

## Praktikum 3 : CSS

(Dosen : Sirojul Munir S.Si,M.Kom)

Tujuan : Mempelajari dan melakukan implementasi CSS  
Dateline : 29 September 2017, 23:55  
Gitlab Project : [https://gitlab.com/NAMA\\_GROUP\\_PROJECT/praktikum03.git](https://gitlab.com/NAMA_GROUP_PROJECT/praktikum03.git)  
Branch Repository : [PRODI\_ROMBEL]\_[NAMASINGKAT]\_[NIM]\_CSS ( ti01\_budi\_0110112001\_CSS)

### Aturan Pengerjaan:

1. Dianjurkan menggunakan sistem OS ubuntu dan apache server
2. Gunakan text editor yang nyaman bagi anda
3. Diperkenankan mengerjakan langsung bagi yang sudah memahami dan menguasai materi
4. Dilarang melakukan tindakan plagiarisme (asisten lab akan mengecek hasil pekerjaan)
  - a. 1x nilai praktikum terkait bernilai 0
  - b. 2x nilai matakuliah pemrograman web E
  - c. 3x mahasiswa akan di sidang komite etik kampus

### Aturan Pengumpulan:

1. Setiap kelas praktikum memiliki **NAMA\_GROUP\_PROJECT** tersendiri, mintalah informasi **NAMA\_GROUP\_PROJECT** gitlab ke asisten lab praktikum !!
2. Buka project gitlab praktikum02 sebelumnya dan lakukan branch untuk menambah fitur CSS pada project sebelumnya
3. Buat **branch repository** sesuai aturan diatas
4. Commit tugas praktikum tidak melebihi batas waktu yang ditentukan
5. Tutorial yang tidak bisa dibuka / compile error / blank page dikenakan pinalti -50.

## Tutorial

### 1. Pengantar CSS

**Cascading Style Sheet (CSS)** adalah salah satu fitur dalam HTML yang memungkinkan anda untuk mengatur tampilan halaman web yang anda bangun. Tidak ada sebuah keharusan dalam menggunakan *style sheet* untuk memformat halaman web, namun menambahkan *style sheet*, dalam hal ini CSS adalah sebuah kelebihan tersendiri untuk tampilan halaman web nantinya.

Bukan itu saja, CSS memungkinkan anda untuk memberi suatu efek-efek khusus pada isi halaman web secara lebih spesifik, dengan lebih sedikit kode yang ditambahkan. Misal, untuk membuat semua teks yang ditulis tebal dan miring, memiliki warna merah, hanya perlu melakukan satu baris tambahan kode untuk melakukan semua itu

CSS dapat digunakan pada satu atau lebih halaman HTML. Untuk mendefinisikan Style beberapa halaman HTML, biasanya CSS disimpan di dalam satu file tersendiri diluar file HTML. Hal ini memudahkan web desainer untuk melakukan perubahan Style, sehingga untuk merubah tampilan beberapa halaman HTML cukup dengan merubah satu file CSS.

Saat ini versi CSS terakhir adalah CSS3, berbeda dengan HTML5 yang didefinisikan sebagai satu kesatuan utuh dari dokumen html, maka CSS3 didefinisikan sebagai modul spesifikasi yang terpisah. Keuntungan dari pemisahan modul ini adalah:

- Lebih mudah dan efisien dalam memperbarui spesifikasi CSS3, karena hanya mengubah bagian yang diinginkan saja
- Karena lebih efisien, maka spesifikasi-spesifikasi tersebut akan update secara lebih cepat

Tetapi cara pendefinisian modular seperti diatas tentu saja bukan tanpa kekurangan. Kekurangan yang paling utama adalah belum terstandarnya CSS3 pada web browser yang ada sekarang, sehingga pengaplikasian CSS3 berbeda pada tiap web browser.

Detail list kompatibel syntak CSS3 bisa dilihat di alamat web ini:

[https://www.w3schools.com/cssref/css3\\_browsersupport.asp](https://www.w3schools.com/cssref/css3_browsersupport.asp)

### 2. Buat branch baru dari project sebelumnya

Untuk memulai praktikum web ini pastikan anda telah mengerjakan praktikum02, untuk kemudian lakukan modifikasi dengan menambahkan CSS pada web yang anda buat

```
$sudo su -  
#cd /var/www/html/praktikum03
```

Buat direktory branch untuk proyek belajar CSS ini sesuai dengan **prodi\_nama\_nim\_CSS** anda, misal:

```
#git checkout -b prodi_nama_nim_css
```

Jangan lupa lakukan commit untuk setiap file yang anda buat

### 3. Penulisan CSS

Sintaks CSS didefinisikan dengan format umum sebagai berikut:

```
selector {
  property : value;
}
```

- Selector : Dapat berupa tag element HTML yang akan didefinisikan atau dapat berupa apa saja.
- Property : Merupakan jenis pengaturan dalam CSS yang akan dilakukan.
- Nilai : Nilai atau masukkan yang sesuai dengan jenis property yang dituliskan.

