

## COURSE 1

# Belajar membuat tampilan web dengan HTML + CSS (Sisi Frontend)

### Material Details

---

---

01 Why Fullstack ? Why now?

---

02 Hello HTML

---

03 CodeSandbox + Element

---

04 How to be a Good Engineer

---

05 Attributes

---

06 Formatting & Form

---

07 HTML 5

---

08 Week 2

---

## **Modul Sekolah Fullstack Cilsy**

Hak Cipta © 2020 PT. Cilsy Fiolution Indonesia

Hak Cipta dilindungi Undang-Undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh isi buku ini dalam bentuk apapun, baik secara elektronis maupun mekanis, termasuk mecropy, merekam atau dengan sistem penyimpanan lainnya, tanpa izin tertulis dari Penulis dan Penerbit.

Penulis : Muhammad Lukman Hakim  
Editor : Muhammad Fakhri Abdillah, Iqbal Ilman Firdaus

Penerbit : **PT. Cilsy Fiolution Indonesia**  
Web Site : <https://cilsyfiolution.com> , <https://sekolahfullstack.com>

### **Sanksi Pelanggaran Pasal 113 Undang-undang Nomor 28 Tahun 2014 tentang Hak Cipta**

1. Setiap orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam pasal 9 ayat (1) huruf i untuk penggunaan secara komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan atau pidana denda paling banyak Rp100.000.000,00 (seratus juta rupiah).
2. Setiap orang yang dengan tanpa hak dan atau tanpa izin pencipta atau pemegang hak cipta melakukan pelanggaran hak ekonomi pencipta sebagaimana dimaksud dalam pasal 9 ayat (1) huruf c, huruf d, huruf f, dan atau huruf h, untuk penggunaan secara komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah)
3. Setiap orang yang dengan tanpa hak dan atau tanpa izin pencipta atau pemegang hak melakukan pelanggaran hak ekonomi pencipta sebagaimana dimaksud dalam pasal 9 ayat (1) huruf a, huruf b, huruf e, dan atau huruf g, untuk penggunaan secara komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan atau pidana denda paling banyak Rp1.000.000.000,00 (satu miliar rupiah)
4. Setiap orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan atau pidana denda paling banyak Rp4.000.000.000,00 (empat miliar rupiah)

## Daftar Isi

Daftar Isi.....	3
2. Hello CSS.....	5
2.1. Learning Outcomes.....	5
2.2. Outline Materi.....	5
2.3. Style Menggunakan CSS.....	5
2.3.1. Syntax.....	8
2.3.2. Block Selector.....	9
2.3.3. Exercise.....	9
2.3.4. Block Declaration.....	10
2.3.5. Comment.....	10
2.3.5.1. Inline.....	11
2.3.5.2. Internal.....	12
2.3.5.3. External.....	13
2.3.6. Exercise.....	14
2.4. Text Formatting.....	14
2.4.1. Mempertebal Tulisan.....	14
2.4.2. Mencetak Miring.....	15
2.4.3. Memperkecil Tulisan.....	15
2.4.4. Marking Tulisan.....	16
2.4.5. Mencoret Tulisan.....	16
2.4.6. Subscript.....	17
2.4.7. Superscript.....	17
2.4.8. Exercise.....	18
2.5. Form.....	18
2.5.1. Input Element.....	18
2.5.2. Text Fields.....	19
2.5.3. Checkboxes.....	20

2.5.4. Radio Buttons.....	20
2.5.5. Selections.....	21
2.5.6. Submit Buttons.....	21
2.5.7. Exercise.....	22
2.6. Table.....	23
2.6.1. Colspan.....	24
2.6.2. Rowspan.....	25
2.6.3. Exercise.....	26
2.7. Nested Table.....	27
2.8. Blocks.....	29
2.8.1. Block Level Element.....	29
2.8.2. Inline Level Element.....	30
2.8.3. DIV Element.....	31
2.8.4. SPAN Element.....	32
2.8.5. Exercise.....	33
2.9. Display.....	33
2.9.1. Display: None.....	33
2.9.2. Override.....	34
2.9.3. Exercise.....	34
2.10. Position.....	35
2.10.1. Static.....	35
2.10.2. Relative.....	36
2.10.3. Fixed.....	37
2.10.4. Absolute.....	38
2.11. HTML Layout.....	41
2.11.1. Exercise.....	43
2.12. Float.....	43
2.13. Studi Kasus Membuat Web Sederhana.....	44
2.14. Exercise.....	49

## 2.

## Hello CSS

### 2.1. Learning Outcomes

Setelah selesai mempelajari bab ini, peserta mampu :

1. Mengetahui apa itu CSS.
2. Memahami penggunaan CSS.
3. Dapat menggunakan CSS.
4. Memiliki skill dalam menggunakan CSS.

### 2.2. Outline Materi

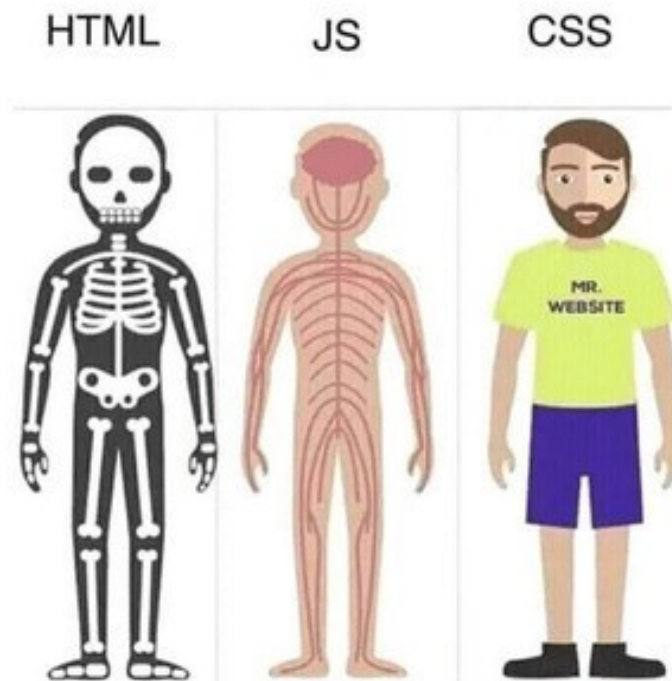
1. CSS
2. Text Formatting
3. Form
4. Table

### 2.3. Style Menggunakan CSS

*Cascading style sheet* (CSS) merupakan aturan untuk mengatur beberapa komponen pada halaman web sehingga lebih terstruktur dan seragam. CSS bukanlah sebuah bahasa pemrograman.

Pada umumnya CSS digunakan untuk memformat tampilan halaman web yang dibangun menggunakan bahasa HTML dan XHTML. CSS dapat mengendalikan ukuran, warna, posisi serta sifat dari sebuah komponen pada halaman web.

Apabila masih bingung, berikut adalah analogi sederhana dari HTML dan CSS.

