

Modul Praktikum Pengenalan Perancangan Web



Disclaimer

Hanya dipergunakan di Lingkungan Internal
Universitas AMIKOM Yogyakarta

KATA PENGANTAR

Puji syukur kami panjatkan kehadiran Tuhan Yang Maha Esa atas berkah dan karunia-Nya, sehingga Modul Praktikum Pengenalan Perancangan Web (PPW) ini dapat diselesaikan. Modul ini dibuat sebagai pedoman dalam melakukan kegiatan praktikum matakuliah Praktikum Pengenalan Perancangan Web yang merupakan salah satu matakuliah pada Jurusan S1 Sistem Informasi Universitas AMIKOM Yogyakarta. Modul ini diharapkan dapat membantu mahasiswa/i, pengembang, atau pun rekan pengajar dalam mempersiapkan dan melaksanakan kegiatan praktikum perkuliahan dengan lebih baik serta menambah wawasan tentang pengenalan web. Pada setiap topik telah ditetapkan tujuan materi yang disampaikan dan semua tugas yang harus dilakukan oleh mahasiswa/i serta teori singkat untuk memperdalam pemahaman mahasiswa/i mengenai materi yang dibahas. Penyusun menyakini bahwa dalam pembuatan modul ini masih jauh dari sempurna. Oleh karena itu penyusun mengharapkan kritik dan saran yang membangun guna penyempurnaan modul ini dimasa yang akan datang

PENGESAHAN

Disusun Oleh	Diperiksa & Dikendalikan Oleh	Disetujui oleh
Irma Rofni Wulandari, S.Pd., M.Eng Aditya Rizki Yudiantika, S.T., M.Eng; Moch Farid Fauzi, M.Kom	Atik Nurmasani, M.Kom	Anggit Dwi Hartanto, M.Kom
Tgl. 5 Maret 2023	Tgl.	Tgl.

***Modul ini syah dan diberlakukan mulai:
Tgl***

Hanif Al Fatta, M.Kom

DAFTAR ISI

(1)	COVER	0
1.	KATA PENGANTAR	1
2.	DAFTAR ISI	2
1.	PERTEMUAN 3 - KONSEP CSS 3	3
1.1.	Tujuan Instruksi Khusus	3
1.2.	Alat dan Bahan	3
1.3.	Dasar Teori	3
3.3.1.	Mengenal CSS dan CSS 3	3
3.3.2.	Skenario Pengaturan Tampilan menggunakan CSS 3	4
3.3.3.	Aturan Penulisan CSS 3 (<i>CSS Syntax</i>)	4
3.3.4.	Cara Menggunakan CSS 3	5
3.3.5.	CSS Comment	10
3.3.6.	Jenis-jenis Selektor	10
1.4.	Instruksi Modul	16
1.5.	Latihan/Tugas	20
1.6.	Sumber Bacaan	20

PERTEMUAN 3 - KONSEP CSS 3

1.1. Tujuan Intruksi Khusus

Setelah mengikuti modul ini, mahasiswa diharapkan mampu mengimplementasikan kerangka dan pengaturan tampilan web menggunakan HTML dan CSS melalui:

- a. Pengenalan Konsep CSS 3
- b. Implementasi CSS 3 pada layout website

1.2. Alat dan Bahan

Alat dan bahan yang diperlukan untuk mengikuti tutorial pada praktikum :

- a. PC Desktop/Laptop
- b. Sistem Operasi
- c. Visual Studio Code : <https://code.visualstudio.com/> atau Sublime : [Sublime Text - Text Editing, Done Right](#)
- d. Web Browser

1.3. Dasar Teori

3.3.1. Mengenal CSS dan CSS 3

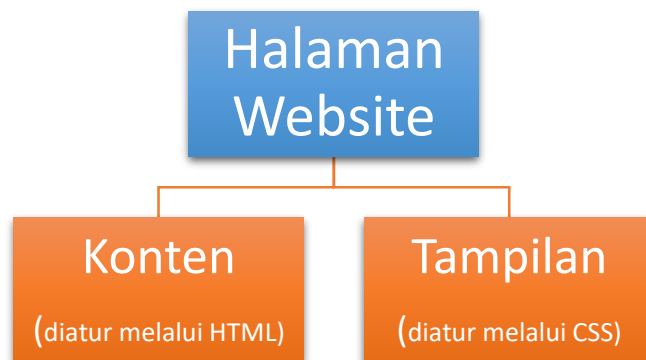
HTML **TIDAK PERNAH** dimaksudkan untuk memformat tampilan halaman web. HTML digunakan untuk menggambarkan konten dalam halaman web [1] dan menunjukkan komponen-komponen penyusun konten dari halaman web. HTML digunakan untuk menggambarkan struktur di dalam web [1].

