

# WEB APPLICATION DEVELOPMENT



## MODUL 1 HTML DAN CSS DENGAN BOOTSTRAP



ENTERPRISE  
APPLICATION  
DEVELOPMENT

## TIM PENYUSUN

<b>Nama</b>	<b>Kode Asisten</b>
Afina Ramadhani	<b>NOY</b>
Alwan Alyafi Mulyawan	<b>ALW</b>
Annisa Syafarani Callista	<b>CAL</b>
Arrizal Fauzan Barliana	<b>AFB</b>
Arsike Cipta Pelangi	<b>ARS</b>
Bima Susila Mukti	<b>BIM</b>
Desita Nur R.	<b>DST</b>
I Komang Gede Andhi Kurniawan	<b>AND</b>
Idellia Muthia Nurbaisa	<b>IDL</b>
Ivan Priyambudi	<b>IVN</b>
M. Ikhsan Pratama P.	<b>ISP</b>
Muhammad Ricky Chandra Dinata	<b>RCD</b>
Mukhamad Furqon	<b>WGS</b>
Nafidzah Kiasati Shadrina	<b>NFZ</b>
Rizky Azis Jayasutisna	<b>SKY</b>
Sabila Chanifah	<b>BYL</b>
Sherly Fratista	<b>SHE</b>
Syfa Nur Lathifah	<b>SYF</b>
Tomi Mulhartono	<b>TOM</b>

## DAFTAR ISI

TIM PENYUSUN.....	1
DAFTAR ISI.....	2
TUJUAN PRAKTIKUM.....	4
1. HTML.....	4
1.1. HTML Elements .....	4
1.2. HTML Attributes .....	8
1.3. HTML Image .....	8
1.4. HTML iframe .....	9
1.5. HTML Table.....	9
1.6. HTML Form.....	10
1.6.1. Action Attribute .....	10
1.6.2. Target Attribute .....	11
1.6.3. Method Attribute .....	11
1.6.4. Form Element .....	11
2. CSS.....	15
2.1. <i>Selector</i> .....	15
2.2. Penulisan CSS .....	16
2.3. <i>Text Formatting</i> .....	18
2.3.1. Warna Teks.....	18
2.3.2. <i>Text Alignment</i> .....	18
2.3.3. <i>CSS Font Family</i> .....	19
2.3.4. <i>CSS Font Style</i> .....	19
2.4. <i>CSS Styling</i> .....	20
2.4.1. <i>Background</i> .....	20
2.4.2. <i>Link</i> .....	21
2.4.3. <i>List</i> .....	22
3. Bootstrap .....	24
3.1. Manfaat Menggunakan Bootstrap .....	24
3.2. Cara Menggunakan Bootstrap .....	25

3.3.	Cara Menggunakan Bootstrap.....	29
3.3.1.	Membuat Tabel Dengan Bootstrap .....	29
3.3.2.	Mengatur Tampilan Gambar Dengan Bootstrap .....	30
3.3.3.	Membuat Pesan Peringatan ( <i>Alert</i> ) Dengan Bootstrap.....	32
3.3.4.	Membuat Tombol Dengan Bootstrap .....	34
3.3.5.	Membuat Panel Dengan Bootstrap .....	35
DAFTAR PUSTAKA .....		38

## TUJUAN PRAKTIKUM

1. Dapat memahami dan mengimplementasikan HTML
2. Dapat memahami dan mengimplementasikan CSS
3. Dapat memahami dan mengimplementasikan Bootstrap

## 1. HTML

HTML adalah singkatan dari *Hypertext Markup Language* yang merupakan bahasa *markup* standar untuk membuat halaman web. HTML dikembangkan pertama kali oleh Tim Berners-Leem Robert Cailliau dan beberapa orang lain pada tahun 1989.

HTML menggambarkan struktur halaman web yang terdiri dari serangkaian elemen dengan memberi tahu *browser* cara menampilkan konten. HTML terdiri dari elemen-elemen yang menyusunnya dan setiap elemen ditandai dengan adanya *tag*. Perlu diperhatikan jika penulisan *tag* HTML ini tidak memiliki sifat *case sensitive*. HTML berjalan di sisi *client* dan **bukan** merupakan bahasa pemrograman karena HTML tidak memiliki perintah untuk menyelesaikan masalah. HTML hanya sebagai tempat untuk meletakkan perintah bahasa pemrograman web seperti PHP dan Javascript.