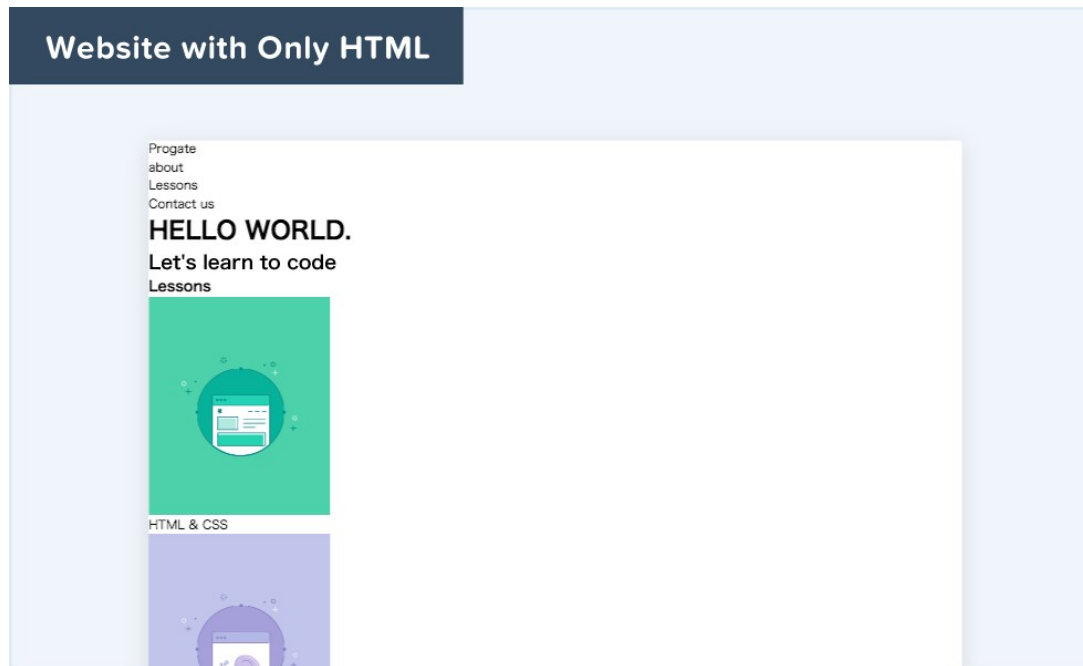


*Analogi Sederhana Perbedaan HTML, JS, dan CSS.*

Img source: <https://medium.com/@vincentdeli/basic-flexbox-with-example-97230ebf8d01>

Jika dianalogikan sebagai sistem tubuh seorang manusia, HTML merupakan kerangka yang menyusun bentuk suatu tubuh, JS merupakan sistem yang berjalan dalam tubuh tersebut sehingga darah dapat mengalir ke tempat yang semestinya, sementara CSS adalah tampilan luar dari fisik manusia, baik itu warna kulit, warna rambut, baju, celana, dan komponen lainnya.

Untuk mudahnya, lihatlah ilustrasi berikut yang merupakan perbedaan web yang menggunakan CSS dan tidak.



*Web Tanpa CSS*



*Website menggunakan HTML+CSS. Image source: [progate.com](https://progate.com)*

Penggunaan CSS untuk mengatur halaman web direkomendasikan oleh World Wide Web Consortium pada tahun 1996. Terdapat tiga versi CSS saat ini, yaitu:

- a. CSS Versi 1 dikembangkan dengan fokus memformat dokumen HTML agar lebih rapih saat ditampilkan.
- b. CSS Versi 2 dikembangkan untuk kebutuhan terharap format dokumen agar bisa di tampilkan dan di cetak (format tidak berubah saat dicetak menggunakan mesin printer).
- c. CSS Versi 3 dikembangkan untuk mendukung posisi konten, dapat di unduh, dapat memanipulasi bentuk font, mengubah desain tabel dan dukungan untuk pencetakan.



### 2.3.1. Syntax

Syntax CSS terdiri dari blok selector dan blok declaration, dimana declaration akan dibungkus menggunakan kurung kurawal.

```
h1 {color:blue; font-size:15px;}
```

'h1' merupakan selector

'color:blue' merupakan declaration



### 2.3.2. Block Selector

Blok selector digunakan untuk memilih target atau bagian HTML mana yang akan diatur tampilannya menggunakan CSS.

```
h1 {color:blue; font-size:15px;}
```

Dengan syntax diatas dapat diartikan yang akan diubah formatnya sesuai *declaration* yang sudah ditulis adalah element h1 karena *selector* yang digunakan adalah *selector element* (termasuk pada simple selector) yang merujuk pada element h1.

Selector dibagi menjadi lima kategori:

1. Simple Selectors (memilih berdasarkan element, id atau class)
2. Combinator Selectors (memilih berdasarkan hubungan antara element)
3. Pseudo-class Selectors (memilih element berdasarkan kondisi tertentu)
4. Pseudo-elements Selectors (memilih dan mengubah sebagian element)
5. Attribute Selectors (memilih element berdasarkan attribute atau nilai attribute tertentu)

### 2.3.3. Exercise

#### A. Diskusi!

Cari tau apa yang disebut Combinator Selector dan bagaimana cara kerjanya! Setelah kamu menemukannya diskusi lah dengan mentor mu.

#### B. Latihan!

Terapkan combinator selector!

### 2.3.4. Block Declaration

Blok Declaration terdiri dari beberapa declaration. Declaration terbangun dari dua bagian, yang pertama adalah property dan yang kedua adalah value. Property merujuk pada aturan apa yang hendak diterapkan dan value merupakan nilai dari aturannya.

```
{color:blue}
```

Pada contoh diatas dapat dilihat property yang digunakan adalah 'color' yang tujuannya menerapkan aturan warna dan value nya berupa 'blue' yang berarti semua warna akan diganti menjadi biru.

Blok declaration akan diawali dengan kurung kurawal dan ditutup dengan kurung kurawal. Setelah penulisan property diikuti dengan penulisan simbol (:) titik dua lalu value yang akan diterapkan. Didalam blok declaration ini dapat memuat lebih dari satu declaration dengan penggunaan pemisah berupa (;) semicolon disetiap declaration nya.

```
{color:blue; font-size:15px}
```

### 2.3.5. Comment

Comment digunakan untuk menjelaskan sebuah kode atau blok kode, yang akan membantu dalam pengembangan kode pada tahap berikutnya. Baris comment tidak akan dibaca oleh web browser dan tidak akan menimbulkan error.

Penulisan comment dapat dilakukan diantara simbol /\* dan \*/, contohnya:

```
h1 {  
  color: blue;  
  /* font akan dibesarkan dengan menggunakan font-size */  
  font-size: 15px;  
}
```

Tulisan "font akan dibesarkan dengan menggunakan font-size" tidak akan dibaca oleh web browser. Adapun penulisan comment yang terdiri dari beberapa baris akan berbentuk seperti berikut:

```
h1 {  
    color: blue;  
    /*  
        font akan dibesarkan dengan menggunakan font-size  
        line yang berada didalam tanda comment tidak akan  
ditampilkan  
        tulis comment mu disini!  
    */  
    font-size: 15px;  
}
```

## Penggunaan

Ketika browser membaca terdapat style sheet, dokumen HTML yang di muat akan diformat sesuai aturan yang ada pada style sheet.

Terdapat tiga cara dalam menggunakan CSS pada HTML:

1. Inline
2. Internal
3. External

### 2.3.5.1. *Inline*

Inline CSS digunakan untuk mengubah format style pada satu element. Menggunakan attribute 'style' pada element lalu masukan syntax CSS yang akan diterapkan. Style dapat menampung berbagai macam property CSS.

```
<!DOCTYPE html>
<html>
  <body>
    <h1 style="color:blue;text-align:center">Ini Heading</h1>
    <p style="color:red;">Ini paragraph</p>
  </body>
</html>
```

### 2.3.5.2. *Internal*

Internal CSS dapat digunakan pada satu halaman HTML yang disimpan pada element `<style>` dan ditempatkan pada element `<head>`.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      h1 {
        color: blue;
        text-align: center;
      }

      p {
        color: red;
      }
    </style>
  </head>

  <body>
    <h1>Ini Heading</h1>
```

```
<p>Ini paragraph</p>
</body>
</html>
```

Gunakan selector untuk memilih bagian atau element mana yang akan mendapatkan style.

### 2.3.5.3. *External*

Dengan menggunakan external style dapat mengubah seluruh aturan style pada web hanya dengan mendeklarasikan aturan style pada satu dokumen CSS. Setiap halaman HTML harus mencantumkan referensi pada external CSS didalam element <link> pada <head>.