Sebuah web dapat terdiri dari beberapa halaman. Apabila ingin mengubah tampilan dari halaman web tersebut maka kita harus mengubahnya satu persatu pada setiap halaman web. Hal ini akan menimbulkan kelemahan seperti membuang banyak waktu bagi pengembang web. Kelemahan tersebut dapat diatasi dengan menggunakan CSS. CSS dapat menghemat banyak pekerjaan karena CSS dapat menyimpan format dan dapat digunakan kapanpun [3]. Definisi *style* akan disimpan dalam file .css eksternal sehingga kita dapat mengubah tampilan seluruh situs web hanya dengan mengubah satu file saja.

Cascading Style Sheets (CSS) adalah bahasa yang digunakan untuk mengatur dokumen HTML dan menjelaskan bagaimana elemen HTML harus ditampilkan di layar[1]. CSS merupakan Bahasa yang digunakan untuk memberikan gaya pada dokumen HTML [1]. CSS digunakan untuk mengatur tampilan dari halaman web, termasuk desain, tata letak, dan variasi

tampilan untuk berbagai perangkat dan ukuran layar[1]. CSS dapat mengendalikan beberapa elemen dalam sebuah web sehingga lebih terstruktur dan seragam[2]. Dengan CSS kita dapat mengatur halaman dan kolom, membuat bingkai, menentukan gambar, memberi warna latar belakang, mengatur lebar maupun tinggi, merubah desain dari text, warna, gambar dan latarbelakang dari hampir semua kode tag HTML[4][5][6].

3.3.2. Skenario Pengaturan Tampilan menggunakan CSS 3



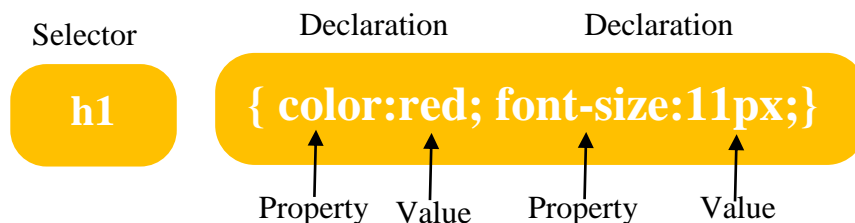
Gambar 3. 1 Skenario Pengaturan Tampilan Web

“HTML FOR CONTENT, CSS FOR PRESENTATION”

3.3.3. Aturan Penulisan CSS 3 (CSS Syntax)

Aturan penulisan CSS berisi set instruksi yang memerintahkan browser untuk mengikuti aturan dan mengubah tampilan elemen HTML berdasarkan pada nilai - nilai yang diberikan [4]. Aturan CSS terdiri dari selektor dan blok deklarasi.

Sintaks CSS :



Keterangan :

1. Selektor merupakan elemen HTML yang akan diberikan *style*. Selektor CSS digunakan untuk “menemukan” (atau memilih) elemen dari HTML.

2. *Block Declaration* berisi satu atau lebih deklarasi style yang dipisahkan dengan titik koma dan berada didalam kurung kurawal.
3. Setiap Deklarasi (*Declaration*) menyertakan nama properti CSS (*Property*) dan nilainya (*Value*)



Gambar 3. 2 Aturan CSS

Aturan penulisan CSS yaitu :

1. CSS bersifat *case-sensitive*, penulisan NIM berbeda dengan Nim
2. Pemberian nama usahakan memiliki makna. Misalkan : menggunakan nama “banner” untuk mendeklarasikan selektor banner, dan bukan dengan “bnr”.
3. Tanda (-) maupun (_) diperbolehkan untuk penulisan nama
4. Penulisan nama dapat menggunakan angka tetapi spasi atau tanda lain tidak diperkenankan.

