

Modul Praktikum
PEMROGRAMAN WEB



Disusun Oleh:
Mardhiyyah Rafrin ST., M.Sc

PROGRAM STUDI ILMU KOMPUTER
JURUSAN TEKNOLOGI PRODUKSI DAN INDUSTRI
INSTITUT TEKNOLOGI BACHARUDDIN JUSUF HABIBIE
2024/2024



Table of Contents

Modul Praktikum.....	<i>i</i>
MODUL I	<i>1</i>
<i>Hypertext Markup Language</i>	<i>1</i>
1.1. Tujuan	1
1.2. Indikator Penilaian	1
1.3. Pendahuluan.....	1
1.4. Elemen HTML	1
1.5. Semantik Elemen HTML	2
1.6. Non-Semantik Elemen HTML.....	3
1.7. Elemen Meta	3
1.8. Atribut HTML.....	7
1.9. Tabel.....	7
1.10. Form HTML.....	10
1.11. Media	17
1.12. Penomoran.....	19
MODUL 2	<i>21</i>
<i>Cascading Style Sheets</i>	<i>21</i>
2.1. Tujuan	21
2.2. Indikator Penilaian	21
2.3. Pengenalan Cascading Style Sheets.....	21
MODUL 3	<i>30</i>
<i>HTML EVENTS.....</i>	<i>30</i>



MODUL I

Hypertext Markup Language

1.1. Tujuan

Tujuan mempelajari HTML ini adalah mahasiswa mampu membuat halaman web dengan menggunakan HTML.

1.2. Indikator Penilaian

Setelah mempelajari Hypertext Markup Language, mahasiswa diharapkan mampu:

1. Memahami semantic dan non-semantic HTML
2. Membuat tabel dengan HTML pada halaman web
3. Membuat form dengan HTML serta menggunakan atribut form dengan tepat pada halaman web
4. Menambahkan media dengan HTML pada halaman web

1.3. Pendahuluan

HyperText Markup Language atau HTML adalah blok bangunan paling dasar dari Web. HTML berfungsi untuk menyusun struktur konten web. Konten web dapat berupa teks, gambar, audio, video dan sebagainya. Saat ini versi HTML terbaru adalah HTML5 yang dirilis pada tahun 2008. Teknologi lain selain HTML umumnya digunakan untuk membuat tampilan atau presentasi halaman web ([CSS](#)) atau fungsionalitas/perilaku ([JavaScript](#)).

"Hypertext" mengacu pada tautan yang menghubungkan halaman web satu sama lain, baik dalam satu situs web atau antar situs web. Tautan adalah aspek mendasar dari Web. Dengan mengunggah konten ke Internet dan menghubungkannya ke halaman yang dibuat oleh orang lain, Anda menjadi peserta aktif di World Wide Web.

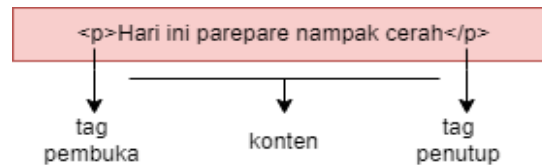
HTML menggunakan "markup" untuk membubuhi keterangan pada teks, gambar, dan konten lainnya untuk ditampilkan di browser Web. Markup HTML mencakup "elemen" khusus seperti [<head>](#), [<title>](#), [<body>](#), [<header>](#), [<footer>](#), [<article>](#), [<section>](#), , [<p>](#), [<div>](#), [](#), [](#), [<aside>](#), [<audio>](#), [<canvas>](#), [<datalist>](#), [<details>](#), [<embed>](#), [<nav>](#), [<search>](#), [<output>](#), [<progress>](#), [<video>](#), [](#), dan banyak lainnya.

1.4. Elemen HTML

HTML terdiri dari serangkaian elemen yang digunakan untuk membungkus konten agar konten tampak atau berperilaku tertentu. Sebagai contoh, jika kita ingin kalimat “Hari ini Parepare tampak cerah” menjadi sebuah konten paragraf pada halaman web, maka kita perlu membungkus konten tersebut dengan element [<p>](#) sehingga hasilnya

[<p>](#)Hari ini Parepare nampak cerah[</p>](#)

Anatomi dari HTML diatas adalah sebagai berikut:



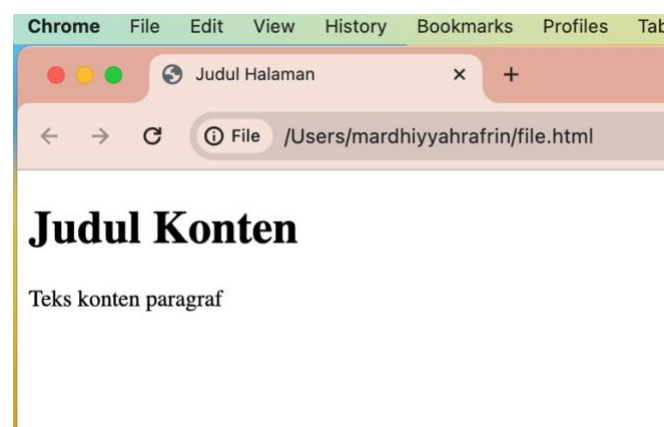
Gambar 1.1. Anatomi elemen HTML

- Tag pembuka yang terdiri dari nama elemen (pada contoh diatas menggunakan elemen *p*) yang dibungkus dengan kurung sudut `<>`.
- Konten merupakan isi dari elemen. Pada contoh di atas, konten ini adalah kalimat *Hari ini Parepare nampak cerah*.
- Tag penutup yang terdiri dari nama elemen yang sama dengan tag pembuka namun sebelum nama elemen ditambahkan garis miring (/) didalam kurung sudut. Kesalahan yang sering terjadi dalam mebangun halaman web adalah tidak memberikan tag penutup sehingga halaman web terlihat berantakan.