```
index.html

<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="style.css">
  </head>

  <body>
    <h1>Ini Heading</h1>
    <p>Ini paragraph</p>
  </body>
</html>
```

```
style.css

h1 {
```

```
    color: blue;
    text-align: center;
}

p {
    color: red;
}
```

### 2.3.6. Exercise

Perbaharuilah cerita yang sebelumnya dibuat dengan ketentuan sebagai berikut:

1. Tidak lagi menggunakan attribute style!
2. Gunakan CSS external untuk menyimpan style.

## 2.4. Text Formatting

Pada bagian sebelumnya telah dipelajari bagaimana menggunakan attribute style untuk mengubah tampilan. Namun HTML menyediakan element yang dapat langsung digunakan untuk memberikan styling pada text seperti mempertebal, memberi garis bawah dan mencetak tulisan menjadi miring.

### 2.4.1. Mempertebal Tulisan

Untuk mempertebal tulisan dapat menggunakan element `<b>` atau `<strong>`. Element `<strong>` memberikan penekanan bahwa text tersebut sangat penting.

```
<p><b>Tulisan ini tebal</b></p>
```

```
<p><strong>Tulisan ini juga tebal tapi lebih penting</strong></p>
```



Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

    <p><b>Tulisan ini tebal</b></p>

    <p><strong>Tulisan ini juga tebal tapi lebih
    penting</strong></p>
  </body>
</html>
```

Tulisan ini tebal

Tulisan ini juga tebal tapi lebih penting

### 2.4.2. Mencetak Miring

Untuk mencetak tulisan dengan style miring dapat menggunakan element `<i>` dan `<em>`. Element `<em>` memberikan penekanan bahwa text tersebut penting.

```
<p><i>Tulisan ini miring</i></p>

<p><em>Tulisan ini miring</em></p>
```

Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

    <p><i>Tulisan ini miring</i></p>

    <p><em>Tulisan ini miring</em></p>
  </body>
</html>
```

*Tulisan ini miring*

*Tulisan ini miring*

### 2.4.3. Memperkecil Tulisan

Dengan menggunakan element `<small>` tulisan dapat menjadi kecil.

```
<p>Tulisan ini <small>kecil</small></p>
```

Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

<p>Tulisan ini <small>kecil</small></p>

</body>
</html>
```

Tulisan ini kecil

#### 2.4.4. Marking Tulisan

Menggunakan element <mark> untuk menampilkan tulisan dengan efek marking.

```
<p>Tulisan ini memiliki <mark>marking</mark></p>
```

Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

<p>Tulisan ini memiliki <mark>marking</mark></p>

</body>
</html>
```

Tulisan ini memiliki marking

#### 2.4.5. Mencoret Tulisan

Dengan menggunakan element <del> tulisan yang ditampilkan dapat memiliki kesan "dicoret".

```
<p> saya <del>tidak</del> jomblo</p>
```

Hasil dari code diatas adalah sebagai berikut:



```
<!DOCTYPE html>
<html>
  <body>

  <p> saya <del>tidak</del> jomblo</p>

</body>
</html>
```

saya tidak jomblo

## 2.4.6. Subscript

Membuat tulisan menjadi subscript menggunakan element `<sub>`.

```
<p>Tulisan ini memiliki <sub>subscripted text</sub></p>
```

Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

  <p>Tulisan ini memiliki <sub>subscripted text</sub></p>

</body>
</html>
```

Tulisan ini memiliki subscripted text

## 2.4.7. Superscript

Membuat tulisan menjadi superscript (penggunaan dalam penulisan pangkat) menggunakan element `<sup>`.

```
<p>Tulisan ini memiliki <sup>superscripted text</sup>
```

Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

  <p>Tulisan ini memiliki <sup>superscripted text</sup>

</body>
</html>
```

Tulisan ini memiliki superscripted text

### 2.4.8. Exercise

Buatlah teks dibawah ini pada halaman HTML:

Rumus Pytagoras

$$a^2 + b^2 = c^2$$

dimana **a** adalah *alas*

dan **b** adalah *tinggi*

sedangkan **c** adalah *bidang datar miring*

## 2.5. Form

Pada bagian form akan dipelajari bagaimana cara nya mengambil masukan yang dilakukan oleh pengguna. Dengan menggunakan element `<form>` sebagai bungkusnya dan beberapa element pendukung lainnya dengan spesifikasi terkait dengan jenis masukan yang pengguna berikan seperti text fields, checkboxes, radio buttons dan submit buttons.

```
<form>
```

```
form elements
```

```
</form>
```

### 2.5.1. Input Element

Element `<input>` adalah element terpenting pada element `<form>`. Element `<input>` dapat ditampilkan dengan beragam bentuk, contohnya:

Tipe	Keterangan
<code>&lt;input type="text"&gt;</code>	Mendefinisikan form input text dengan bentuk single line.

<code>&lt;input type="radio"&gt;</code>	Mendefinisikan form input radio yang hanya dapat dipilih satu dari sekian banyak pilihan.
<code>&lt;input type="submit"&gt;</code>	Mendefinisikan submit button yaitu tombol yang digunakan untuk mengirim masukan pengguna.

### 2.5.2. Text Fields

Membuat sebuah form untuk masukan text dengan bentuk single line menggunakan attribute `type="text"` pada element `<input>` yang tentunya dibungkus dengan element `<form>`.

```
<form>
  Nama : <br>
  <input type="text" name="nama"> <br>

  Umur : <br>
  <input type="text" name="umur">
</form>
```

Hasil dari code diatas adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
  <body>

    <form>
      Nama : <br>
      <input type="text" name="nama"> <br>

      Umur : <br>
      <input type="text" name="umur">
    </form>

  </body>
</html>
```

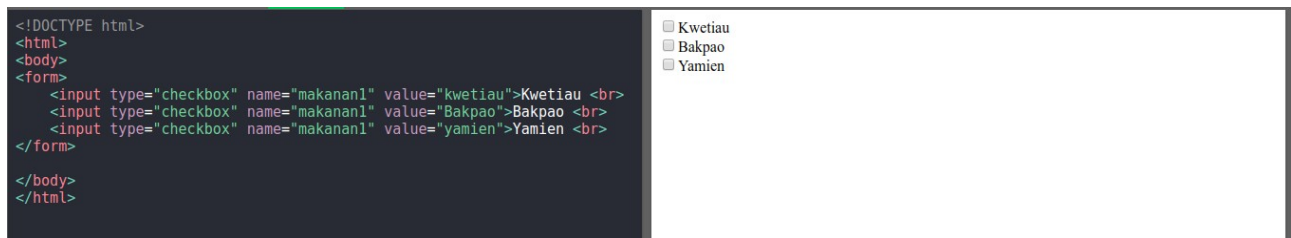
Nama :  
  
 Umur :

### 2.5.3. Checkboxes

Membuat sebuah pilihan menggunakan checkbox yang dapat memilih beberapa pilihan. Menggunakan attribute `type="checkbox"` pada element `<input>`.

```
<form>
  <input type="checkbox" name="makanan1" value="kwetiau">Kwetiau
<br>
  <input type="checkbox" name="makanan1" value="Bakpao">Bakpao
<br>
  <input type="checkbox" name="makanan1" value="yamien">Yamien
<br>
</form>
```