3.3.4. Cara Menggunakan CSS 3

Ada tiga (3) cara menyisipkan *style sheet* :

a. Inline CSS

Inline Style Sheet dapat digunakan untuk menerapkan style untuk satu elemen saja. Cara ini dapat digunakan dengan menyisipkan langsung script CSS ke dalam tag HTML yang diinginkan. **Cara ini sebaiknya hanya digunakan apabila anda ingin memformat suatu elemen satu kali saja.** Sintaks *Inline Style Sheet*:

```
<tag HTML style="Properti:value/nilai">...</tag HTML>
```

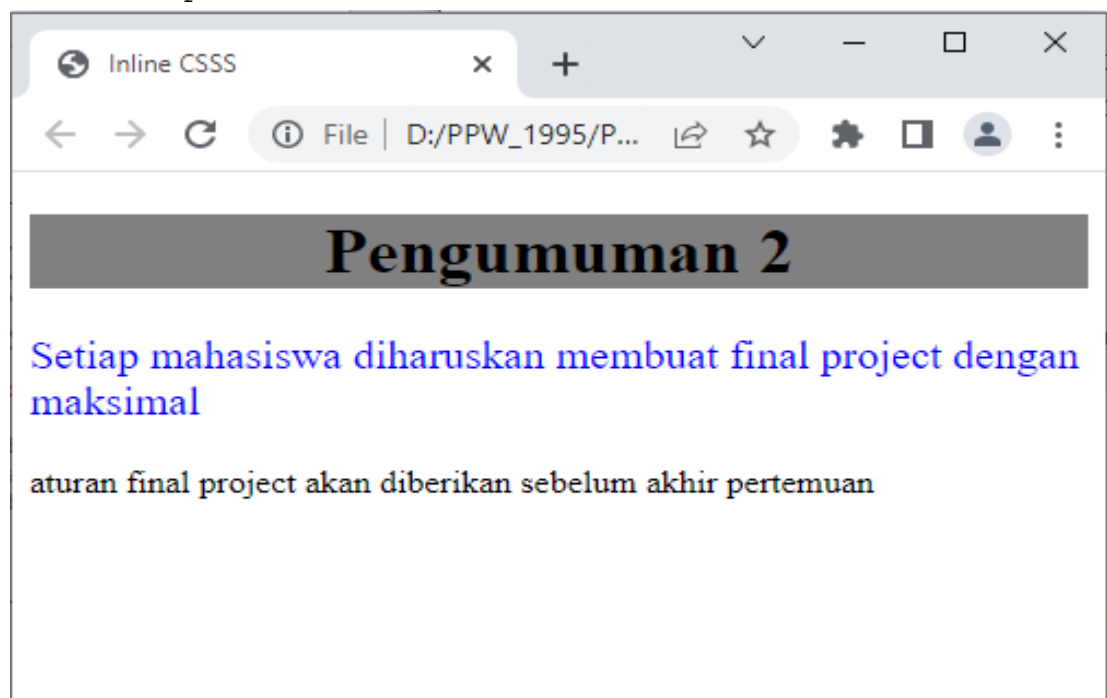
Contoh Inline CSS :

Nama File : inlinecss.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Latihan CSS Inline</title>
</head>
<body>
  <h1 style="background-color:grey; color:black; text-align:center">Pengumuman 2</h1>
  <p style="font-size: 20px; color: blue">Setiap Mahasiswa
    diharuskan membuat final project dengan maksimal.</p>
  <p>aturan final project akan diberikan sebelum akhir
    pertemuan.</p>
</body>
</html>
```

CSS BERADA DI
DALAM TAG
HTML

Hasil Output :



b. Internal CSS

Internal CSS atau *Embedded Style sheet* digunakan dengan menyisipkan kode CSS kedalam bagian `<head>` dan `</head>`. Penulisan CSS dengan cara ini diawali dengan tag `<style>` dan diakhiri dengan tag `</style>`.

Sintaks Internal CSS :