Struktur HTML secara sederhana terdiri atas elemen head dan elemen body. Pada baris pertama sebelum membuat elemen head dan body, kita menggunakan tag `<!DOCTYPE>` untuk memberi tahu browser bahwa dokumen ini adalah HTML. Pada HTML5, tag ini ditambahkan atribut `html` sehingga menjadi `<!DOCTYPE html>`. Perhatikan code HTML di bawah ini.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Judul Halaman</title>
  </head>
  <body>
    <h1>Judul Konten</h1>
    <p>Teks konten parahrat</p>
  </body>
</html>
```

Saat dijalankan di browser, hasil code di atas adalah sebagai berikut



Gambar 1.2. Halaman web sederhana

1.5. Semantik Elemen HTML

Semantik elemen digunakan untuk memudahkan programmer dan browser mengenali elemen-elemen HTML. Berikut ini beberapa semantik elemen.

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>



Gambar 1.3 Bagian-bagian pada halaman web yang menggunakan semantik elemen. Sumber: w3schools.com

1.6. Non-Semantik Elemen HTML

Non-Semantik elemen merupakan elemen yang tidak memiliki makna khusus atau penjelasan khusus mengenai isi kontennya. Contoh non-semantik elemen adalah <div> dan .

1.7. Elemen Meta

Tag <meta> digunakan sebagai informasi atau data yang dikandung oleh dokumen HTML dan diletakkan di dalam elemen head. Contoh penggunaan meta:

1. Untuk mendeskripsikan keyword yang dikandung dokumen HTML
`<meta name="keywords" content="HTML, CSS, JavaScript">`



2. Untuk mendeskripsikan isi konten dokumen HTML.
`<meta name="description" content="Tutorial HTML dan CSS gratis ">`
3. Untuk menambahkan nama author (penulis)
`<meta name="author" content="Mardhiyyah Rafrin">`
4. Untuk mengatur viewport agar terlihat bagus pada berbagai ukuran perangkat
`<meta name="viewport" content="width=device-width, initial-scale=1.0">`

`width=device-width` menandakan lebar halaman dokumen sama dengan lebar perangkat dan `initial-scale=1.0">` menandakan bahwa zoom halaman saat dibuka pertama kali sama dengan 1 atau ukuran normal.

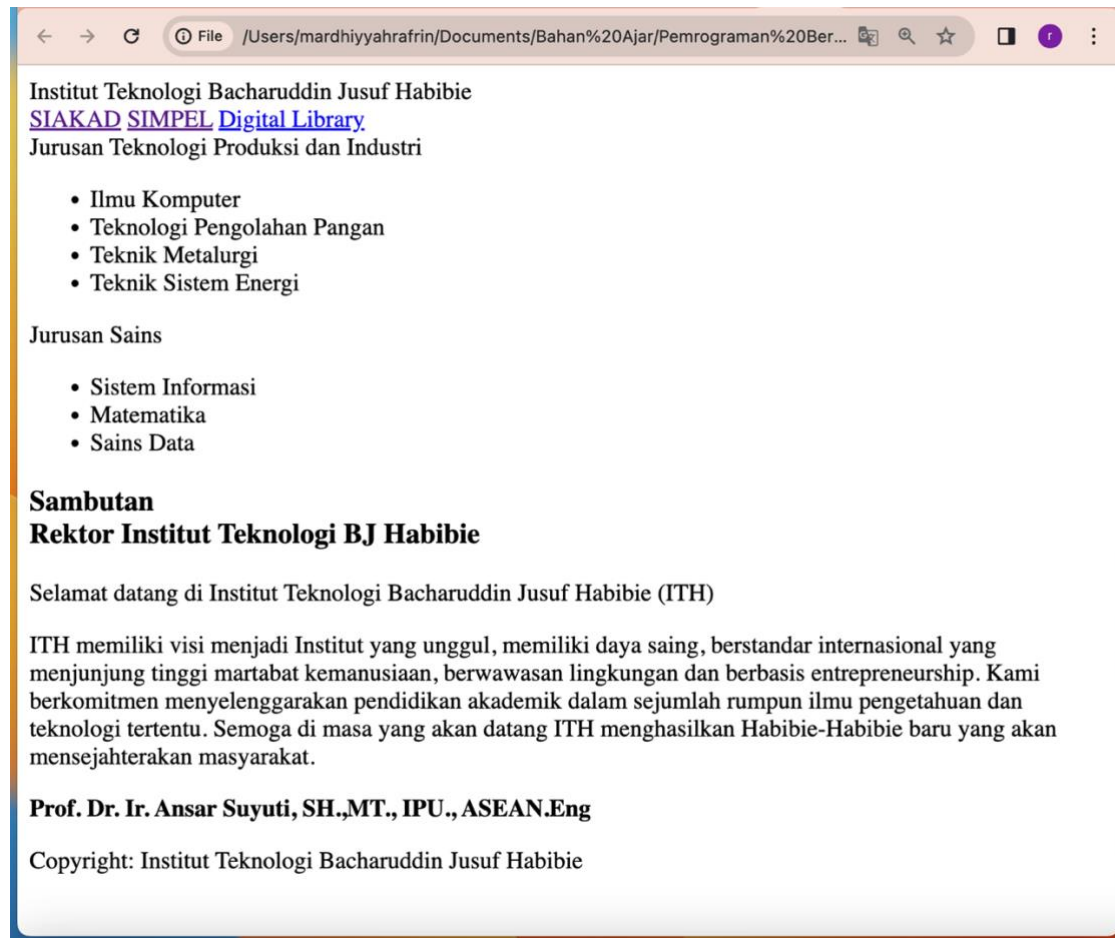


Praktikum 1. Buatlah halaman HTML sederhana dengan menggunakan kode HTML berikut ini!

```
<> halaman.html > html > body > div
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Pemrograman Bebas Web</title>
7    </head>
8    <body>
9      <header>
10     <div>Institut Teknologi Bacharuddin Jusuf Habibie</div>
11     <nav>
12       <a href="https://siakad.ith.ac.id/">SIAKAD</a>
13       <a href="https://simpl.ith.ac.id/">SIMPEL</a>
14       <a href="https://digilab.ith.ac.id/">Digital Library</a>
15     </nav>
16   </header>
17   <div>
18     <aside>
19       <div>
20         <div>Jurusan Teknologi Produksi dan Industri</div>
21         <ul>
22           <li>Ilmu Komputer</li>
23           <li>Teknologi Pengolahan Pangan</li>
24           <li>Teknik Metalurgi</li>
25           <li>Teknik Sistem Energi</li>
26         </ul>
27       </div>
28       <div>
29         <div>Jurusan Sains</div>
30         <ul>
31           <li>Sistem Informasi</li>
32           <li>Matematika</li>
33           <li>Sains Data</li>
34         </ul>
35       </div>
36     </aside>
37     <section>
38       <h3>Sambutan <br />Rektor Institut Teknologi BJ Habibie</h3>
39       <p>Selamat datang di Institut Teknologi Bacharuddin Jusuf Habibie (ITH)</p>
40       <p>
41         ITH memiliki visi menjadi Institut yang unggul, memiliki daya saing, berstandar internasional yang
42         menjunjung tinggi martabat kemanusiaan, berwawasan lingkungan dan berbasis entrepreneurship. Kami berkomitmen
43         menyelenggarakan pendidikan akademik dalam sejumlah rumpun ilmu pengetahuan dan teknologi tertentu. Semoga di
44         masa yang akan datang ITH menghasilkan Habibie-Habibie baru yang akan mensejahterakan masyarakat.
45       </p>
46       <p><strong>Prof. Dr. Ir. Ansar Suyuti, SH.,MT., IPU., ASEAN.Eng</strong></p>
47     </section>
48   </div>
49   <footer>
50     Copyright: Institut Teknologi Bacharuddin Jusuf Habibie
51   </footer>
52 </body>
53 </html>
```

Gambar 1.4 Halaman HTML sederhana

Setelah selesai, simpan file dengan ekstensi .html. Contoh nama file **halaman.html**. Setelah disimpan, klik 2 kali pada file dan lihat hasilnya pada browser anda.



Gambar 1.5 Output dari source code pada gambar 1.5

Agar tampilan lebih menarik, tambahkan tautan **style.css** ke dalam halaman web. File ini berisi kode css yang akan kita pelajari pada materi berikutnya. Cara menambahkan tautan tersebut adalah dengan menambahkan elemen **link** pada bagian head seperti pada baris 7 di bawah ini!