Hasil dari code diatas adalah sebagai berikut:



```
<!DOCTYPE html>
<html>
<body>
<form>
  <input type="checkbox" name="makanan1" value="kwetiau">Kwetiau <br>
  <input type="checkbox" name="makanan1" value="Bakpao">Bakpao <br>
  <input type="checkbox" name="makanan1" value="yamien">Yamien <br>
</form>
</body>
</html>
```

☐ Kwetiau  
☐ Bakpao  
☐ Yamien

### 2.5.4. Radio Buttons

Sama seperti checkboxes hanya radio button hanya memperbolehkan untuk memilih satu pilihan saja.

```
<form>
  <input type="radio" name="gender" value ="male" checked> Male
<br>
  <input type="radio" name="gender" value ="female"> Female <br>
</form>
```

Hasil dari code diatas adalah sebagai berikut:

```

<!DOCTYPE html>
<html>
<body>

<form action="/action_page.php">
  <input type="radio" name="gender" value="male" checked> Male<br>
  <input type="radio" name="gender" value="female"> Female<br>
</form>

</body>
</html>

```

☒ Male  
☐ Female

### 2.5.5. Selections

Element `<select>` digunakan untuk membuat drop-down list. Menggunakan element `<option>` untuk menyimpan pilihan.

```

<select>
  <option value="lada">Lada</option>
  <option value="volvo">Volvo</option>
  <option value="audi">Audi</option>
</select>

```

Hasil dari code diatas adalah sebagai berikut:

```

<!DOCTYPE html>
<html>
<body>

<select>
  <option value="lada">Lada</option>
  <option value="volvo">Volvo</option>
  <option value="audi">Audi</option>
</select>

</body>
</html>

```

Lada ▼

### 2.5.6. Submit Buttons

`<input type="submit">` mendefinisikan tombol untuk "submitting" data dari form ke handlernya. Form handler biasanya berupa server yang menerima request dan memproses data masukan. Form handler didefinisikan menggunakan attribute action yang disisipkan

pada element `<form>`. Setelah tombol submit di klik, maka data pada form akan dikirimkan pada alamat yang di definisikan pada attribute action.

```
<form action="/action.php">
  Nama : <br>
  <input type="text" name="nama" value="Hatsune Miku"> <br>

  Umur : <br>
  <input type="text" name="umur" value="17"> <br>

  <input type="submit" value="submit">
</form>
```

Hasil dari code diatas adalah sebagai berikut:



```
<!DOCTYPE html>
<html>
  <body>
    <form action="/action.php">
      Nama : <br>
      <input type="text" name="nama" value="Hatsune Miku">
      <br>
      Umur : <br>
      <input type="text" name="umur" value="17"> <br>
      <input type="submit" value="submit">
    </form>
  </body>
</html>
```

## 2.5.7. Exercise

Buatlah sebuah form yang digunakan untuk mengisi biodata! Item yang wajib ada adalah:

1. Nama (boleh langsung nama lengkap atau dibuat terpisah).
2. Tanggal Lahir.
3. Kota Asal.
4. Jenis Kelamin (menggunakan radio button).



5. Pilihan pendidikan terakhir (SD, SMP, SMA/SMK, D3, S1) menggunakan checkbox.
6. Pekerjaan (buat beberapa pilihan) menggunakan selection.

## 2.6. Table

Untuk membuat table pada HTML digunakan element `<table>` dengan bantuan beberapa element untuk detail tabel nya. Element `<tr>` digunakan untuk mendefinisikan table row. Element `<th>` digunakan untuk table heading atau penempatan judul dan element `<td>` digunakan untuk table data. Berikut contoh struktur kode tabel:

```
<table style="width:100%">
  <tr>
    <th>Nama Depan</th>
    <th>Nama Belakang</th>
    <th>Umur</th>
  </tr>
  <tr>
    <td>Lukman</td>
    <td>Hakim</td>
    <td>100</td>
  </tr>
  <tr>
    <td>Randy</td>
    <td>Jackson</td>
    <td>56</td>
  </tr>
</table>
```

Hasil dari code diatas adalah sebagai berikut:



```
<!DOCTYPE html>
<html>
<body>

<table style="width:100%">
  <tr>
    <th>Nama Depan</th>
    <th>Nama Belakang</th>
    <th>Umur</th>
  </tr>
  <tr>
    <td>Lukman</td>
    <td>Hakim</td>
    <td>100</td>
  </tr>
  <tr>
    <td>Randy</td>
    <td>Jackson</td>
    <td>56</td>
  </tr>
</table>
</body>
</html>
```

Nama Depan	Nama Belakang	Umur
Lukman	Hakim	100
Randy	Jackson	56

### 2.6.1. Colspan

Menggabungkan beberapa kolom menjadi satu.

```
<table style="width:100%">
  <tr>
    <th>Nama</th>
    <th colspan="2">Telepon</th>
  </tr>
  <tr>
    <td>Lukman Hakim</td>
    <td>(022) 6633563</td>
    <td>081320176234</td>
  </tr>
</table>
```

Hasil dari code diatas adalah sebagai berikut:



```
<!DOCTYPE html>
<html>
  <body>

  <table style="width:100%">
    <tr>
      <th>Nama</th>
      <th colspan="2">Telepon</th>
    </tr>
    <tr>
      <td>Lukman Hakim</td>
      <td>(022) 6633563</td>
      <td>081320176234</td>
    </tr>
  </table>

</body>
</html>
```

Nama		Telepon
Lukman Hakim	(022) 6633563	081320176234

## 2.6.2. Rowspan

Menggabungkan beberapa baris menjadi satu.

```
<table style="width:100%">
  <tr>
    <th>Nama:</th>
    <td>Lukman Hakim</td>
  </tr>
  <tr>
    <th rowspan="2">Telepon:</th>
    <td>(022) 6633563</td>
  </tr>
  <tr>
    <td>081320176234</td>
  </tr>
</table>
```

Hasil dari code diatas adalah sebagai berikut:

<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;table style="width:100%"&gt;   &lt;tr&gt;     &lt;th&gt;Nama:&lt;/th&gt;     &lt;td&gt;Lukman Hakim&lt;/td&gt;   &lt;/tr&gt;   &lt;tr&gt;     &lt;th rowspan="2"&gt;Telepon:&lt;/th&gt;     &lt;td&gt;(022) 6633563&lt;/td&gt;   &lt;/tr&gt;   &lt;tr&gt;     &lt;td&gt;081320176234&lt;/td&gt;   &lt;/tr&gt; &lt;/table&gt;  &lt;/body&gt; &lt;/html&gt; </pre>	<b>Nama:</b> Lukman Hakim <b>Telepon:</b> (022) 6633563 081320176234
---	--

## 2.6.3. Exercise

### A. Diskusi!

Tambahkan border pada sebuah tabel dengan menggunakan CSS table (border: 1px solid black) dan perhatikan apa yang terjadi.

Jika terdapat dua garis pada setiap cell nya, bagaimana cara menghilangkan dua garis tersebut agar menjadi satu garis?