```
<style type = "text/css">
    <!-- atribut -->
</style>
```

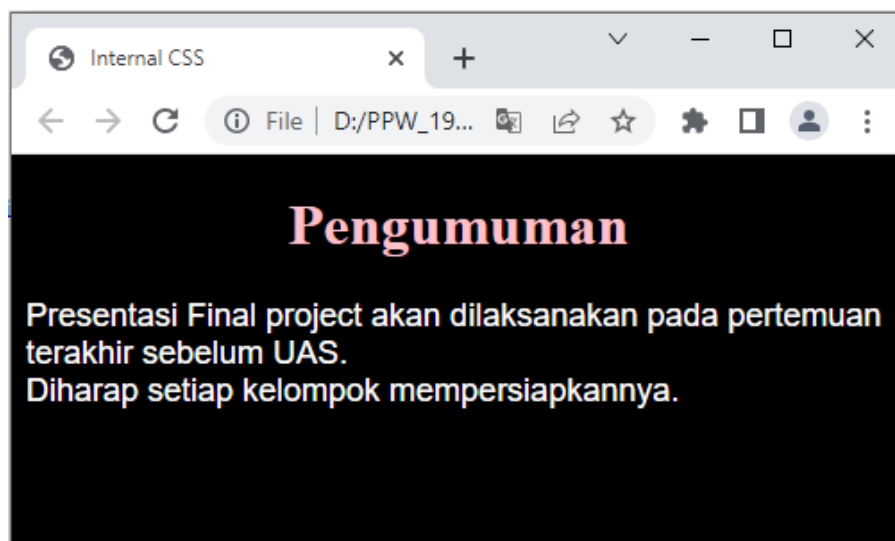
Contoh Internal CSS :

Nama File : internalcss.html

```
<!DOCTYPE html>
<html>
<head>
    <title>LATIHAN CSS EMBEDDED</title>
    <style type="text/css">
        body{
            background-color: black;
            color: white;
        }
        h1{
            text-align: center;
            color: pink;
        }
        p{
            text-align: left;
            font-size: 18px;
            font-family: arial;
        }
    </style>
</head>
<body>
    <h1>Pengumuman</h1>
    <p>Presentasi Final Project akan dilaksanakan pada
        pertemuan terakhir sebelum UAS. <br/>
        Diharap setiap kelompok mempersiapkannya.
    </p>
</body>
</html>
```

KODE CSS BERADA
DI DALAM FILE
HTML

Hasil Output :



c. External CSS

Dengan menggunakan *eksternal style sheet*, anda dapat mengubah tampilan seluruh situs web hanya dengan mengubah satu file saja. Setiap halaman HTML harus menyertakan referensi ke file *style sheet eksternal* di dalam elemen `<link>`, di dalam bagian `<head>`. Kode CSS Eksternal ditulis dalam **satu file terpisah yang disimpan dengan ekstensi .css**. File eksternal .css ini harus dipanggil terlebih dahulu pada semua halaman web yang memerlukannya. File .css eksternal tidak boleh berisi tag HTML apapun. Sintaks *Linked Style Sheet* :

```
<link rel="stylesheet" href="namafile.css" type="text/css">
```

Aturan penggunaan:

- Kode HTML dan CSS dipisahkan
- Lebih mudah untuk mengatur dan mengubah sebuah tampilan
- Dianjurkan untuk digunakan pada pengerjaan sebuah desain web
- Kode CSS dijalankan dengan cara memanggil file *.css yang telah dibuat

Catatan :

Simpanlah *.css dan file-file css berikutnya dalam folder terpisah dengan nama folder "CSS". Foldering seperti ini penting agar lebih rapi dan memudahkan proses editing karena file css tidak bercampur dengan file-file lain (HTML, PHP, dsb) [3].

Contoh *Linked Style Sheet/External CSS*:

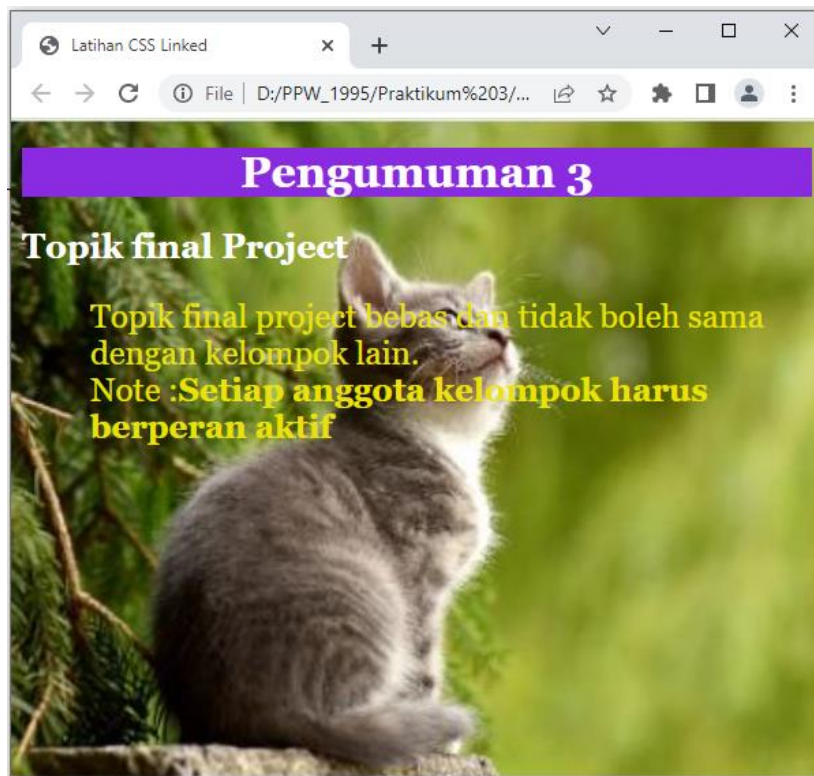
Nama File : externalcss.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Latihan CSS Linked</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <h1>Pengumuman 3</h1>
  <h2>Topik final Project</h2>
  <p>Topik final project bebas dan tidak boleh sama dengan kelompok lain. <br/>
  Note :<b>Setiap anggota kelompok harus berperan aktif</b></p>
</body>
</html>
```

