sama pada beberapa file HTML

<html>

<head>

<title> penggunaan embedded CSS</title>

<style type = “text/css”>

/\* style dengan tag <body>\*/

Body {

Background : black;

Color : white;

}

</style>

</head>

<body>

Demo embedded Style

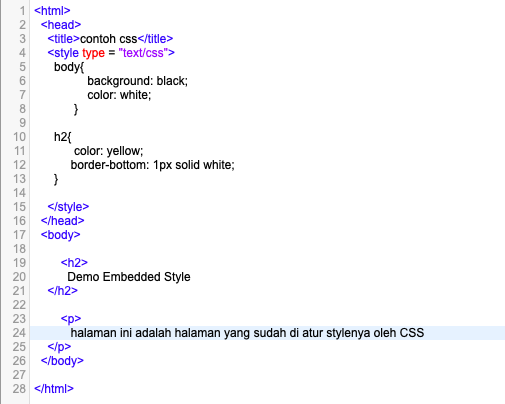
<p> halaman ini merupakan halaman yang sudah diatur style-nya oleh kode CSS </p>

</body>

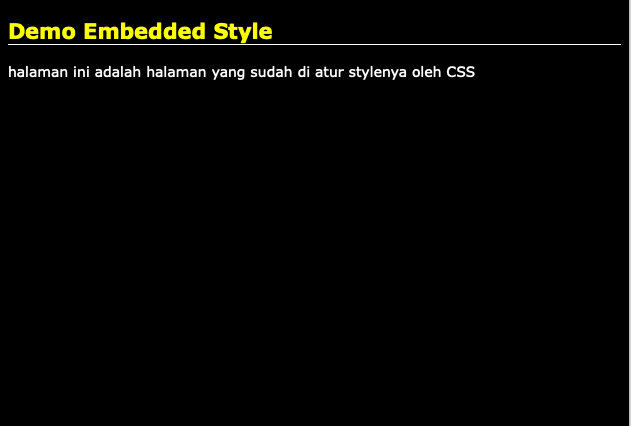
</html>

**Catatan** : tanda /\* dan \*/ pada CSS digunakan untuk membuat komentar

**Contoh penerapan embedded style :**



Tampilan pada halaman web :



1. **Linked styles**

Merupakan menulis kode CSS pada file terpisah dan dipanggil melalui tag <link> dari dokumen HTML. Pada cara ini kode CSS ditulis secara terpisah pada file tersendiri (dengan ekstensi .CSS). file tersebut selanjutnya dapat dikaitkan atau dihubungkan dengan dokumen HTML melalu tag <link>. Sehingga kode CSS dengan file .CSS dapat digunakan oleh beberapa dokumen HTML yang ingin menerapkan style. Dengan cara ini kode CSS akan mudah untuk dipelihara dan dimodifikasi.

Tag <link> yang digunakan untuk memanggil file .CSS diletakkan pada bagian header dokumen HTML, berikut pentuk penggunaan dari linked style.

<link rel = “stylesheet” href = “nama\_file.css” type = “type/css”>

Contoh :

**File style .css**

Body {

Background : black;

Color : white;

}

**File sampel .html**

<html>

<head>

<title> penggunaan linked style </title>

<link rel = “stylesheet” href = “style.css” type = “text/css”>

</head>

<body>

Demo linked Style

</body>

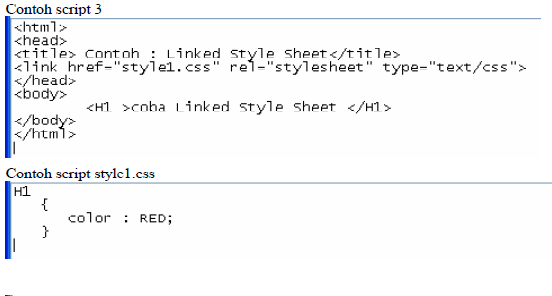
</html>

Keuntungan dari penggunaan linked style, diantaranya :

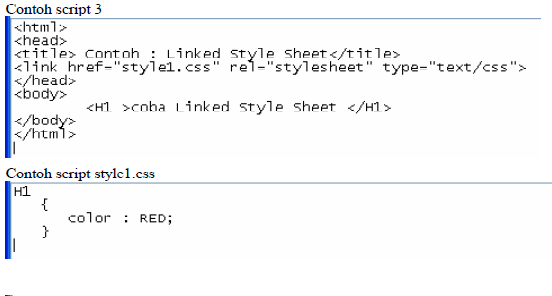
1. Dengan menggunakan cara ini kode pada CSS maupun HTML terlihat lebih rapi dan mudah untuk dipahami dan dibaca.
2. Lebih mudah dikembangkan atau dimodifikasi
3. Kode CSS dapat digunakan oleh dokumen – dokumen HTML lainnya yang ingin menerapkan style tersebut
4. Perubahan pada file .css akan berpengaruh terhadap semua dokumen HTML yang menerapkannya, dengan begitu perubahan style hanya dilakukan sekali pada file .css saja.