### B. Latihan!

1. Buatlah sebuah tabel dengan 3 row dan 3 kolom.
2. Buatlah sebuah tabel yang memiliki 2 row dimana pada row pertama memiliki 5 colom sedangkan pada row kedua memiliki 6 kolom.
3. Buatlah sebuah tabel seperti contoh berikut


4. Buatlah sebuah tabel yang berisi daftar menu dari sebuah restoran dengan bentuk dan warna semenarik mungkin!

## 2.7. Nested Table

Membuat tabel yang berada di dalam sebuah tabel disebut nested table atau tabel bersarang. Teknik ini cukup “tricky” dimana element `<table>` akan dimasukan kedalam element `<td>`. Contoh penggunaan nested table:

Nested pada kolom

```
<table>
  <tr>
    <td>Cell pertama pada tabel</td>
    <td>
      <table>
        <tr><td>nested table</td></tr>
        <tr><td>nested table</td></tr>
      </table>
    </td>
  </tr>
</table>
```

Hasilnya akan seperti ini:

Cell pertama pada tabel	nested table
	nested table

Untuk membuat nested table pada row berikutnya kita tidak menyimpan tabel baru pada element `<tr>`. Tabel tetap disimpan pada element `<td>` hanya saja dibuatkan `<tr>` baru yang kode nya akan seperti ini:

Nested pada row

```
<table>
  <tr>
    <td>Cell pertama pada tabel</td>
  </tr>

  <tr>
    <td>
      <table>
        <tr><td>nested table</td></tr>
        <tr><td>nested table</td></tr>
      </table>
    </td>
  </tr>
</table>
```

Hasilnya akan seperti berikut:

Cell pertama pada tabel		
nested table		
nested table		

#### Latihan!

1. Buatlah sebuah tabel dengan 2 row dan 3 column dimana pada row 1 column 3 diberi nested table yang memiliki 1 row dan 3 column. Untuk row ke-2 pada tabel utama dijadikan 1 column.
2. Buatlah sebuah tabel dengan 3 row dan 2 column dimana pada row 1 dibuat menjadi 1 column, pada row 2 diberi nested tabel yang memiliki 1 row dan 3



column sedangkan row ke-3 tidak diubah apapun.

## 2.8. Blocks

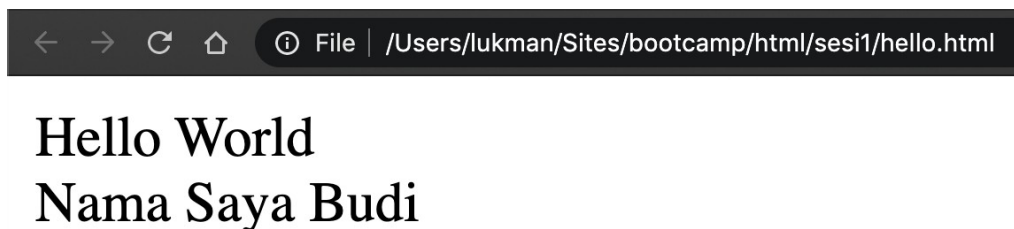
Setiap element HTML memiliki nilai default display berdasarkan apa tipe element tersebut. Ada dua nilai default display yaitu: block dan inline.

### 2.8.1. Block Level Element

Sebuah block level element selalu diawali dengan baris baru dan mengambil penuh atas lebar yang tersedia.

```
<div>Hello Wolrd</div>
<div>Nama Saya Budi</div>
```

Hasil dari kode diatas adalah:



Satu element `<div>` akan mengambil lebar yang tersedia, dengan demikian element `<div>` yang kedua berada di bawah element `<div>` yang pertama.

Block level element yang ada pada HTML adalah:

<code>&lt;address&gt;</code>	<code>&lt;aside&gt;</code>	<code>&lt;canvas&gt;</code>	<code>&lt;div&gt;</code>	<code>&lt;nav&gt;</code>
<code>&lt;figure&gt;</code>	<code>&lt;form&gt;</code>	<code>&lt;header&gt;</code>	<code>&lt;li&gt;</code>	<code>&lt;fieldset&gt;</code>
<code>&lt;p&gt;</code>	<code>&lt;section&gt;</code>	<code>&lt;tfoot&gt;</code>	<code>&lt;video&gt;</code>	<code>&lt;noscript&gt;</code>

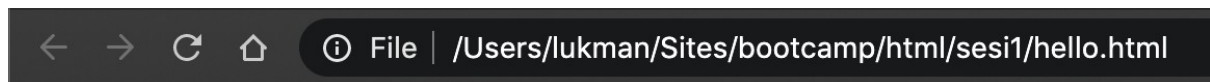


<code>&lt;article&gt;</code>	<code>&lt;blockquote&gt;</code>	<code>&lt;dd&gt;</code>	<code>&lt;dl&gt;</code>	<code>&lt;figcaption&gt;</code>
<code>&lt;footer&gt;</code>	<code>&lt;h1&gt;-&lt;h6&gt;</code>	<code>&lt;hr&gt;</code>	<code>&lt;main&gt;</code>	<code>&lt;ol&gt;</code>
<code>&lt;pre&gt;</code>	<code>&lt;table&gt;</code>	<code>&lt;ul&gt;</code>	<code>&lt;dt&gt;</code>	

## 2.8.2. Inline Level Element

Berbeda dengan block level element, Inline level element tidak mengambil seluruh lebar yang ada namun hanya mengambil lebar secukupnya. Jadi ketika ada dua element inline maka konten berikutnya tidak mengambil baris baru tapi melanjutkan.

```
<span>Hello World</span>
<span> Nama Saya Budi</span>
```



# Hello World Nama Saya Budi

Selama lebar mencukupi, element berikutnya akan berada pada posisi inline. Beberapa element yang memiliki sifat inline adalah:

<code>&lt;a&gt;</code>	<code>&lt;select&gt;</code>	<code>&lt;input&gt;</code>	<code>&lt;big&gt;</code>	<code>&lt;sup&gt;</code>
<code>&lt;em&gt;</code>	<code>&lt;acronym&gt;</code>	<code>&lt;span&gt;</code>	<code>&lt;label&gt;</code>	<code>&lt;button&gt;</code>
<code>&lt;script&gt;</code>	<code>&lt;img&gt;</code>	<code>&lt;bdo&gt;</code>	<code>&lt;sub&gt;</code>	<code>&lt;object&gt;</code>
<code>&lt;abbr&gt;</code>	<code>&lt;email&gt;</code>	<code>&lt;kbd&gt;</code>	<code>&lt;br&gt;</code>	<code>&lt;textarea&gt;</code>
<code>&lt;i&gt;</code>	<code>&lt;b&gt;</code>	<code>&lt;strong&gt;</code>	<code>&lt;map&gt;</code>	<code>&lt;cite&gt;</code>



<code>&lt;output&gt;</code>	<code>&lt;code&gt;</code>	<code>&lt;tt&gt;</code>	<code>&lt;samp&gt;</code>
<code>&lt;time&gt;</code>	<code>&lt;q&gt;</code>	<code>&lt;dfn&gt;</code>	<code>&lt;var&gt;</code>

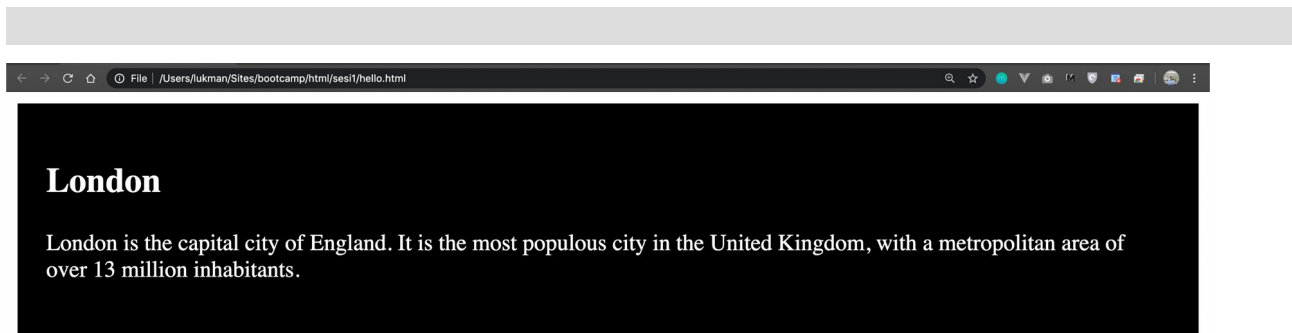
### 2.8.3. DIV Element

Element `<div>` biasa digunakan sebagai kontainer untuk element HTML lainnya. Dalam penggunaannya element `<div>` tidak memerlukan attribute saat digunakan namun biasanya element `<div>` disisipkan attribute `style`, `class` dan `id`. Ketika digunakan bersama CSS, element `div` dapat digunakan untuk memberikan style pada satu block (berpengaruh pada semua element yang ada didalam `<div>`).