```
2  <html>
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Pemrograman Bebas Web</title>
7    <link rel="stylesheet" type="text/css" href="style.css" />
8  </head>
```

Gambar 1.6. Cara menambahkan eksternal CSS ke dalam HTML

Selain menambahkan 1 baris elemen tersebut, tambahkan file **style.css** yang telah disediakan ke dalam folder yang sama Anda menyimpan kode html di atas. Setelah itu, refresh halaman browser anda dan hasilnya akan tampil seperti di bawah ini.



Gambar 1.7. Cara menambahkan eksternal CSS ke dalam HTML

1.8. Atribut HTML

Atribut HTML merupakan bagian dari elemen HTML yang menyediakan informasi tambahan mengenai elemen HTML tersebut. Setiap elemen memiliki atribut tertentu. Umumnya atribut terdiri dari nama atribut dan nilai atribut. Perhatikan contoh di bawah ini.

```
<a href="https://siakad.ith.ac.id/">SIAKAD</a>
```

Pada kode di atas, **a** merupakan elemen, **href** merupakan nama atribut, <https://siakad.ith.ac.id> merupakan nilai dari atribut href.

```
<div class="prodi" id="prodi-tpi">Jurusan Teknologi Produksi dan Industri</div>
```

Pada kode di atas, **div** merupakan nama elemen, **class** merupakan atribut dan *prodi* merupakan nilai dari atribut class, **id** juga merupakan atribut dan nilainya adalah *prodi-tpi*. Sedangkan teks “Jurusan Teknologi Produksi dan Industri” merupakan konten dari elemen div

1.9. Tabel

Elemen dalam membuat tabel adalah `<table>` dan di dalam elemen tabel terdapat elemen-elemen untuk mendeskripsikan baris dan kolom.

Tabel. 1.1 Elemen Tabel

No	Elemen	Deskripsi
1	<code><table></code>	Mendefinisikan table
2	<code><th></code>	Mendefinisikan table head



3	<tr>	Mendefinisikan baris
4	<td>	Mendefinisikan sell
5	<colgroup>	Mendefinisikan grup kolom untuk kebutuhan format kolom table.
6	<col>	Mendefinisikan kolom yang ada dalam <colgroup>
7	<thead>	Mendefinisikan konten pada grup header dalam tabel
8	<tbody>	Mendefinisikan konten pada grup body dalam tabel
9	<tfoot>	Mendefinisikan konten pada grup footer dalam tabel



Perhatikan contoh kode membuat tabel di bawah ini!

```
<table border="1">
  <thead>
    <tr>
      <th>Judul Kolom 1</th>
      <th>Judul Kolom 2</th>
      <th>Judul Kolom 3</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>sel 1.1</td>
      <td>sel 1.2</td>
      <td>sel 1.3</td>
    </tr>
    <tr>
      <td>sel 2.1</td>
      <td>sel 2.2</td>
      <td>sel 2.3</td>
    </tr>
  </tbody>
</table>
```

Gambar 1.8. Contoh HTML tabel sederhana

Hasil tabel yang dibuat adalah:

Judul Kolom 1	Judul Kolom 2	Judul Kolom 3
sel 1.1	sel 1.2	sel 1.3
sel 2.1	sel 2.2	sel 2.3

Gambar 1.9. Output tabel dari source code 1.8

Untuk melakukan penggabungan sel, kita gunakan atribut `colspan` untuk menggabungkan kolom dan `rowspan` untuk menggabungkan baris. Nilai dari atribut tersebut merupakan integer jumlah kolom atau baris yang akan digabung. Perhatikan kode html di bawah ini, kita membuat header tabel untuk nama, memiliki atribut **`colspan="2"`**.

```
<table>
  <tr>
    <th colspan="2">Nama</th>
    <th>Prodi</th>
  </tr>
  <tr>
    <td>John</td>
    <td>Doe</td>
    <td>Ilmu Komputer</td>
  </tr>
  <tr>
    <td>Jane</td>
    <td>Doe</td>
    <td>Sistem Informasi</td>
  </tr>
</table>
```

Gambar 1.10. Contoh HTML tabel dengan menggabungkan sel (kolom)



Nama		Prodi
John	Doe	Ilmu Komputer
Jane	Doe	Sistem Informasi

Gambar 1.11. Output tabel dari source 1.10

Untuk melakukan penggabungan baris, kita bisa menambahkan atribut `rowspan` dengan nilai jumlah baris yang akan digabung. Berikut ini contohnya

```
<table border="1">
  <tr>
    <th>Nama</th>
    <td>John Doe</td>
  </tr>
  <tr>
    <th rowspan="2">Nomor Telepon</th>
    <td>123-4567</td>
  </tr>
  <tr>
    <td>123-123</td>
  </tr>
</table>
```

Gambar 1.12. Contoh HTML tabel dengan menggabungkan sel (baris)

Nama	John Doe
Nomor Telepon	123-4567
	123-123

Gambar 1.13. Output tabel dengan source code 1.12

1.10. Form HTML

Form merupakan salah satu elemen HTML yang digunakan untuk mendapatkan input dari pengguna web. Form HTML merupakan elemen yang paling sering dipakai untuk mengumpulkan dan mengirim data user ke server. Elemen form diawali dengan tag `<form>`. Elemen-elemen di dalam form bisa dilihat dalam tabel 1.2. berikut.

Tabel 1.2. Elemen Form

No	Elemen	Deskripsi
1	<code><form></code>	Digunakan untuk mendeskripsikan form
2	<code><input></code>	Elemen untuk mendapatkan input user berdasarkan tipe input
3	<code><label></code>	Untuk memberi label pada elemen input dalam form
4	<code><select></code>	Untuk mendapatkan pilihan user dalam bentuk drop-down
5	<code><textarea></code>	Untuk mendapatkan input user berupa teks yang lebih dari satu baris. (misalnya satu paragraph)



6	<fieldset>	Fieldset digunakan untuk mengelompokkan elemen dalam form yang memiliki data berkaitan
7	<legend>	Digunakan untuk caption sebuah fieldset
8	<datalist>	Digunakan sebagai untuk menspesifikasikan pilihan input user
9	<output>	Untuk menampilkan output
10	<option>	Untuk menampilkan pilihan dalam drop-down
11	<optgroup>	Untuk menampilkan kelompok pilihan yang berkaitan dalam drop-down

1.10.1. Attribut Form

Tabel 1.3. Daftar atribut form

Attribut	Deskripsi
accept-charset	Mendeskripsikan tipe karakter yang dapat diterima saat form disubmit. Contohnya UTF-8
action	Menspesifikasikan tujuan data form disubmit. Nilai atribut adalah string
autocomplete	Mendeskripsikan apakah form dilengkapi secara otomatis atau tidak. Atribut ini adalah boolean
enctype	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
method	Menspesifikasikan method HTTP yang digunakan saat mengirim data dari form (form-data)
name	Mendeskripsikan nama form
novalidate	Menspesifikasikan jika form tidak divalidasi ketika disubmit
rel	Specifies the relationship between a linked resource and the current document
target	Spesifikasikan dimana respon akan ditampilkan setelah form disubmit.

1.10.2 Element Form HTML

Beberapa elemen form yang sering dipakai elemen input, select, textarea dan button.

a. Input

Input memiliki banyak tipe untuk membedakan jenis input dari user. Tipe-tipe input dapat dilihat pada daftar di bawah ini.

- `<input type="button" />`
- `<input type="checkbox" />`

- `<input type="color" />`
- `<input type="date" />`
- `<input type="datetime-local" />`
- `<input type="email" />`
- `<input type="file" />`
- `<input type="hidden" />`
- `<input type="image" />`
- `<input type="month" />`
- `<input type="number" />`
- `<input type="password" />`
- `<input type="radio" />`
- `<input type="range" />`
- `<input type="reset" />`
- `<input type="search" />`
- `<input type="submit" />`
- `<input type="tel" />`
- `<input type="text" />`
- `<input type="time" />`
- `<input type="url" />`
- `<input type="week" />`

Perhatikan contoh form sederhana dengan elemen input tipe teks, password, dan date.