### 1.1. HTML Elements

Setiap elemen pada HTML akan diawali oleh *tag* pembuka **<elemen>** dan diakhiri oleh *tag* penutup **</elemen>**, kecuali pada elemen kosong seperti *tag* **<br>**, **<img>** dan **<hr>**. Perlu diingat, di antara *tag* pembuka dan penutup terdapat sebuah konten.

Contoh penulisan elemen HTML:

```
<html>
  <head>
    <title>EAD Laboratory</title>
  </head>

  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Dari kode di atas, hasilnya akan seperti gambar di bawah ini:



Adapun beberapa elemen penyusun pada halaman web diatas, yaitu:

Elemen	Penjelasan
<!DOCTYPE html>	Deklarasi mendefinisikan bahwa versi dari HTML yang digunakan adalah HTML5.
<html>	Elemen <i>root</i> dari halaman HTML
<head>	Elemen yang berisikan informasi meta tentang halaman HTML.
<title>	Elemen untuk menentukan judul pada halaman HTML (yang ditampilkan di tab halaman).
<body>	Elemen yang merupakan wadah untuk semua konten yang terlihat.
<h1>	Elemen ini mendefinisikan <i>heading</i> dengan ukuran besar.
<p>	Elemen ini mendefinisikan sebuah paragraf.

Selain elemen diatas, adapun beberapa elemen yang digunakan untuk melakukan *formatting* pada halaman web:

Elemen	Penjelasan
<div>	Mendefinisikan blok konten yang bisa diterapkan <i>style</i> dan penempatannya sebagai satu unit.
<frame>	Digunakan untuk menempatkan konteks browser yang independen.
<iframe>	Membuat <i>frame inline</i> yang menyimpan dokumen HTML independen ke dokumen saat ini.
<form>	Digunakan untuk menyimpan data yang dimasukkan oleh pengguna.
<input>	Digunakan sebagai tempat pengguna dapat memasukkan data pada form.
<select>	Digunakan dengan satu atau lebih tag <b>&lt;option&gt;</b> untuk membuat menu <i>dropdown</i> .
<option>	Digunakan bersama dengan tag <b>&lt;select&gt;</b> , tiap tag <b>&lt;option&gt;</b> akan menyimpan satu pilihan <i>dropdown</i> .
<table>	Digunakan untuk membuat tabel HTML.
<th>	Digunakan untuk membuat kolom judul pada tabel HTML.
<tr>	Mendefinisikan baris pada tabel HTML.
<td>	Mendefinisikan kolom pada tabel HTML.
<b> atau <strong>	Digunakan untuk menampilkan teks dalam format <i>bold</i> .
<i> atau <em>	Digunakan untuk menampilkan teks dalam format <i>italic</i> .
<u>	Digunakan untuk menampilkan teks dalam format <i>underline</i> .
<a>	Digunakan untuk mendefinisikan <i>hyperlink</i> .

<code>&lt;img&gt;</code>	Digunakan untuk menyisipkan sebuah gambar.
<code>&lt;button&gt;</code>	Digunakan untuk memproses data inputan.
<code>&lt;ul&gt;</code>	Daftar yang tidak berurut.
<code>&lt;ol&gt;</code>	Daftar yang berurut.
<code>&lt;li&gt;</code>	Elemen <i>item</i> di dalam tag <code>&lt;ul&gt;</code> atau tag <code>&lt;ol&gt;</code> .
<code>&lt;h1&gt;</code> , <code>&lt;h2&gt;</code> , ..., <code>&lt;h6&gt;</code>	Elemen yang menampilkan teks judul. Semakin besar angka setelah "h", maka ukuran teks semakin kecil.

Terdapat dua macam daftar (*list*) pada HTML, yaitu *unordered list* `<ul>` dan *ordered list* `<ol>`.

```
Unordered List:
<ul>
  <li>Apel</li>
  <li>Jeruk</li>
  <li>Mangga</li>
</ul>

Ordered List:
<ol>
  <li>Apel</li>
  <li>Jeruk</li>
  <li>Mangga</li>
</ol>
```

Dari kode di atas, hasilnya akan seperti gambar dibawah ini:

```
Unordered List:

• Apel
• Jeruk
• Mangga

Ordered List:

1. Apel
2. Jeruk
3. Mangga
```



## 1.2. HTML Attributes

Setiap elemen pada HTML dapat mempunyai satu atau lebih atribut. Atribut akan menyediakan informasi tambahan mengenai elemen tersebut. Atribut selalu terletak pada *tag* pembuka pada suatu elemen. Semua elemen HTML dapat memiliki atribut.