Nama File : style.css

```
body{
  background-image: url(image/cat.JPG);
  background-size: cover;
  background-repeat: no-repeat;
  background-position: center;
  background-attachment: fixed;
  height: 100%;
  font-family: Georgia, 'Times New Roman', Times, serif;
}
h1{
  background-color: blueviolet;
  text-align: center;
  color: white;
}
h2{
  color: white;
  font-size: 25px;
}
p{
  margin-left: 50px;
  color: rgb(235, 227, 7);
  font-size: 24px;
}
```

Hasil Output :



3.3.5. CSS Comment

Penulisan komentar pada CSS ditempatkan pada `<style>` elemen, dan dimulai dengan `/*` dan diakhiri dengan `*/`. Komentar digunakan untuk menjelaskan kode dan dapat membantu membuat catatan untuk mengedit kode sumber suatu hari nanti. Komentar tidak akan ditampilkan ke browser. Sintaks komentar :

```
/* This is a single-line comment */
```

3.3.6. Jenis-jenis Selektor

Selector CSS digunakan untuk “menemukan ” atau memilih elemen HTML yang akan diberi pengaturan tampilan. Selektor dapat dibagi menjadi lima kategori yaitu :

1. *Simple selectors* yaitu pilihan elemen berdasarkan nama, id, kelas. Simple selector terdiri dari :

a. Tag / Elemen HTML

Menggunakan tag yang terdapat pada HTML. Setiap tag yang ada dalam HTML dapat dijadikan selektor. Contoh sintaks :

```
h1 {  
    color : blue;  
}  
p {  
    text-align: center;  
    color: red;  
}
```

b. ID

Selektor id digunakan untuk menentukan style bagian unik dari html. Unik disini artinya satu nama id hanya bisa digunakan satu kali pada sebuah halaman web. Apabila satu nama ID digunakan lebih dari satu kali pada sebuah halaman, maka style hanya akan berlaku pada ID yang pertama saja. Penggunaan selector ID akan diawali dengan **tanda pagar (#)**. Selector ID mendefinisikan tag secara individual. Tag ID hanya boleh digunakan dalam sebuah elemen saja. Nama id tidak boleh diawali dengan angka. Contoh Sintaks :

```
#bagian1 {  
    text-align:left;  
    color:white;  
}  
#atas {
```

```
background-color:red;
color:white;
font-size:50px;
text-align:center;
}
```

c. Class

Selector class digunakan untuk menentukan style dari sebuah grup elemen. Berbeda dengan ID, selector class dapat dipakai berulang kali pada sebuah dokumen web. Artinya sebuah nama class dapat dipanggil beberapa kali pada elemen-elemen yang ada pada sebuah halaman web. Penggunaan selektor kelas akan diawali dengan **tanda titik** (.) di awal penulisannya. Kemudian pada tag HTML ditambahkan class = (nama kelas).