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello</title>
  </head>

  <body>
    <div style="background-
color:black;color:white;padding:20px;">
      <h2>London</h2>
      <p>
        London is the capital city of England.
        It is the most populous city in the United
Kingdom,
        with a metropolitan area of over 13 million
inhabitants.
      </p>
    </div>
  </body>
</html>
```



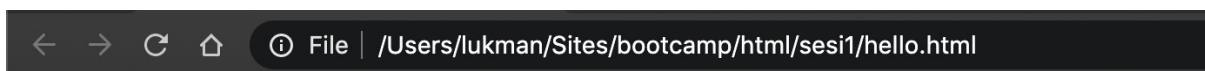


## 2.8.4. SPAN Element

Element `<span>` biasa digunakan untuk container sebuah text. Sama seperti `<div>`, element `<span>` tidak memerlukan attribute untuk digunakan tetapi biasanya disisipkan attribute style, class dan id. Ketika menggunakan CSS, element ini dapat digunakan untuk memberikan stye yang berbeda pada setiap elementnya.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello</title>
  </head>

  <body>
    <h1>My <span style="color:red">Important</span>
Heading</h1>
  </body>
</html>
```



# My **Important** Heading



## 2.8.5. Exercise

### 1. Diskusi

Lakukan percobaan untuk nested div sedalam 3 level! jika setiap div diberi warna yang berbeda perhatikan apa yang terjadi.

Lakukan percobaan jika setelah element div diberi element span, apakah output akan menjadi block level ataukah inline level. (coba sebaliknya)

Cari informasi apakah element block seperti `<div>` dapat digunakan secara inline? Lakukan percobaan.

### 2. Latihan!

Buatlah 3 buah element `<div>` dan susun menjadi inline dengan lebar masing-masing div 30%.

## 2.9. Display

Property display menspesifikasikan bagaimana sebuah element ditampilkan. Property ini memegang peran penting dalam layouting. Setiap element HTML memiliki nilai default untuk menampilkan konten yang ada didalamnya tergantung pada jenis dari element tersebut. Nilai default dari sebuah element yang umum adalah block dan inline.

### 2.9.1. Display: None

Biasanya digunakan untuk menampilkan atau menyembunyikan element tanpa menghapusnya.

Lakukan percobaan berikut!

```
<h1>Ini akan terlihat!</h1>
<h1 style="display: none;">Ini tidak akan terlihat</h1>
```



### 2.9.2. Override

Sebelumnya kita sudah pernah belajar mengenai display blok dan element, kita juga sudah tau element mana saja yang memiliki default display block atau inline. Pada section ini kita akan melakukan override terhadap nilai default sebuah display menggunakan property display.

Seperti yang kita ketahui, element `<li>` memiliki default display dengan nilai block namun dengan kode berikut:

```
li {  
  display: inline;  
}
```

Maka browser akan menampilkan menu secara horizontal.

Sama halnya dengan element `<span>` yang memiliki default display inline akan berubah menjadi block ketika diberi property display block:

```
span {  
  display: block;  
}
```

### 2.9.3. Exercise

1. Lakukan implementasi terhadap element `<li>` menjadi inline dan `<span>` menjadi block.
2. Lakukan minimal 3 element dengan default display block menjadi inline dan element dengan default display inline menjadi block.

## 2.10. Position

Property position menspesifikasikan tipe dari metode posisi yang digunakan pada sebuah element HTML. Terdapat lima nilai posisi static, relative, fixed, absolute dan sticky.

### 2.10.1. Static

Element HTML secara default diposisikan sebagai static. Posisi static ini tidak terpengaruh dengan property top, bottom, right dan left. Element yang memiliki posisi static tidak lah ditempatkan pada posisi yang special, posisinya akan berada pada kondisi yang normal berdasarkan urutan kode pada website.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      div.static {
        position: static;
        border: 3px solid #73AD21;
      }
    </style>
  </head>

  <body>
    <h2>position: static;</h2>

    <p>Element yang memiliki posisi static akan diposisikan sesuai
dengan posisi element tersebut pada
kode</p>

    <div class="static">
      Element position: static;
```



```
</div>
</body>
</html>
```

### 2.10.2. Relative

Element yang memiliki posisi relative akan diposisikan relative pada posisi normal. Melakukan pengaturan pada top, bottom, right dan left element akan secara relative menyesuaikan posisinya. Konten lainnya (selain element yang ditambahkan position: relative;) tidak akan menyesuaikan dan mengisi kekosongan yang ada karena penyesuaian element yang memiliki property position: relative;.

Praktekan kode berikut dan ubahlah property top, bottom, right dan left nya serta ganti nilainya agar dapat melihat reaksi apa yang terjadi.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      div.relative {
        position: relative;
        left: 30px;
        border: 3px solid #73AD21;      }
    </style>
  </head>

  <body>
    <h2>position: relative;</h2>
```



<p>Element yang memiliki posisi relative akan diposisikan sesuai dengan posisi normalnya.</p>

```
<div class="relative">
  Element position: relative;
</div>
</body>
</html>
```

### 2.10.3. Fixed

Element yang menggunakan position: fixed; akan diposisikan relative pada viewport, yang artinya posisinya akan tetap sama walaupun halaman di scroll. Digunakan property top, bottom, left dan right untuk mengatur posisi element.

Praktekan kode berikut dan ubahlah property top, bottom, right dan left nya serta ganti nilainya agar dapat melihat reaksi apa yang terjadi.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      div.relative {
        position: fixed;
        bottom: 0;
        right: 0;
        width: 300px;
        border: 3px solid #73AD21;
      }
    </style>
  </head>
```

```
<body>
  <h2>position: fixed;</h2>

  <p>Element yang memiliki posisi fixed akan diposisikan
relative pada viewport.</p>

  <div class="fixed">
    Element position: fixed;
  </div>
</body>
</html>
```

#### 2.10.4. Absolute

Element dengan posisi absolute akan relative mendekati posisi induknya. Ketika absolute digunakan tanpa ada posisi induknya, maka element `<body>` akan dianggap sebagai induknya.

Praktekan kode berikut dan ubahlah property top, bottom, right dan left nya serta ganti nilainya agar dapat melihat reaksi apa yang terjadi.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      div.relative {
        position: relative;
        width: 400px;
        height: 200px;
        border: 3px solid #73AD21;
```

```
}

div.absolute {
  position: absolute;
  top: 80px;
  right: 0;
  width: 200px;
  height: 100px;
  border: 3px solid #73AD21;
}
</style>
</head>

<body>
  <h2>position: absolute;</h2>

  <p>Element yang memiliki posisi absolute akan diposisikan
  relative pada induknya.</p>

  <div class="relative">
    Element position: relative;
    <div class="absolute">Element position: absolute;</div>
  </div>
</body>
</html>
```

### 2.11.1. Sticky

Element yang memiliki property sticky akan diposisikan berdasarkan user scroll position. Sticky position dapat menyesuaikan kondisi antara relative dan fixed tergantung dari scrolling position.