```
<form>
  <div>
    <label for="username">Username</label>
    <input type="text" name="username" id="username" placeholder="Masukkan Username" />
  </div>
  <div>
    <label for="password">Password</label>
    <input type="password" name="password" id="password" placeholder="Masukkan Password" />
  </div>
  <div>
    <label for="tanggal_lahir">Tanggal Lahir</label>
    <input type="date" name="tanggal_lahir" id="tanggal_lahir" />
  </div>
</form>
```

Gambar 1.14. Contoh source code menggunakan form input

Username

Password

Tanggal Lahir

February 2024

↑

↓

S	M	T	W	T	F	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	1	2
3	4	5	6	7	8	9

Clear

Today

Gambar 1.15. Output form dengan source code 1.14

b. Select

Elemen `<select>` digunakan untuk mendapatkan pilihan user dengan menggunakan drop-down. Di dalam tag `<select>`, harus terdapat tag `<option>` atau `<optgroup>` sebagai pilihan yang tersedia. Perhatikan code select pada contoh di bawah ini dan silahkan salin ke halaman html anda untuk melihat hasilnya!

```
<div>
  <label for="kota">Domisili</label>
  <select id="kota" name="kota">
    <option>Parepare</option>
    <option>Barru</option>
    <option>Sidrap</option>
    <option>Pinrang</option>
  </select>
</div>
```

Gambar 1.16. Sourcecode sederhana menggunakan elemen select dan option

c. Textarea

Textarea digunakan untuk input user yang terdiri dari beberapa baris. Cara menggunakannya adalah:

```
<textarea rows="7" cols="30">
  Hari ini Parepare sangat panas
</textarea>
```

d. Button

Button merupakan elemen HTML yang dapat diklik. Cara menggunakannya adalah:

```
<button type="button">Click Me!</button>
```



Terdapat 3 macam nilai tipe pada button yaitu tipe submit yang digunakan untuk submit form, tipe reset yang digunakan untuk menghapus isi, dan tipe button (tidak memiliki tujuan yang spesifik).

1.10.3. Atribut Elemen Input

Atribut digunakan untuk memberi informasi tambahan sebuah elemen. Atribut pada form dan input sangat penting pada form untuk memudahkan user mengisi data dan untuk memudahkan pengiriman data ke server. Berikut ini atribut yang digunakan pada form dan input.

Tabel. 1.4. Daftar atribut form

No	Atribut Form	Nilai Atribut	Fungsi
1	type	text, radio, checkbox, dll	Untuk mendeskripsikan tipe input
2	value	String	Untuk memberi nilai pertama saat halaman dibuka
3	name	String	Untuk memberi nama elemen yang akan digunakan dalam mengirim data ke server
4	placeholder	String	Untuk memberi petunjuk singkat mengenai data yang diharapkan dari user pada elemen input
5	readonly	Boolean	Untuk membuat input hanya dapat dibaca, tidak dapat diisi. Nilai input dapat disalin dan dapat dikirim ke server
6	disabled	Boolean	Untuk membuat input hanya dapat dibaca, tidak dapat diisi. Nilai input tidak dapat digunakan dan tidak dapat dikirim ke server
7	min dan max	Number	Untuk menetapkan Batasan paling rendah dan paling tinggi untuk input tipe number, range, date, datetime-local, month, time and week
9	pattern	Regex	Untuk menetapkan pattern input yang dapat diterima
10	multiple	Boolean	Untuk membuat input dapat diisi lebih dari sekali. Atribut ini hanya digunakan oleh input tipe file dan email
11	size	Number	Untuk mengatur jumlah maksimal karakter input. Atribut ini hanya digunakan pada input tipe teks, url, email, password, search, dan tel. Nilai default adalah 20
14	autofocus	Boolean	Untuk memfokuskan elemen input pada saat halaman dibuka pertama kali
15	autocomplete	Boolean	
12	width dan height	Number	Untuk mengatur lebar dan tinggi input tipe gambar (image)



- | | | | |
|----|----------|---------|--|
| 13 | required | Boolean | Untuk mengatur input wajib diisi. Jika dikosongkan, form tidak dapat disubmit. |
|----|----------|---------|--|

Praktikum 2. Buatlah Form dengan menggunakan source code HTML di bawah ini!

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Formulir Pendaftaran</title>
7      <link rel="stylesheet" type="text/css" href="form.css" />
8    </head>
9    <body>
10     <form>
11       <fieldset>
12         <legend>Data Pribadi</legend>
13         <div>
14           <label for="username">Username</label>
15           <input type="text" name="username" id="username" placeholder="Masukkan Username" />
16         </div>
17         <div>
18           <label for="password">Password</label>
19           <input type="password" name="password" id="password" placeholder="Masukkan Password" />
20         </div>
21         <div>
22           <label for="kota">Domisili</label>
23           <select id="kota" name="kota">
24             <option>Parepare</option>
25             <option>Barru</option>
26             <option>Sidrap</option>
27             <option>Pinrang</option>
28           </select>
29         </div>
30         <div>
31           <label for="tempat_lahir">Tempat Lahir</label>
32           <input type="text" id="tempat_lahir" name="tempat_lahir" />
33           <datalist id="daftar_tempat_lahir">
34             <option value="Parepare">
35             <option value="Barru">
36             <option value="Sidrap">
37             <option value="Pinrang">
38           </datalist>
39           </select>
40         </div>
41         <div>
42           <label for="tanggal_lahir">Tanggal Lahir</label>
43           <input type="date" name="tanggal_lahir" id="tanggal_lahir" />
44         </div>
45         <div>
46           <label for="jenis_kelamin">Jenis Kelamin</label>
47           <div class="flex">
48             <input type="radio" name="jenis_kelamin" value="Laki-laki" id="l" /> <span>Laki-Laki</span>
49             <input type="radio" name="jenis_kelamin" value="Perempuan" id="p" /> <span>Perempuan</span>
50           </div>
51         </div>
52       </fieldset>
```



```
53     <fieldset>
54         <legend>Data Tambahan</legend>
55         <label for="jenis_kelamin">Study Club yang diikuti</label>
56         <div class="flex">
57             <input type="checkbox" name="jenis_kelamin" value="Laki-laki" id="coding_club" />
58             <span>Habibie Coding Club</span>
59         </div>
60         <div class="flex">
61             <input type="checkbox" name="jenis_kelamin" value="Perempuan" id="techno_club"/>
62             <span>Technopreneurship</span>
63         </div>
64         <div class="flex">
65             <input type="checkbox" name="jenis_kelamin" value="Perempuan" id="english_club"/>
66             <span>English Club</span>
67         </div>
68         <div>
69             <label for="passphoto">Foto</label>
70             <input type="file" name="passphoto" id="passphoto" />
71         </div>
72         <div>
73             <label for="pendapatan">Pendapatan Orang Tua</label>
74             <div class="flex">
75                 <span>500.000</span>
76                 <input type="range" name="pendapatan" id="pendapatan" />
77                 <span>1.000.000</span>
78             </div>
79         </div>
80         <div>
81             <label for="warna">Warna Latar Profil</label>
82             <input type="color" name="warna" id="warna" placeholder="Pilih warna latar" />
83         </div>
84         <div>
85             <label for="alasan">Alasan Memilih ITH</label>
86             <textarea name="alasan" id="alasan" placeholder="Pilih warna latar"></textarea>
87         </div>
88     </fieldset>
89 </form>
90 </body>
91 </html>
```