Contoh sintaks :

```
.body {
    font-family : verdana;
    font-size: 30px;
}
```

Dalam contoh ini semua elemen HTML dengan class="body" akan memiliki font verdana dan ukuran font 30px.

```
.isi {
    font-family:arial;
    font-size:30px;
    color:blue
}
```

Anda juga dapat menentukan bahwa hanya elemen HTML tertentu yang harus terpengaruh oleh suatu kelas :

```
p.center {
    text-align: center;
    color: red;
}
```

Dalam contoh ini hanya elemen <p> dengan class="center" yang akan berwarna merah dan rata tengah.

d. Universal

Selektor Universal (*) digunakan untuk memilih semua elemen HTML pada halaman web. Contoh sintaks :

```
*{  
    Text-align: center;  
    Color: blue;  
}
```

e. Grouping Selector

Grouping selector digunakan untuk memilih semua elemen HTML dengan definisi gaya yang sama. Lihat contoh berikut ini, elemen h1, h2, dan p memiliki definisi style yang sama :

```
h1 {  
    text-align: center;  
    color: red;  
}  
  
h2 {  
    text-align: center;  
    color: red;  
}  
  
p {  
    text-align: center;  
    color: red;  
}
```

Hal ini akan lebih baik untuk mengelompokkan selector dengan style yang sama, untuk meminimalkan penulisan code. Untuk mengelompokkan selector, pisahkan setiap selector dengan koma. Contoh hasil pengelompokan :

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

f. Tag DIV dan SPAN

Tag DIV dan SPAN digunakan untuk mengelompokkan element-element HTML. Span digunakan untuk mendefinisikan *inline content*, sementara div digunakan untuk *block level content*. Perbedaan dari kedua tag tersebut adalah sebagai berikut :

- 1) Tag div akan membagi halaman web secara otomatis meskipun di dalam tag div tersebut tidak diatur format stylenya (css). Digunakan untuk *block level conten*.

- 2) Tag span hanya akan mengatur konten atau isi yang dilingkupinnya tanpa membagi halaman web ke dalam bagian-bagian kecil (seperti yang dilakukan tag div). Digunakan untuk mendefinisikan *inline content*.

2. *Combinator selectors* yaitu memilih elemen berdasarkan hubungan spesifik.

Sebuah kombinator adalah sesuatu yang menjelaskan hubungan antar selektor. Sebuah selektor CSS dapat berisi lebih dari satu selektor sederhana. Diantara selektor sederhana, kita dapat menyertakan kombinator. Ada empat kombinator berbeda pada CSS :

a) *Descendant selector (spasi)*

Descendant selector cocok dengan semua elemen yang merupakan turunan dari elemen tertentu. Contoh berikut memilih semua elemen `<p>` di dalam elemen `<div>` sehingga seluruh p didalam div memiliki *background color* berwarna kuning:

```
div p {  
    background-color: yellow;  
}
```

b) *Child selector (>)*

Child selector memilih semua elemen yang merupakan anak dari elemen tertentu. Contoh berikut memilih semua elemen `<p>` yang merupakan anak dari elemen `<div>` sehingga seluruh p akan memiliki background color berwarna kuning.

```
div > p {  
    background-color: yellow;  
}
```

c) *Adjacent sibling selector (+)*

Adjacent sibling selector digunakan untuk memilih elemen yang secara langsung setelah elemen tertentu lainnya. Elemen saudara harus memiliki elemen induk yang sama, dan "berdekatan" berarti "segera mengikuti". Contoh berikut memilih elemen `<p>` pertama yang ditempatkan setelah elemen `<div>`:

```
div + p {  
    background-color: yellow;  
}
```

d) *General sibling selector (~)*

General sibling selector (~) digunakan untuk memilih semua elemen yang merupakan saudara kandung berikutnya dari elemen tertentu. Contoh berikut memilih semua elemen <p> yang merupakan saudara kandung berikutnya dari elemen <div>:

```
div ~ p {  
    background-color: yellow;  
}
```

3. *Pseudo-class selectors* yaitu memilih elemen berdasarkan status tertentu. *Pseudo-class selector* digunakan untuk mendefinisikan keadaan khusus suatu elemen. Pseudo-class selector dapat digunakan untuk :

- a. Memberikan style pada elemen saat pengguna mengarahkan mouse ke atasnya
- b. Memberikan style pada tautan yang dikunjungi dan yang belum dikunjungi secara berbeda
- c. Memberikan style elemen saat mendapatkan fokus.