Contoh penerapan linked style :

File .CSS



File .HTML



Tampilan pada halaman web :



1. **Inline styles**

Merupakan menulis kode CSS langsung pada tag yang akan diubah atau diatur style-nya. Cara ini merupakan cara yang sangat rumit dam susah untuk dibaca, karena kode CSS ditulis atau dimasukkan ke dalam atribut style pada tag yang akan digunakan. Sebagai contoh, jika anda memiliki beberapa tag <h2> dalam dokumen HTML yang anda buat, maka anda harus menuliskan style pada setiap tag yang ada. Jika anda ingin memodifikasi style tersebut, maka anda harus mengubah setiap tag yang ada.

**Manfaat Inline Style :**

1. Berguna jika ingin menguji dan melihat perubahan
2. Berguna untuk perbaikan cepat
3. Permintaan HTTP yang lebih Kecil

**Kekurangan Inline Style :**

Inline style harus diterapkan pada setiap elemen

Format penulisan inline style :

<html>

<head>

<title> penggunaan inline style </title>

</head>

<body>

<h2>Demo inline style</h2>

<p style = “color: red; text-decoration: underline;”>

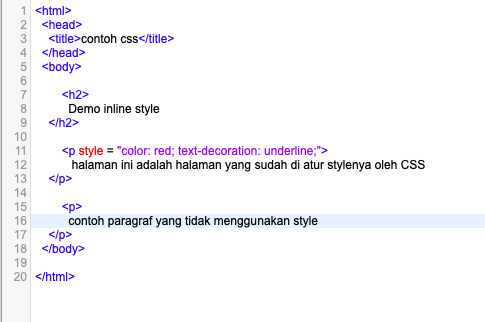
Contoh paragraph yang menggunakan inline style. </p>

<p> contoh paragraph yang tidak menggunakan inline style</p>

</body>

</html>

Contoh penerapan inline style :



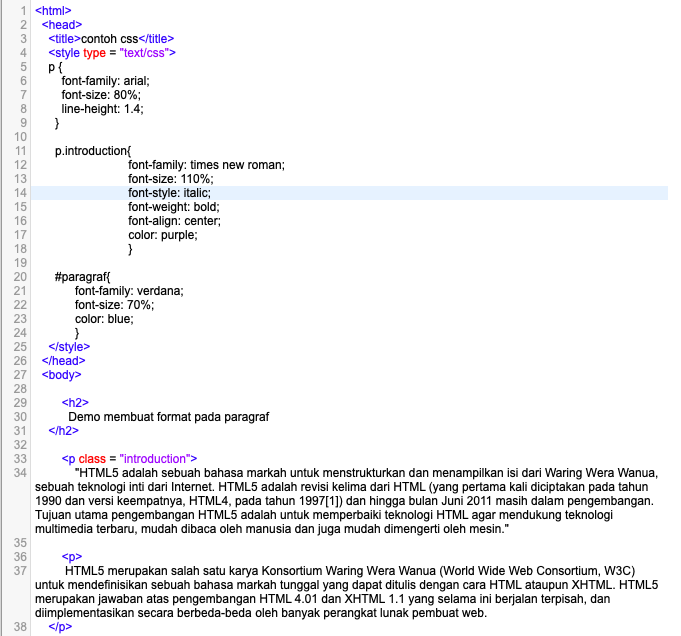
Tampilan pada halaman web :

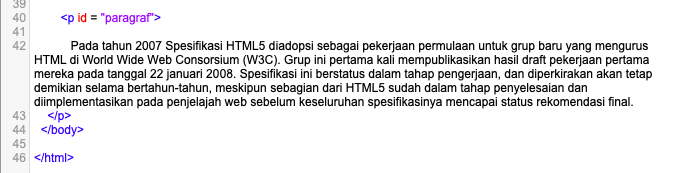


**Membuat format pada paragraph**

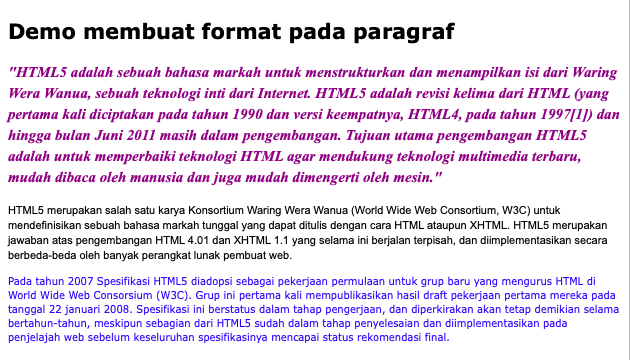
Berikut ini adalah contoh dalam membuat style dan menerapkannya untuk suatu teks di dalam paragraph tertentu. Berikut ini contoh dari pembuatan format pada paragraph