Agar tampilan teratur, tambahkan file **form.css** yang telah disediakan ke dalam folder yang sama dengan file form yang anda buat. Jika anda mengikuti code HTML dan menambahkan file form.css dengan benar, tampilan halaman web tersebut seperti tampilan di bawah ini.



The screenshot shows a web browser window with a URL: `/Users/mardhiyyahrafrin/Documents/Bahan%20Ajar/Pemrograman%20Berbasis%20Web/Bahan%20Modul/form.html`. The form is titled "Formulir Pendaftaran" and is divided into two main sections: "Data Pribadi" and "Data Tambahan".

Data Pribadi:

- Username:
- Password:
- Domisili:
- Tempat Lahir:
- Tanggal Lahir:
- Jenis Kelamin: ☐ Laki-Laki ☐ Perempuan

Data Tambahan:

- Study Club yang diikuti: ☐ Habibie Coding Club, ☐ Technopreneurship, ☐ English Club
- Foto: No file chosen
- Pendapatan Orang Tua: 500.000 to 1.000.000
- Warna Latar Profil:
- Alasan Memilih ITH:

1.11. Media

Multimedia pada halaman web merupakan gambar, audio, video, animasi dan lain sebagainya yang dapat ditayangkan pada halaman web. HTML juga dapat mendukung penyematan multimedia. Format multimedia yang dapat ditampilkan bervariasi tergantung browser yang digunakan.

Tabel 1.5. Elemen Media

No	Elemen	Deskripsi
1	<code></code>	Menampilkan gambar
2	<code><video></code>	Menampilkan video
3	<code><audio></code>	Menampilkan audio
4	<code><object></code>	Menampilkan objek yang dapat berupa dokumen atau halaman situs lain
5	<code><embed></code>	Menampilkan objek yang dapat berupa dokumen atau halaman situs lain
6	<code><iframe></code>	Menampilkan objek yang dapat berupa dokumen atau halaman situs lain

Contoh HTML code menampilkan gambar:

```

```

Contoh HTML code menampilkan video:

```
<video width="500" height="500" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">
```



Browser anda tidak mendukung penayangan video.

</video>

Elemen source dengan atribut src digunakan untuk memberi informasi file video. Saat halamana web dimuat, file video pada elemen <source> pertama yang akan dimuat pertama kali. Jika browser yang digunakan tidak mendukung tipe atau format video, maka browser akan memuat elemen <source> yang kedua dan seterusnya. Jika tidak ada elemen <source> yang dapat dimuat browser, akan tampil text Browser anda tidak mendukung penanganan video.

Cara yang sama juga berlaku pada elemen audio.

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Browser Anda tidak mendukung pemutar audio
</audio>
```

Perhatikan baik pada elemen audio dan video terdapat attribute controls. Atribut ini digunakan untuk memberi keleluasaana kepada user untuk memainkan ataupun menghentikan pemutaran video/audio selayaknya aplikasi pemutar video/audio. Jika atribut controls dihilangkan, user tidak dapat mengendalikan media tersebut. Selain attribute controls, ada beberapa attribute lain yang bisa digunakan. Atribut tersebut dapat dilihat pada tabel di bawah ini.

Tabel 1.6. Atribut elemen media

No	Atribut	Deskripsi
1	controls	Menampilkan control video/audio
2	autoplay	Memutar audio/video secara otomatis saat halaman pertama kali dimuat
3	muted	Menghilangkan suara media saat pertama kali halaman dimuat
4	width	Mengatur lebar media
5	height	Mengatur tinggi media

Selain audio dan video, kita juga dapat menggunakan plug-ins seperti <object>, <embed> dan <iframe>. Elemen ini dapat digunakan untuk menampilkan frame halaman web, dokumen pdf, foto, video dan sebagainya.

Contoh penggunaan iframe.

```
<iframe width="90%" height="500px"
  src="https://www.youtube.com/embed/tgbNymZ7vqY">
</iframe>
```

Contoh penggunaan elemen <object>.

```
<object width="100%" height="500px"
  data="https://siakad.ith.ac.id/"></object>
```

Contoh penggunaan elemen <embed>.



```
<embed width="100%" height="500px" src="document.pdf">
```

1.12. Penomoran

Penomoran juga dapat dilakukan menggunakan HTML dengan tiga macam penomoran, pertama adalah **unordered list** dan kedua adalah **ordered list** dan terakhir adalah **description list**. Elemen yang dapat digunakan dapat dilihat pada tabel 1.7.

Tabel 1.7. Daftar elemen penomoran

No	Elemen	Deskripsi
1		ordered list
2		unordered list
3		Membuat item di dalam list
4	<dl>	description list
5	<dt>	Membuat term/istilah di dalam list deskripsi
6	<dd>	Membuat deskripsi istilah atau term

Onordered list merupakan penoram tanpa urutan tetapi hanya menggunakan tanda seperti dot sedangkan ordered list merupakan penomoran dengan urutan 1, 2, 3, dan seterusnya. Perhatikan contoh dibawah ini untuk ordered dan unordered list.

Contoh ordered list:

```
<ol>
  <li>Merah</li>
  <li>Kuning</li>
  <li>Hijau</li>
</ol>
```

Hasil ordered list di atas:

1. Merah
2. Kuning
3. Hijau

Contoh unordered list:

```
<ul>
  <li>Merah</li>
  <li>Kuning</li>
  <li>Hijau</li>
</ul>
```

Hasil unordered list di atas:

- Merah
- Kuning



- Hijau

Contoh deskripsi list:

```
<dl>
  <dt>Kopi</dt>
  <dd>- minuman panas tanpa gula</dd>
  <dt>Susu</dt>
  <dd>- minuman yang baik untuk kesehatan</dd>
</dl>
```

Hasil description list di atas:

Kopi	- minuman panas tanpa gula
Susu	- minuman yang baik untuk kesehatan

TUGAS 1. Membuat Halaman Web Menggunakan HTML

Buatlah sebuah halaman tabel dimana halaman tersebut terdiri dari konten berupa form, tabel, gambar, video, audio, file pdf, serta konten dengan menggunakan penomoran ordered list, unordered list dan description list.



MODUL 2

Cascading Style Sheets

2.1. Tujuan

Tujuan mempelajari *Cascading Style Sheets* adalah agar mahasiswa mampu mendesain tampilan web agar lebih representatif

2.2. Indikator Penilaian

Indikator penilaian pada modul 2 ini adalah:

1. Mahasiswa mampu menggunakan sintaks CSS dengan benar
2. Mahasiswa mampu menggunakan CSS selector dengan tepat.

2.3. Pengenalan Cascading Style Sheets

Cascading Style Sheets atau sering disingkat css merupakan sebuah bahasa yang digunakan untuk mengatur bagaimana elemen HTML ditampilkan atau direpresentasikan di halaman web. Representasi HTML ini dapat berupa posisi, ukuran, warna, dan sebagainya. CSS juga dapat mengatur bagaimana halaman web ditampilkan di berbagai ukuran layar perangkat yang digunakan oleh user. Sintaks CSS dapat dibuat di halaman yang sama dengan HTML, dapat juga dibuat menggunakan file eksternal yang disimpan dengan menggunakan ekstensi **.css**.

2.3 Sintaks CSS

Sintaks CSS sangat sederhana, hanya terdiri dari selector dan blok kode css yang ditandai dengan kurung kurawal ({ }). Isi blok kode tersebut adalah beberapa property dan nilai. Setiap property ditutup dengan tanda titik koma (;). Untuk lebih jelasnya, perhatikan sintaks di bawah ini.

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

Perhatikan contoh di bawah ini untuk mengetahui selector, property dan value!

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

dimana **p** adalah selector, yaitu elemen HTML yang dipilih untuk diatur tampilannya yang dalam hal ini adalah elemen `<p>` atau paragraf. **color** dan **text-align** merupakan property sedangkan **red** dan **center** merupakan nilai dari properti. Setiap satu pasang properti:nilai dipisahkan dengan semicolon (;). Properti **color** digunakan untuk mengatur warna teks, dalam hal ini warna teks diubah menjadi merah sedangkan **text-align** merupakan property yang digunakan untuk mengatur rata teks, dalam hal ini teks diatur menjadi rata tengah.

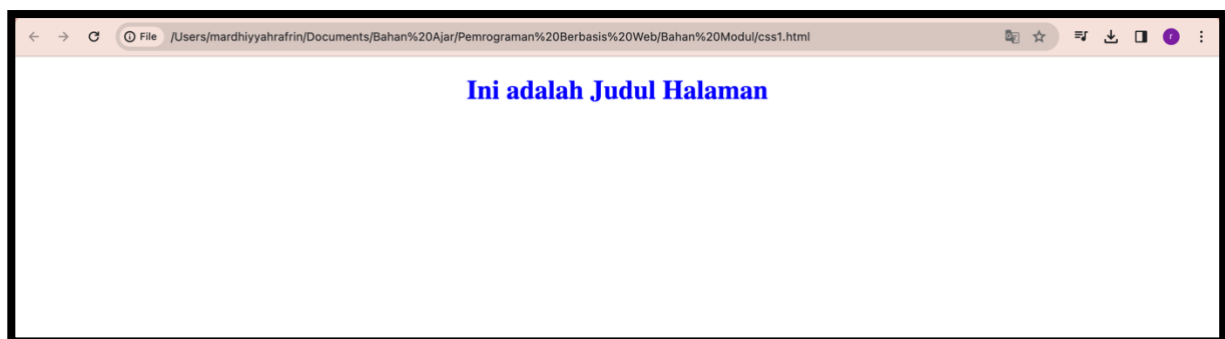


Kode CSS dapat digunakan pada internal HTML maupun eksternal HTML. Jika CSS dibuat dalam internal HTML, maka harus dibuat dalam elemen **<style>** ataupun dalam **attribute style**.

Contoh di bawah ini merupakan code CSS di dalaman elemen **<style>**

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      h1 {
        color: blue;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <h1>Ini adalah Judul Halaman </h1>
  </body>
</html>
```

Hasil dari code di atas adalah sebagai berikut!



Contoh di bawah ini merupakan code CSS yang merupakan nilai dari atribut style.

```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <h1 style="color:blue; text-align:center;">Ini adalah Judul Halaman
  </h1>
  </body>
</html>
```

Adapun jika menggunakan eksternal HTML, maka file CSS disimpan dengan ekstensi .css dan ditautkan ke dalam HTML dengan menggunakan elemen **<link>**. Perhatikan contoh di bawah ini!

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="style.css">
```




```
</head>
<body>
  <h1>Ini adalah Judul Halaman </h1>
</body>
</html>
```

Pada code HTML di atas, file CSS disimpan dengan nama `style.css`. Perhatikan pula folder lokasi file ini di simpan. Pastikan HTML dapat terhubung dengan file CSS. Pada file CSS, elemen style sudah tidak diperlukan untuk memulai kode CSS. Cukup selector, properti dan nilai propertinya saja.

Penggunaan CSS juga dapat digabung antara internal dan eksternal. Kode CSS dieksekusi secara berurut dari baris awal hingga baris akhir. Jika kita menulis pada file `style.css` dengan kode:

```
h1 {
  color: black;
  text-align: left;
}
```

Kemudian kita membuat CSS pada internal HTML dengan elemen `<style>` seperti di bawah ini, maka warna `h1` akan menjadi blue.

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="style.css">
    <style>
      h1 {
        color: blue;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <h1 style="color:red; text-align:center;">
      Ini adalah Judul Halaman
    </h1>
  </body>
</html>
```

Mengapa warna `h1` berubah jadi biru bukan hitam? Karena HTML dieksekusi dari baris awal hingga akhir. Warna `h1` black yang dibuat di file `style.css` dibuat di baris pertama dalam elemen `<head>`, sedangkan code CSS dalam elemen `<style>` dibuat di baris berikutnya sehingga CSS inilah yang akan di eksekusi. Perhatikan code di bawah ini! Warna apakah elemen `<h1>` saat tampil di browser?