Sintaks *pseudo-class*:

```
selector:pseudo-class {  
    property: value;  
}
```

Contoh *pseudo-class* pada link :

```
<style>  
/* unvisited link */  
a:link {  
    color: red;  
}  
  
/* visited link */  
a:visited {  
    color: green;  
}  
  
/* mouse over link */  
a:hover {  
    color: hotpink;  
}  
  
/* selected link */  
a:active {  
    color: blue;  
}  
</style>
```

4. *Pseudo-elements selectors* yaitu memilih dan menata bagian elemen. *Pseudo – elements selector* digunakan untuk menata bagian tertentu dari suatu elemen. Dapat digunakan untuk :

- a. Memberikan style huruf pertama, atau baris, dari suatu elemen
- b. Menyisipkan konten sebelum, atau setelah, konten elemen

Sintaks elemen pseudo:

```
selector::pseudo-element {  
    property: value;  
}
```

5. *Attribute selectors* digunakan untuk memilih elemen berdasarkan atribut atau nilai atribut. Dimungkinkan untuk menata elemen HTML yang memiliki atribut atau nilai atribut tertentu. Selector [attribute] digunakan untuk memilih elemen dengan atribut tertentu. Contoh berikut memilih semua <a> elemen dengan atribut target pada link:

```
a[target] {  
    background-color: yellow;  
}
```


1.4. Instruksi Modul

3.4.1 Internal dan Inline CSS

1. Cobalah script dibawah ini dan amati hasil outputnya :
2. Simpan dengan nama File : **praktikum3a_zzzz.html** (zzzz = 4 digit nim terakhir anda)
3. Kembangkan tampilan dengan memodifikasi CSS

```
Praktikum 3 > <> praktikum3a.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Internal dan Inline CSS</title>
8      <style type="text/css">
9          body{
10             background-color: #palegoldenrod;
11             color: white;
12          }
13          h1{
14             text-align: center;
15             color: black;
16          }
17          h2,p {
18             text-align: left;
19             font-size: 18px;
20             font-family: arial;
21          }
22      </style>
23 </head>
24 <body>
25     <h1>Latihan Internal CSS</h1>
26     <h2>Pengumuman</h2>
27     <p>Ini adalah contoh paragraf yang akan menggunakan inline css <br> semua mahasiswa wajib mempraktikkannya</p>
28     <br>
29     <h1 style="color: blue;text-align:center;">Ini Inline CSS</h1>
30     <p style="color: red;">Paragraf ini menggunakan Inline CSS</p>
31
32 </body>
33 </html>
--
```

3.4.2 External CSS

Aturan penggunaan *Linked Style Sheet* / External CSS:

- Kode HTML dan CSS dipisahkan
- Dianjurkan untuk digunakan pada pengerjaan sebuah desain web
- Kode CSS dijalankan dengan cara memanggil file *.css yang telah dibuat

Langkah pertama adalah buatlah terlebih dahulu file HTMLnya dan simpan dengan nama File : **praktikum3b_zzzz.html** (zzzz = 4 digit nim terakhir anda)

Praktikum 3 > <> praktikum3b.html > ...

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>External css</title>
8      <link rel="stylesheet" href="style1.css">
9  </head>
10 <body>
11     <h1>This is a heading</h1>
12     <h2>This is a heading 2</h2>
13     <p>This is a paragraph.</p>
14 </body>
15 </html>
```

Langkah berikutnya adalah buat file baru kemudian ubah menjadi file CSS (pada plaintext pojok kanan VSCode ganti menjadi CSS). Kemudian simpan dengan nama file : style1.css (ingat ekstensi file wajib CSS dan isi didalamnya adalah CSS bukan HTML).

Nama file : style1.css

style1.css X

Praktikum 3 > # style1.css > ...

```
1  body{
2      background-image: url(image/cat.JPG);
3      background-repeat: repeat;
4      background-position: left top;
5      background-attachment: fixed;
6      font-family: Georgia;
7  }
8  h1{
9      background-color: blueviolet;
10     text-align: center;
11     color: white;
12 }
13 h2{
14     color: red;
15     font-size: 25px;
16 }
17 p{
18     margin-left: 50px;
19     color: white;
20     font-size: 24px;
21 }
```

Kembangkan dan tambahkan CSS agar tampilan dan background menjadi lebih menarik.

3.4.3 Menggunakan Selector , DIV dan SPAN

Langkah pertama adalah buatlah terlebih dahulu file HTMLnya dan simpan dengan nama File : praktikum3c_zzzz.html