```
<h1>HTML Attribute</h1>
<p>My first paragraph.</p>

<a href="https://www.instagram.com/ead.fse/">Klik Disini!</a>
```

Dari kode diatas, hasilnya akan seperti gambar di bawah ini:

### HTML Attribute

My first paragraph.

[Klik Disini!](https://www.instagram.com/ead.fse/)

Seperti contoh di atas, terdapat atribut **href** pada elemen **<a>**. Elemen **<a>** merupakan *hyperlink* yang dapat menyimpan *link* dari suatu halaman lain, sedangkan atribut **href** tersebut akan mendefinisikan URL yang akan dituju.

## 1.3. HTML Image

HTML *Image* memiliki *tag* **<img>** yang digunakan untuk menyisipkan sebuah gambar pada halaman web. Elemen ini biasanya diikuti oleh atribut **src** yang akan mendefinisikan URL dari gambar yang ditampilkan tersebut. Sementara **alt** adalah teks alternatif yang akan ditampilkan saat gambar tidak dapat ditampilkan.

```

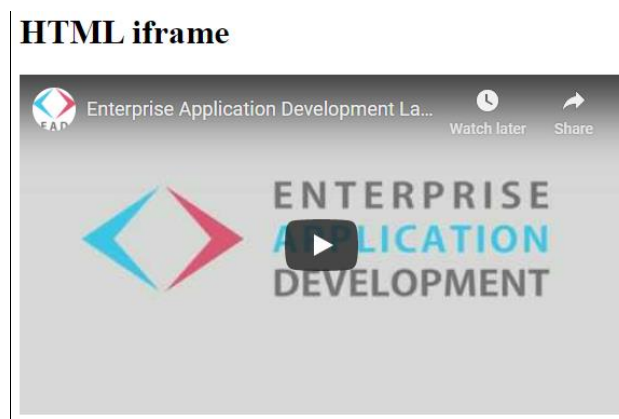
```

#### 1.4. HTML iframe

HTML *iframe* memiliki tag **<iframe>** dan akan menyimpan *frame inline* yang akan menyimpan dokumen HTML independen. Tag **<iframe>** ini dapat digunakan untuk menampilkan halaman *file* PDF atau video. Selain itu, tag ini berfungsi untuk menambahkan atau menyertakan konten dari *source* (sumber) yang lain.

```
<h1>HTML iframe</h1>
<iframe width="560" height="315" src="https://www.youtube.com/embed/AwMeQJ_u73I" frameborder="0"
allow="accelerometer; autoplay; encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
```

Dari kode diatas, hasilnya akan seperti gambar di bawah ini:



#### 1.5. HTML Table

HTML Table akan diawali dengan tag **<table>**. Tabel tersebut merupakan cara untuk menampilkan informasi dalam bentuk sel yang terdiri atas baris dan kolom. Sebuah tabel dibagi menjadi baris-baris, dan tiap baris dibagi ke dalam *cell-cell*. Pada tabel ini digunakan tag **<tr>** untuk mendefinisikan baris, tag **<th>** digunakan untuk mendefinisikan kolom *header*, dan tag **<td>** digunakan untuk mendefinisikan kolom. Untuk membuat satu kolom memenuhi lebih dari satu baris, dapat menggunakan atribut **colspan**, sedangkan untuk membuat satu baris memenuhi lebih dari satu baris, dapat menggunakan atribut **rowspan**.

```
<table border="1">
  <tr>
    <th>Full Name</th>
    <th>Laboratory</th>
  </tr>
  <tr>
    <td>Tom i Mulhartono</td>
    <td>EAD Laboratory</td>
  </tr>
</table>
```

Dari kode di atas, hasilnya akan seperti gambar dibawah ini:

Full Name	Laboratory
Tom i Mulhartono	EAD Laboratory

## 1.6. HTML Form

Tag elemen HTML **<form>** digunakan untuk menyimpan *input* dari pengguna. Elemen **<form>** berisi elemen *input* seperti *text fields*, *checkboxes*, *radio buttons*, *submit buttons*, dan masih banyak lagi. Form akan diproses guna mendapatkan informasi tertentu dari atau untuk user.

### 1.6.1. Action Attribute

Atribut **action** mendefinisikan aksi yang akan dilakukan saat form di-submit, biasanya melalui sebuah tombol. Atribut tersebut mendefinisikan tindakan apa yang akan dilakukan ketika form dikirimkan.