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="style.css">
    <style>
```



```
h1 {  
    color: blue;  
    text-align: center;  
}  
</style>  
</head>  
<body>  
    <h1 style="stylesheet">Ini adalah Judul Halaman</h1>  
</body>  
</html>
```

2.4. CSS Selectors

Selectors digunakan untuk memilih elemen-elemen yang ingin kita atur menggunakan CSS. Ada beberapa cara menggunakan selectors.

- *Basic selectors* (memilih elemen berdasarkan nama elemen, id, atau class)
- *Combinator selectors* (memilih elemen berdasarkan hubungan yang spesifik)
- *Pseudo-class selectors* (memilih elemen berdasarkan tahapan tertentu)
- *Pseudo-element selectors* (memilih elemen khusus pada keadaan tertentu)
- *Attribute selectors* (memilih elemen berdasarkan atribut atau nilai atribut sebuah elemen)

2.4.1 Basic Selectors

- a. Berdasarkan elemen

Misalnya kita ingin mengatur tampilan elemen input, maka code css nya adalah:

```
input {  
    color: black;  
    width: 23rem;  
    height: 1.8rem;  
}
```

- b. Berdasarkan id

Misalkan kita ingin mengatur elemen dengan ID *input_username*, seperti di bawah ini

```
<input type="text" id="input_username" />
```

Maka selectors harus diawali dengan karakter tagar (#) diikuti id elemen. Contohnya:

```
#input_username {  
    color: black;  
    width: 23rem;  
    height: 1.8rem;  
}
```

- c. Berdasarkan class

Misalkan kita ingin mengatur elemen input dengan class="*form_input*" seperti di bawah ini:

```
<input type="text" id="input_username" class="form_input" />
```



Maka selectors harus diawali dengan karakter titik (.) diikuti nama class elemen. Jika melihat elemen input diatas, maka nama classnya adalah `form_input` sehingga kode CSS nya menjadi:

```
.form_input {
  color: black;
  width: 23rem;
  height: 1.8rem;
}
```

e. Pengelompokan Elemen

Kita juga dapat mengatur tampilan beberapa elemen hanya dengan menggunakan sebuah blok kode CSS. Caranya dengan menggabung elemen yang dipisahkan dengan menggunakan tanda koma (,) pada selector. Contoh di bawah ini merupakan blok kode CSS untuk mengubah elemen dengan id `input_username`, elemen dengan class `form_input`, juga semua elemen `<div>` dalam satu blok code CSS.

```
#input_username, form_input, div {
  color: black;
  width: 23rem;
  height: 1.8rem;
}
```

2.4.2 Combinator Selectors

Combinator selectors merupakan pemilihan elemen HTML berdasarkan relasi antar elemen, diantaranya:

Combinator Selectors	Contoh	Deskripsi
<i>descendant selectors</i>	<code>div p</code>	Memilih semua elemen <code><p></code> yang berada dalam elemen <code><div></code>
<i>child selectors</i>	<code>div > p</code>	Memilih semua elemen <code><p></code> yang merupakan anak dari elemen <code><div></code>
<i>general sibling selector</i>	<code>div ~ p</code>	Memilih semua elemen <code><p></code> yang letaknya setelah elemen <code><div></code>
<i>adjacent sibling selector</i>	<code>div + p</code>	Memilih satu elemen <code><p></code> yang letaknya setelah elemen <code><div></code>

2.4.3 Pseudo-class selectors

Pseudo-class selectors digunakan untuk memilih elemen berdasarkan keadaan tertentu. Selector ini bisa digunakan saat user menggunakan mouse, saat user telah meng-klik tautan atau link ataupun ketika elemen HTML sedang difokuskan dan sebagainya.

<code>:active</code>	<code>a:active</code>	Memilih tautan yang sedang aktif
<code>:checked</code>	<code>input:checked</code>	Memilih elemen input dengan tipe checkbox yang sedang dicentang



<u>:disabled</u>	input:disabled	Memilih elemen input yang menggunakan atribut disabled
<u>:empty</u>	p:empty	Memilih setiap elemen p yang tidak memiliki anak elemen
<u>:enabled</u>	input:enabled	Memilih elemen input yang tidak disabled
<u>:first-child</u>	p:first-child	Selects every <p> elements that is the first child of its parent
<u>:first-of-type</u>	p:first-of-type	Selects every <p> element that is the first <p> element of its parent
<u>:focus</u>	input:focus	Selects the <input> element that has focus
<u>:hover</u>	a:hover	Selects links on mouse over
<u>:in-range</u>	input:in-range	Selects <input> elements with a value within a specified range
<u>:invalid</u>	input:invalid	Selects all <input> elements with an invalid value

Praktikum 3. Membuat Form dengan style CSS

Buatlah sebuah form login sederhana dengan style yang menarik! Source code di bawah ini hanya sebagai contoh, anda dapat berkreasi semenarik mungkin.

Source code HTML untuk halaman Login

```

<!DOCTYPE html>
<html>
  <head>
    <meta name="author" content="Tulis Nama Anda Disini – NIM">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="Latihan menggunakan CSS Sederhana">
    <link rel="stylesheet" href="login.css" />
    <title>Login</title>
  </head>
  <body>
    <div class="container">
      <div class="left-container">
        <form>
          <h2>Sign In</h2>
          <input type="email" class="form-login" placeholder="Email"/>
          <input type="password" class="form-login" placeholder="Password"/>
          <button type="button" id="sign-in">Sign In</button>
        </form>
      </div>
    </div>
  </body>
</html>

```



```
</form>
</div>
<div class="right-container">
  <h2>Welcome to login</h2>
  <p>Don't have an account?</p>
  <div style="display: flex">
    <button type="button" id="sign-up">Sign Up</button>
  </div>
</div>
</div>
</body>
</html>
```

Source code css dengan nama file login.css

```
body {
  background-color: #f8f9fd;
}
.container {
  display: flex;
  border-radius: 5px;
  box-sizing: border-box;
  box-shadow: 1px ;
  margin: 5% auto;
  width: 60vw;
}
.left-container {
  background: #fff;
  width: 50%;
  height: 50vh;
  h2 {
    color: #000;
  }
}
form {
  margin: 2rem auto;
  width: 80%;
}
h2 {
```



```
font-size: 2rem;
font-weight: 900;
margin-top: 15%;
}

p, h2 {
  color: #fff;
  text-align: center;
}

p {
  margin-bottom: 1rem;
}

button {
  font-size: 15px;
  padding: 10px 20px;
  border-radius: 50px;
  text-align: center;
  height: 48px;
}

#sign-in {
  border: 1px solid #f35588;
  color: #fff;
  background: linear-gradient(135deg, #f75959 0%, #f35587 100%);
  width: 90%;
  margin: 1rem;
}

#sign-in:hover {
  background: linear-gradient(135deg, #f28899 0%, #d14159 100%);
}

#sign-up {
  border: 1px solid #fff;
  background: transparent;
  color: #fff;
  margin: auto;
  width: 35%;
}