Terdapat 3 cara penulisan CSS:

#### 1) Inline Style

Inline style dilakukan dengan mendefinisikan style css pada tag html dengan menggunakan atribut style yang berisi property CSS. Cara penulisan inline style memiliki banyak kerugian dibandingkan dengan cara penulisan yang lainnya, ini menyebabkan banyak keuntungan-keuntungan dari style sheet berkurang karena cara penulisan ini mencampurkan sisi tampilan dengan isi yang harus ditampilkan. Gunakanlah cara penulisan ini bilamana kita ingin menerapkan suatu definisi style sekali saja pada sebuah elemen tag HTML.

```
<p style="color:green;margin-left:20px"> Selamat Belajar CSS </p>
```

#### 2) Internal Style

Internal style dilakukan dengan mendefinisikan style css pada tag html dengan menggunakan tag <style> pada header dokumen atau dalam tag <head>. Cara penulisan internal style sheet sebaiknya digunakan jika suatu dokumen tunggal memiliki sebuah unique style.

```
1. <head>
2. <style type="text/css">
3.   hr {color: sienna}
4.   p {margin-left: 20px}
5.   body {background-image: url("images/back40.gif")}
6. </style>
7. </head>
```

#### 3) Eksternal Style

Cara penulisan css dengan cara ini sangat ideal jika style sheet diterapkan pada banyak dokumen (pages). Dengan cara external style sheet, kita dapat merubah keseluruhan tampilan web site dengan hanya merubah satu file css. Setiap halaman atau dokumen harus memiliki link ke file style sheet dengan menggunakan tag <link>. Tag <link> ditulis di dalam bagian head, yaitu diantara tag <head> dan </head>.

```
1.<head>
2.   <link rel="stylesheet" type="text/css" href="css/style.css" />
3.</head>
```

Browser akan membaca semua definisi style dari file style.css yang terletak dalam folder css, dan memformat dokumen sesuai dengan definisi style.

Sebuah file external style sheet dapat ditulis menggunakan text editor, tetapi file tersebut tidak boleh mengandung tag-tag html didalamnya. Dan file style sheet harus disimpan dengan **extension .css**. Contoh file style sheet sbb:

```
hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("assets/img/bg.gif") }
```

Struktur direktori aplikasi web anda menjadi seperti ini:

```
/praktikum02/
| --- index.html
| --- css/
|   --- style.css
|   --- bootstrap.css
| --- assets/
|   --- img/
|       --- bg.gif
```

**Pada praktikum kali ini menggunakan eksternal style !!!**

#### 4. CSS Selector

CSS Selector digunakan untuk memilih elemen HTML yang akan diberikan sebuah style. Terdapat 3 buah Selector yang banyak digunakan pada dokumen HTML yaitu **element selector**, **id selector** dan **class selector**.

##### 1) Element Selector

Berikut ini akan diberikan style pada table yang pernah dibuat dalam project sebelumnya, berikut sintaks yang harus anda tambahkan pada file style.css.

```
...
th {
    background-color:yellow;
    color: #ff000;
}
...
```

Sintaks css diatas akan mengaplikasikan style pada atribut html pada elemen table <th>.

##### 2) ID Selector

Selector ID digunakan pada tag html sesuai dengan atribut ID yang diberikan. Pada contoh praktikum sebelumnya pada file index.html terdapat kode berikut ini:

```
<div id="main" class="active">
    <h1>STT Terpadu Nurul Fikri</h1>
    <p>Kampus Technopreneur di Indonesia</p>
</div>
```

Akan dibuat style untuk h1 dengan diberi ID *"sttnf"*, dengan mengubah kode diatas menjadi kode berikut ini:

```
<div id="main" class="active">
    <h1 id="sttnf">STT Terpadu Nurul Fikri</h1>
    <p>Kampus Technopreneur di Indonesia</p>
</div>
```

Selector ID pada style sheet dipetakan dengan menggunakan tanda pagar # sebelum nama ID. Pada praktikum ini edit file style.css dan tambahkan baris kode css berikut ini:

```
...
#sttnf {
    color: #ff000;
    font-family: serif;
    font-weight: bold;
}
...
```

Sintaks css diatas akan mengaplikasikan style ID Selector #sttnf pada tag html <h1>.