```
<form action="/action_page.php">
```

Pada contoh di atas, data pada form akan dikirim ke halaman yang memiliki nama `action_page.php` pada server yang mengandung *server-side script* yang akan mengolah data.

### 1.6.2. Target Attribute

Atribut **target** menjelaskan jika data yang di-submit akan membuka *tab* baru pada *browser*, sebuah *frame*, atau tetap pada halaman saat ini. Nilai *default* dari atribut **target** adalah **\_self** yang membuat form akan di-submit pada halaman saat ini. Untuk membuat form membuka halaman baru, dapat digunakan *value* **\_blank**.

### 1.6.3. Method Attribute

Atribut **method** menjelaskan *method* HTTP (**GET** atau **POST**) yang digunakan saat melakukan *submit* data. Hal ini berguna untuk menentukan metoda apa yang digunakan untuk mengirim data ke *script* tujuan. *Method default* yang digunakan saat melakukan *submit* data adalah GET. Namun, perlu diingat jika data yang dikirim akan terlihat pada alamat URL halaman, biasanya tertulis setelah tanda ? pada akhir URL. Maka, disarankan untuk selalu menggunakan *method* POST saat mengirimkan data sensitif.

```
<form action="/action_page.php" method="GET">
```

```
<form action="/action_page.php" method="POST">
```

### 1.6.4. Form Element

#### a) Input Element

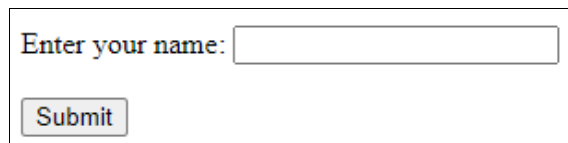
Elemen **<input>** merupakan elemen yang penting, **<input>** dapat ditampilkan dalam beberapa cara, tergantung dari atribut *type* yang ditulis. Elemen ini digunakan untuk menunjukkan sebuah *inputan* (masukkan) dalam bentuk kotak dan sejenisnya yang dapat diedit/diketik untuk diisi data tertentu.

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

Berikut adalah contoh penggunaan elemen **<input>**:

```
<form action="/action_page.php" method="POST">
  Enter your name:
  <input type="text" name="name" id="">
  <br><br>
  <input type="submit" value="submit">
</form>
```

Dari kode di atas, hasilnya akan seperti gambar di bawah ini:



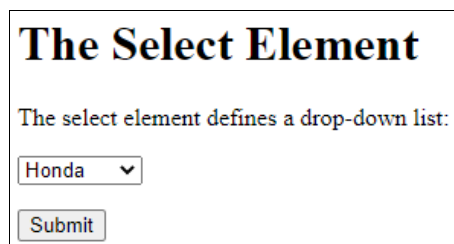
Enter your name:

## b) Select Element

Elemen **<select>** mendefinisikan pilihan dalam bentuk daftar *dropdown*. Elemen ini membuat objek *form* yang berupa *list* pilihan yang dapat dipilih oleh pengguna. Berikut adalah contoh penggunaan elemen **<select>**:

```
<form action="/action_page.php" method="POST">
  <select name="motor" id="">
    <option value="Honda">Honda</option>
    <option value="Kawasaki">Kawasaki</option>
    <option value="Yamaha">Yamaha</option>
    <option value="Suzuki">Suzuki</option>
  </select>
  <input type="submit" value="submit">
</form>
```

Dari kode di atas, hasilnya akan seperti gambar di bawah ini:



**The Select Element**

The select element defines a drop-down list:

Honda ▼

Submit

Tag elemen **<option>** mendefinisikan pilihan apa saja yang dapat dipilih. **<option>** pertama akan menjadi nilai pertama yang terpilih.

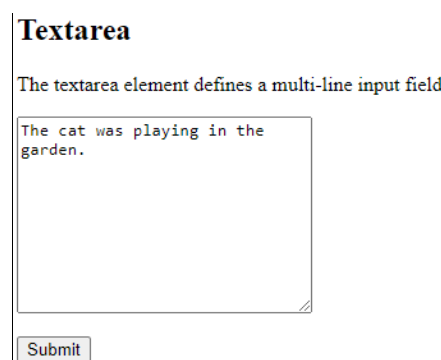
### c) Text Area Element

Tag elemen **<textarea>** mendefinisikan tempat *input* teks banyak baris. Berikut adalah contoh penggunaan elemen **<textarea>** :