Praktekan kode berikut dan ubahlah property top, bottom, right dan left nya serta ganti nilainya agar dapat melihat reaksi apa yang terjadi.

```
<!DOCTYPE html>
<html>
<head>
<style>
div.sticky {
  position: -webkit-sticky;
  position: sticky;
  top: 0;
  padding: 5px;
  background-color: #cae8ca;
  border: 2px solid #4CAF50;
}
</style>
</head>
<body>

<p>Try to <b>scroll</b> inside this frame to understand how sticky
positioning works.</p>

<p>Note: IE/Edge 15 and earlier versions do not support sticky
position.</p>

<div class="sticky">I am sticky!</div>

<div style="padding-bottom:2000px">
  <p>In this example, the sticky element sticks to the top of the
page (top: 0), when you reach its scroll position.</p>
  <p>Scroll back up to remove the stickyness.</p>
  <p>Some text to enable scrolling.. Lorem ipsum dolor sit amet,
illum definitiones no quo, maluisset concludaturque et eum, altera
```





```
fabulas ut quo. Atqui causae gloriatur ius te, id agam omnis
evertitur eum. Affert laboramus repudiandae nec et. Inciderint
efficiantur his ad. Eum no molestiae voluptatibus.</p>
```

```
<p>Some text to enable scrolling.. Lorem ipsum dolor sit amet,
illum definitiones no quo, maluisset concludaturque et eum, altera
fabulas ut quo. Atqui causae gloriatur ius te, id agam omnis
evertitur eum. Affert laboramus repudiandae nec et. Inciderint
efficiantur his ad. Eum no molestiae voluptatibus.</p>
```

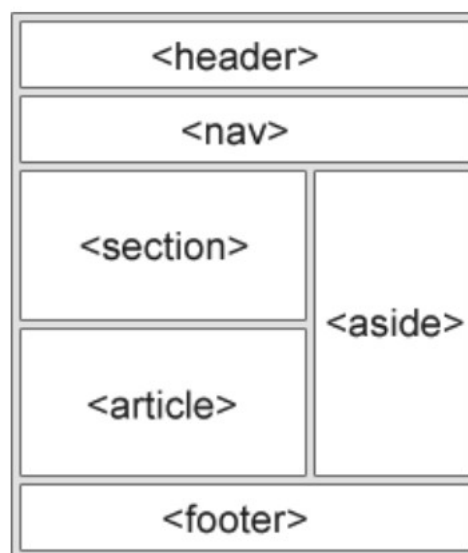
```
</div>
```

```
</body>
```

```
</html>
```

## 2.11. HTML Layout

Terkadang website menampilkan konten dengan menggunakan beberapa kolom seperti koran atau majalah. Pada HTML terdapat element khusus yang dapat digunakan untuk mendefinisikan beberapa bagian dari sebuah website.



1. `<header>` digunakan untuk mendefinisikan section untuk header dokumen.
2. `<nav>` digunakan untuk menyimpan link untuk navigasi.
3. `<section>` digunakan untuk menyimpan section.

4. `<article>` digunakan untuk menyimpan artikel.
5. `<aside>` digunakan untuk menyimpan konten tambahan atau biasanya sidebar.
6. `<footer>` digunakan sebagai penutup sebuah website.

Dalam membuat layout menggunakan HTML terdapat beberapa teknik, setiap teknik memiliki pro dan kontra. Berikut teknik yang dapat digunakan :

- HTML Tabel (tidak direkomendasikan)

Element `<table>` tidak di desain untuk digunakan sebagai layouting tool. Tujuan dari element `<table>` adalah untuk menampilkan data tabular. Jadi penggunaan tabel untuk layouting tidak direkomendasikan.

- CSS Float Property

Penggunaan CSS Float property sangat sering digunakan pada website. Float sangat mudah dipelajari dan di aplikasikan. Kekurangannya adalah floating element terikat pada dokumen flow yang menjadikan layout tidak flexible.

- CSS Flexbox

Dengan menggunakan flexbox tampilan akan memiliki fleksibilitas dan dapat diakses dari berbagai bentuk dan resolusi tampilan layar dengan baik. Hanya saja tidak dapat berjalan pada browser Internet Explorer.

- CSS Framework

Dengan bantuan framework layout dapat memiliki fleksibilitas seperti penggunaan flexbox.

- CSS Grid

Dengan modul CSS Grid yang menawarkan fitur grid-based layout system dengan baris dan kolom membuat layouting lebih mudah tanpa perlu floating dan positioning. Hanya saja tidak dapat berjalan pada browser Internet Explorer dan Edge.

### 2.11.1. Exercise

Buatlah sebuah layout menggunakan table, berikan background color yang berbeda untuk setiap section nya (minimal terdapat 6 section).

## 2.12. Float

Property float digunakan untuk memberikan posisi dan memformat konten, contohnya memberikan gambar disamping tulisan seperti di koran. Value dari property float adalah:

- Left – Element akan berada disisi kiri (floating left) dari kontainernya.
- Right – Element akan berada disisi kanan (floating right) dari kontainernya.
- None – Element tidak akan float (akan ditampilkan sesuai dengan posisi aslinya).
- Inherit – Element akan mengikuti nilai float dari parent nya.

Ubahlah value float dari left ke right dan none pada contoh kasus dibawah:

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
  float: left;
}
</style>
</head>
<body>

<p>Perhatikan gambar yang ada ketika nilai float diubah.</p>

<p>
```



```


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus
imperdiet, nulla et dictum interdum, nisi lorem egestas odio,
vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est,
ultrices nec congue eget, auctor vitae massa. Fusce luctus
vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed
ornare eu, lobortis in odio. Praesent convallis urna a lacus
interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed
ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis
imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida
venenatis. Integer fringilla congue eros non fermentum. Sed
dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris
quis diam velit.