File .HTML





**Tampilan pada Halaman web :**

****

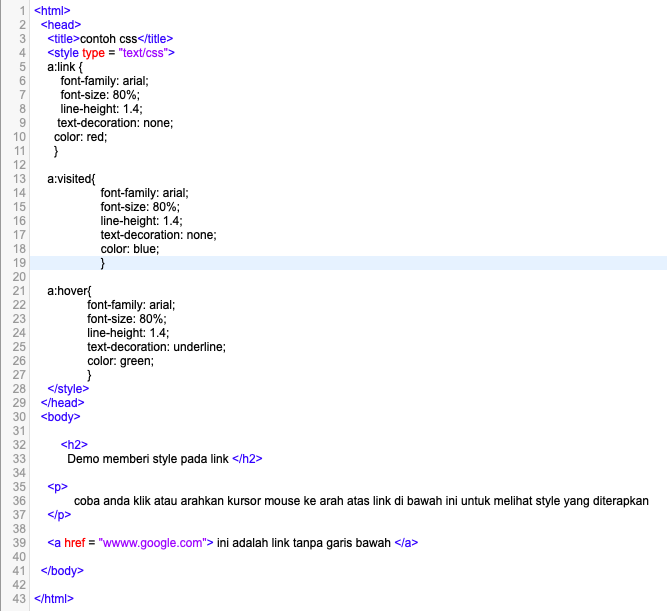
**Membuat style pada link yang berupa teks**

Suatu link dalam dokumen HTML memiliki beberapa keadaan, yaitu :

1. Normal, yaitu link standar atau link yang belum dikunjungi
2. Visited, yaitu link yang sudah pernah dikunjungi
3. Hover, yaitu kursor berada di atas (menunjuk) ke link
4. Active, yaitu keadaan dimana link dipilih atau di – klik
5. Focus, yaitu link dalam keadaan focus dan siap menerima input, seperti klik atau mouse down

Contoh penerapannya :

File .HTML



Tampilan pada halaman web :

* Tampilan dalam keadaan sebelum di arahkan kursor atau mouse (a:link)



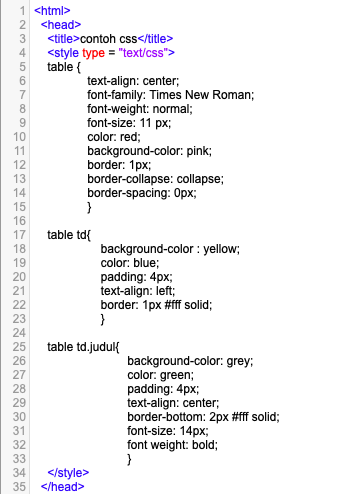
* Tampilan dalam keadaan diarahkan kursor atau mouse (a:hover)

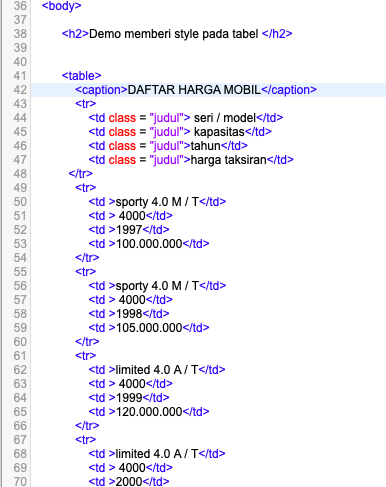


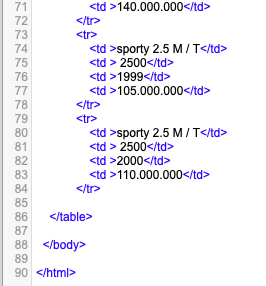
**Membuat Style pada Tabel**

Agar tampilan data pada suatu tabel lebih menarik, maka dapat membuat style khusus pada elemen – elemen tabel seperti <td> dan <th>. Berikut contoh penerapan pembuatan style pada tabel :

FILE .HTML







Tampilan pada Halaman Web :