#sign-up:hover {
```



```
background: #fff;
color: #000;
}
.form-login {
height: 48px;
background: rgba(0, 0, 0, 0.05);
color: #000;
font-size: 16px;
border-radius: 50px;
border: 1px solid transparent;
padding: 0 20px;
width: 80%;
margin: 1rem;
}
.form-login:focus {
outline: none;
background: #e6e6e6;
}
.right-container {
background: linear-gradient(135deg, #f75959 0%, #f35587 100%);
width: 50%;
height: 50vh;
display: block;
justify-content: center;
}
```

Properti CSS

CSS memiliki banyak properti. Pada modul ini hanya 5 properti yang dijelaskan yaitu property pengaturan warna dan property untuk pengaturan letak.

a. Pengaturan Teks

Properti warna pada CSS paling banyak digunakan untuk mengatur warna latar elemen atau property background-color, untuk mengatur warna text atau color, dan untuk mengatur warna garis atau border-color. Nilai dari properti warna tersebut boleh berupa kode HEXA warna atau RGB ataupun nama warna (dalam bahasa inggris tapi pilihan terbatas).

b. Posisi elemen

c.



MODUL 3

HTML EVENTS

3.1 HTML Events

HTML menyediakan atribut berupa events atau kejadian yang terjadi dalam menggunakan web. Contohnya kejadian saat halaman web dimuat, saat user mengetik menggunakan keyboard, saat user mengklik elemen, saat user memindahkan kursor mouse dan sebagainya. Namun penggunaan attribute event hanya dapat dilakukan bersamaan dengan penggunaan Javascript. Oleh karena itu, menguasai javascript juga sangat penting dalam pengembangan interface sebuah aplikasi berbasis web.

Event HTML sangat banyak macamnya dan tidak dapat dibahas satu persatu dalam modul ini. Anda bisa belajar mandiri menggunakan link dokumentasi [w3school](https://www.w3schools.com/html/html_events.asp). Dalam modul ini, kita hanya akan membahas beberapa event yang sering digunakan dalam pengembangan web.

1. Form Events

Attribut Event	Nilai	Deskripsi
onblur	<i>script</i>	Diaktifkan saat elemen tidak difokuskan
onchange	<i>script</i>	Diaktifkan saat nilai input atau select berubah
onfocus	<i>script</i>	Diaktifkan Ketika elemen difokuskan
oninput	<i>script</i>	Diaktifkan Ketika user menginput nilai
oninvalid	<i>script</i>	Diaktifkan Ketika nilai elemen tidak benar
onreset	<i>script</i>	Diaktifkan Ketika button reset diklik
onsearch	<i>script</i>	Diaktifkan Ketika user menginput nilai pada input tipe search
onselect	<i>script</i>	Diaktifkan Ketika sebuah teks diblok
onsubmit	<i>script</i>	Diaktifkan Ketika user mengklik button submit

2. Mouse Events



Attribute	Value	Description
onclick	<i>script</i>	Diaktifkan Ketika elemen diklik sekali
ondblclick	<i>script</i>	Diaktifkan Ketika elemen diklik dua kali
onmousedown	<i>script</i>	Diaktifkan Ketika mouse ditekan pada elemen
onmouseover	<i>script</i>	Diaktifkan Ketika kursor mouse melewati element
onwheel	<i>script</i>	Diaktifkan Ketika kursor mouse ditarik ke atas atau ke bawah

3. Keyboard Events

Attribute	Value	Description
onkeydown	<i>script</i>	Diaktifkan Ketika user menekan key
onkeyup	<i>script</i>	Diaktifkan Ketika user melepas key setelah menekannya

4. Drag Events

Attribute	Value	Description
ondrag	<i>script</i>	Diaktifkan Ketika elemen ditarik
ondragend	<i>script</i>	Diaktifkan Ketika elemen telah selesai ditarik
ondragenter	<i>script</i>	Diaktifkan Ketika elemen ditarik/dipindahkan ke posisi yang sesuai dengan elemen yang ditargetkan
ondragleave	<i>script</i>	Diaktifkan Ketika elemen ditarik/dipindahkan namun meninggalkan posisi elemen yang ditargetkan
ondragover	<i>script</i>	Diaktifkan Ketika elemen ditarik/dipindahkan melewati posisi elemen yang ditargetkan
ondragstart	<i>script</i>	Diaktifkan Ketika elemen mulai ditarik



ondrop	<i>script</i>	Script to be run when dragged element is being dropped
onscroll	<i>script</i>	Script to be run when an element's scrollbar is being scrolled

5. Clipboard Events

Attribute	Value	Description
oncopy	<i>script</i>	Diaktifkan Ketika user menyalin konten sebuah elemen
oncut	<i>script</i>	Diaktifkan Ketika user meng-cut (menyalin dan menghapus) sebuah elemen
onpaste	<i>script</i>	Diaktifkan Ketika user menempel konten pada elemen

6. Media Events

Attribute	Value	Description
onabort	<i>script</i>	Script to be run on abort
oncanplay	<i>script</i>	Script to be run when a file is ready to start playing (when it has buffered enough to begin)
oncanplaythrough	<i>script</i>	Script to be run when a file can be played all the way to the end without pausing for buffering
oncuechange	<i>script</i>	Script to be run when the cue changes in a <track> element
ondurationchange	<i>script</i>	Script to be run when the length of the media changes
onemptied	<i>script</i>	Script to be run when something bad happens and the file is suddenly unavailable (like unexpectedly disconnects)
onended	<i>script</i>	Script to be run when the media has reach the end (a useful event for messages like "thanks for listening")
onerror	<i>script</i>	Script to be run when an error occurs when the file is being loaded
onloadeddata	<i>script</i>	Script to be run when media data is loaded
onloadedmetadata	<i>script</i>	Script to be run when meta data (like dimensions and duration) are loaded



onloadstart	<i>script</i>	Script to be run just as the file begins to load before anything is actually loaded
onpause	<i>script</i>	Script to be run when the media is paused either by the user or programmatically
onplay	<i>script</i>	Script to be run when the media is ready to start playing
onplaying	<i>script</i>	Script to be run when the media actually has started playing
onprogress	<i>script</i>	Script to be run when the browser is in the process of getting the media data
onratechange	<i>script</i>	Script to be run each time the playback rate changes (like when a user switches to a slow motion or fast forward mode)
onseeked	<i>script</i>	Script to be run when the seeking attribute is set to false indicating that seeking has ended
onseeking	<i>script</i>	Script to be run when the seeking attribute is set to true indicating that seeking is active
onstalled	<i>script</i>	Script to be run when the browser is unable to fetch the media data for whatever reason
onsuspend	<i>script</i>	Script to be run when fetching the media data is stopped before it is completely loaded for whatever reason
ontimeupdate	<i>script</i>	Script to be run when the playing position has changed (like when the user fast forwards to a different point in the media)
onvolumechange	<i>script</i>	Script to be run each time the volume is changed which (includes setting the volume to "mute")
onwaiting	<i>script</i>	Script to be run when the media has paused but is expected to resume (like when the media pauses to buffer more data)

3.2. JAVASCRIPT

Javascript atau sering disingkat JS merupakan salah satu Bahasa pemrograman yang sangat populer terutama dalam pengembangan web. Bahasa Javascript dipakai untuk mengembangkan web yang lebih dinamis dan lebih fungsional. Banyak aplikasi web ternama yang menggunakan Bahasa javascript seperti Google dan Facebook.

Bahasa javascript dalam pengembangan antarmuka aplikasi web sering digunakan untuk mengelolah *HTML events*. Penulisan code javascript dapat dilakukan baik pada internal maupun eksternal HTML. Code javascript hanya dapat dieksekusi jika diletakkan di dalam elemen `<script>`. Contoh dibawah ini merupakan penggunaan javascript.



```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Javascript</title>
7    </head>
8    <script>
9      alert("Selamat Datang Di Kelas Pemrograman Web");
10   </script>
11   <body>
12   </body>
13 </html>
```



DAFTAR PUSTAKA

mdn web docs. HTML: Bahasa Markup Hiperteks. <https://developer.mozilla.org/en-US/docs/Web/HTML> diakses pada 8/02/2024.