```
<> praktikum3c.html X
Praktikum 3 > <> praktikum3c.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Selector, Div dan Span</title>
8      <link rel="stylesheet" href="aturan.css">
9  </head>
10 <body>
11     <h1>AKU</h1>
12     <div id="note">(Penggalan Puisi Aku, Chairil Anwar)</div>
13     <div class="isi">
14         <div id="bagian1">
15             <p>Kalau sampai waktuku</p>
16             <p>Kutak mau seorang kan merayu</p>
17         </div>
18         <p>Tidak juga kau</p>
19         <p>Tak perlu sedu sedan itu</p>
20     </div>
21     <div class="isi" id="bagian2" >
22         <p>Aku ini binatang jalang</p>
23         <p>Dari kumpulannya terbang</p>
24         <p>Biar peluru menembus kulitku</p>
25         <p>Aku tetap meradang menerjang</p>
26     </div>
27     <div>
28         <p>Ini Contoh Penggunaan DIV</p>
29     </div>
30     <span style="font-family: serif;color: #FF0000; font-size:24px">
31         Ini Contoh dalam Span dengan warna merah
32     </span>
33
34 </body>
35 </html>
```

Langkah berikutnya adalah buat file baru kemudian ubah menjadi file CSS (pada plaintext pojok kanan VSCode ganti menjadi CSS). Kemudian simpan dengan nama file : **aturan.css** (ingat ekstensi file wajib CSS dan isi didalamnya adalah CSS bukan HTML).

Nama File : aturan.css

```
# aturan.css X
Praktikum 3 > # aturan.css > ...
1  h1{
2      background-color: blueviolet;
3      text-align: center;
4      color: white;
5  }
6
7  .isi{
8      background-color: grey;
9  }
10
11 #bagian1{
12     text-align: left;
13     font-size: 20px;
14     font-family: verdana;
15     color: darkblue;
16 }
17
18 #bagian2{
19     text-align: right;
20     font-family: arial;
21     color: beige;
22 }
23
24 #note{
25     text-align: center;
26     font-size: 16px;
27     font-family: verdana;
28 }
29
30 div{
31     font-family: sana-serif;
32     color: #0000FF;
33     background-color: #FFFF00;
34     font-size: 36px;
35     text-align: center;
36     width: 80%;
37 }
```

Hasil Output terlihat kurang proposional dan tampil tidak full. Identifikasi penyebabnya dan buat tampilan menjadi satu tampilan penuh seperti gambar 1.



Gambar1. Tampilan penuh praktikum 3c

1.5. Latihan/Tugas

1. Praktikkan praktikum3a_zzzz.html, praktikum3b_zzzz.html, praktikum3c_zzzz.html.
2. Kumpulkan File .html, .css beserta asset (contoh : gambar) yang anda gunakan.
3. Letakkan pada satu folder dengan format penamaan NIM_P3. Kemudian kompress folder dalam bentuk .ZIP, dan kumpulkan ke waskita.
4. Setelah mengumpulkan hasil praktikum anda ke waskita kemudian Kerjakan kuis yang ada pada waskita
5. Kuis bersifat individu

1.6. Sumber Bacaan

- [1] W3Schools. HTML Tutorial. [Online]. Available : [HTML Tutorial \(w3schools.com\)](https://www.w3schools.com/html/). [Accessed March, 19, 2022]
- [2] Khafidli, M. Firgiawan. (2011). Trik Menguasai HTML5, CSS3, PHP Aplikatif. Yogyakarta: Loko Media
- [3] Hidayatullah, P., & Kawistara, J.K., (2017). Pemrograman Web. Bandung: Informatika Bandung
- [4] Tim Dosen Pengenalan Perancangan Web (2021). Modul Praktikum PPW. Universitas AMIKOM Yogyakarta
- [5] Arifin, Oki. (2017). Modul Pemrograman Web Lanjut. Yogyakarta : Universitas AMIKOM Yogyakarta.

- [6] Farida, Lilis Dwi. (2017). Modul Pengenalan Perancangan Web . Yogyakarta: Universitas AMIKOM Yogyakarta.
- [7] Enterprize, Jubile, (2014). HTML 5 Manual Book, Penerbit Elex Media Komputindo: Jakarta
- [8] Sidik,Betha.Ir, Pohan, I.Husni. (2012). Pemrograman Web dengan HTML. Bandung : Informatika.