### 3) Class Selector

Atribut class digunakan untuk mendefinisikan style yang dipakai untuk elemen html. Class juga dapat digunakan untuk mendefinisikan style yang berbeda untuk satu tipe elemen yang sama.

Sebagai contoh, kita ingin mendefinisikan dua tipe paragraf, yaitu: satu dengan alignment kanan dan satu dengan alignment kiri:

```
p.right {text-align: right}
p.center {text-align: center}
```

Untuk menggunakan style paragraf diatas, gunakan atribut class di dalam elemen HTML. Contoh:

```
1. <p class="right">
2.     Paragraf ini memiliki alignment kanan.
3. </p>
4. <p class="center">
5.     Paragraf ini memiliki alignment kiri
6. </p>
```

Kita dapat mendefinisikan style dengan memilih selector tanpa menggunakan elemen tag HTML, sehingga style tersebut dapat digunakan oleh beberapa elemen HTML. Contoh:

```
.center {text-align: center}
```

Style di atas dapat digunakan untuk elemen <h1> dan elemen <p> dengan menggunakan atribut class. Berikut contoh penggunaannya:

```
1. <h1 class="center">
2. Header text ini akan memiliki alignment tengah.
3. </h1>
4. <p class="center">
5. Paragraf ini akan memiliki alignment tengah.
```

## 6. &lt;/p&gt;

Berikut ini contoh penerapan selector class pada table. Terlebih dahulu buat file daftar\_harga.html dengan kode html berikut ini:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link type="text/css" rel="stylesheet" href="css/style.css"/>
</head>
<body>

<h1>Daftar Harga Adek Restaurant</h1>

<table>
  <thead>
    <tr >
      <th>No</th>
      <th>Aneka Minuman</th>
      <th>Harga</th>
    </tr>
  </thead>
  <tbody>
    <tr class="row_ganjil">
      <td>1</td>
      <td>Mangga</td>
      <td>10.000</td>
    </tr>
    <tr>
      <td>2</td>
      <td>Alpukat</td>
      <td>12.000</td>
    </tr>
    <tr class="row_ganjil">
      <td>3</td>
      <td>Durian</td>
      <td>15.000</td>
    </tr>
    <tr>
      <td>4</td>
      <td>Jambu</td>
      <td>10.000</td>
    </tr>
  </tbody>
</table>

</body>
</html>
```

Dan berikut ini style css selector didefinisikan pada file css/style.css

```
h1 {
  color:blue;
}

.row_ganjil {
  background-color: blanchedalmond;
```

```

    color: red;
}

```

## 5. CSS Pseudo-class

CSS Pseudo-class adalah select dalam CSS yang berfungsi memberikan efek pada bagian HTML yang tidak bisa diakses dengan selector biasa.

Berikut adalah pseudo-class yang biasa digunakan:

- ✓ :link : untuk menambahkan style pada link yang belum pernah di kunjungi atau klik
- ✓ :hover : untuk menambahkan style pada elemen pada saat mouse berada di atasnya
- ✓ :active : untuk menambahkan style pada elemen yang sedang aktif
- ✓ :visited : untuk menambahkan style pada link yang sudah pernah di kunjungi

Pseudo-class dituliskan dalam format berikut ini:

```

selector:pseudo-class {
    property: value;
}

```

Berikut ini contoh penggunaan pseudo-clas untuk efek pada tag <a href> .

```

a:hover {
    color:red;
}

```

## Praktikum Mandiri

Pada praktikum sebelumnya anda telah membuat halaman web (**landing page**) untuk web **perusahaan tour & travel** menggunakan HTML. Sekarang anda tambahkan style web menggunakan CSS agar websitenya lebih menarik.

Lakukan penulisan CSS menggunakan eksternal style dan beri nama file stylesheet mystyle.css, pada praktikum CSS ini anda diminta implementasi minimal property-property CSS berikut ini telah ada pada web anda:

- ✓ Background: background-color , background-image
- ✓ Text dan Font: : textalign, textdecoration, lineheight, fontsize, fontweight, fontstyle
- ✓ Box Model : margin, border, padding, width
- ✓ Position dan display : position, float, display
- ✓ Navigation dan List : list-style

Selain dari property-property diatas anda juga diminta untuk mencoba transisi dan animasi CSS sederhana.

## Aturan Pengerjaan

- a. Hasil pekerjaan di **commit ke server gitlab** anda sesuai dengan project praktikum 2 ini
- b. Dilarang keras melakukan plagiarisme