</p>

</body>
</html>
```

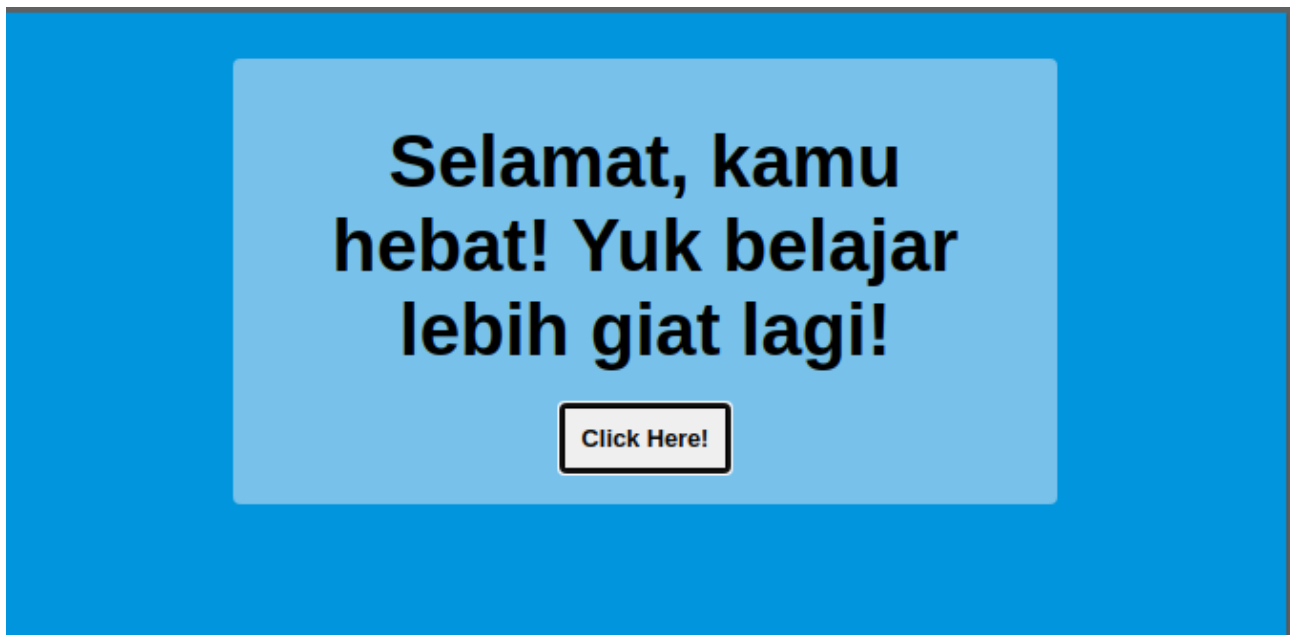
## 2.13. Studi Kasus Membuat Web Sederhana

Pada bagian ini, kita akan mencoba melihat step-step pembuatan sebuah web sederhana menggunakan HTML dan CSS di 1 file yang sama. Ada yang tahu, ini jenis penggunaan CSS kategori apa?

Pada kali ini, halaman web yang akan kita coba adalah membuat sebuah Mystery button. Tampilannya akan seperti gambar dibawah ini.



Nantinya, apabila kita klik tombol tersebut, tulisan diatas akan berubah menjadi seperti ini:



Penasaran, kan? Yuk ikuti langkah-langkah berikut!

Seperti membangun sebuah bangunan, tentunya perlu ada pondasi dan rangka dari rumah tersebut. Berikut adalah pondasi dan rangka yang akan kita gunakan dalam pembuatan web sederhana ini.

```
<!DOCTYPE html>
<html lang="en-us">

<head>
  <meta charset="utf-8">
  <title>Hello...</title>
</head>

<body>

</body>
```

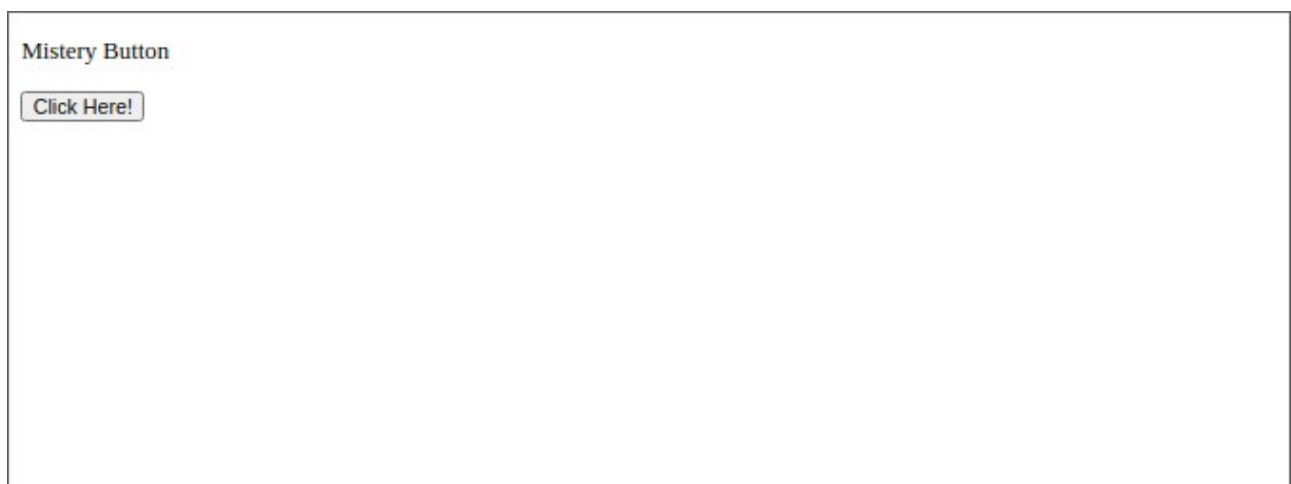
```
</html>
```

Pada script diatas, kita baru menambahkan kerangka HTML dan judul dari page tersebut saja. Selanjutnya, kita akan coba tambahkan konten dari web tersebut.

Di dalam tag <body>, kalian dapat memasukkan script berikut

```
<div>
  <p>?</p>
  <button>click me</button>
</div>
```

Jika sudah, kalian pasti akan memiliki tampilan web sebagai berikut



Selanjutnya, kita akan styling web tersebut agar lebih menarik. Karena kita akan menggabungkan css file dan html nya, makakita perlu menambahkan tag <style> di script html kita. Disini kita akan mencoba mengubah backgroud terlebih dahulu. Di dalam tag <head>, setelah <title>, kita masukkan script berikut:

```
<head>
  <meta charset="utf-8">
  <title>Hello...</title>

  <style>
    body {
```

```
background-color: #3498DB;
}
</style>
</head>
```

Setelah disave, Anda pasti akan mendapati halaman web akan berubah seperti berikut



Selanjutnya, kita bisa menambahkan script untuk styling komponen yang lainnya pada halaman tersebut. Berikut adalah script yang teman-teman bisa tambahkan.

```
<style>
body {
  background-color: #3498DB;
}

div {
  background-color: #85C1E9;
  padding: 20px;
  font-family: sans-serif;
  font-weight: bold;
  text-align: center;
  max-width: 500px;
  border-radius: 5px;
  margin: 0 auto;
  margin-top: 30px;
}

p {
  font-size: 3em;
  margin: 20px;
}

button {
  font-size: 16px;
  font-weight: bold;
  padding: 10px;
  border: 4px solid #000;
}

button:hover {
  background-color: #9FFFE3;
}

button:active {
  background-color: #000;
  color: #FFF;
}
</style>
```

ps: Silahkan diketik sendiri ya, sengaja kami tidak sediakan script yang dapat di copy-paste agar teman-teman banyak berlatih. Oiya, jangan asal salin juga ya, sebisa mungkin pahami script yang teman-teman ketikkan. Apabila ada yang tidak paham, silakan tanyakan kepada instruktur :)

Setelah script diatas ditambahkan, berikut adalah tampilan web kita:



Mantap kan? Nah langkah terakhir adalah menambahkan interaksi pada tombol (button) tersebut. Disini kita akan menggunakan **Javascript**. Kita memang belum belajar tentang Javascript, tapi, harapan penulis dengan adanya latihan ini bisa memicu teman-teman agar lebih “penasaran” dengan materi Javascript yang akan kita bahas di modul selanjutnya. Silakan teman-teman tambahkan script berikut di bagian body setelah tag `</div>`

```
<script>
  // accessing the elements from our page
  var buttonElement = document.querySelector("button");
  var textElement = document.querySelector("p");

  // listening to a click event on the button
  buttonElement.addEventListener("click", changeText, false);
```



Save dokumen html kalian, dan cobalah di browser!

Buatlah sebuah tampilan semirip mungkin dengan koran seperti gambar dibawah ini menggunakan metode float untuk memposisikan gambarnya.