```
<h2>Textarea</h2>
<p>The textarea element defines a multi-line input field.</p>

<form action="/action_page.php">
  <textarea name="message" cols="30" rows="10">The cat was playing in the garden.</textarea>
  <br><br>
  <input type="submit" value="Submit">
</form>
```

Dari kode di atas, hasilnya akan seperti gambar di bawah ini:



**Textarea**

The textarea element defines a multi-line input field.

The cat was playing in the garden.

Submit

#### d) Button Element

*Tag* elemen **<button>** mendefinisikan tombol yang dapat ditekan untuk menjalankan tindakan tertentu. Berikut adalah contoh penggunaan elemen **<button>**:

```
<h2>The Button Element</h2>  
<button type="button" onclick="alert('Hello World!')">Click Me!</button>
```

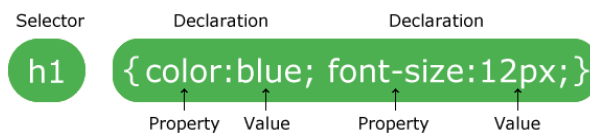
Dari kode di atas, hasilnya akan seperti gambar di bawah ini:



## 2. CSS

CSS merupakan singkatan dari **Cascading Style Sheet** yang bertujuan untuk menghias dan mengatur gaya tampilan atau *layout* halaman web agar lebih menarik. CSS menentukan bagaimana elemen HTML ditampilkan, seperti menentukan posisi, mengubah warna teks dan *background*, dan lain sebagainya.

### 2.1. Selector



*Selector* pada CSS menunjuk ke elemen yang akan dimodifikasi. Pada kode di atas, elemen yang akan dirujuk oleh kode CSS adalah **<h1>**. Elemen **<h1>** tersebut akan dimodifikasi sehingga berukuran **12px** dan berwarna **biru**. Sehingga untuk mengubahnya, kita akan memberi *property* **font-size** dengan *value* **12px** dan *property* **color** dengan *value* **blue**. Penulisan *property* dan *value* pada CSS menggunakan tanda titik dua (:). Untuk memisahkan antara *property style* satu dengan lainnya, kita dapat menggunakan tanda titik koma (;). Penulisan *property* dan *value* diletakan di antara tanda kurawal.

Dalam CSS, *id selector* menggunakan atribut *id* dari elemen HTML untuk memilih elemen tertentu. *Id* suatu elemen di dalam suatu halaman adalah **unik**, jadi **id selector** digunakan untuk memilih satu **elemen unik**. Untuk memilih *id* elemen tertentu, maka gunakan tanda *hash* (**#**) dan diikuti oleh *id* elemen.

*Class selector* memilih elemen HTML dengan atribut kelas tertentu. Untuk memilih elemen dengan kelas tertentu, maka gunakan tanda titik (**.**) dan diikuti oleh nama kelas.



```
<DOCTYPE html>
<html>

<head>
<title>Paragraph</title>
<style>
  #top {
    background-color: #F5F5F5;
    padding: 20px;
  }

  #judul {
    color: red;
  }

  .paragraf {
    color: blue;
    font-weight: bold;
  }
</style>
</head>

<body>
<div id="top">

<h1 id="judul">Mulan</h1>
<p class="paragraf">Mulan akhirnya rilis pada 4 September kemarin melalui platform streaming Disney+.</p>
<p class="paragraf">Film ini merupakan live-action remake yang diadaptasi dari film animasi Disney, Mulan pada tahun 1998.</p>
</div>
</body>

</html>
```

Dari kode di atas, hasilnya akan seperti gambar di bawah ini:



## 2.2. Penulisan CSS

Ada tiga metode penulisan kode CSS, yaitu *inline*, *internal*, dan *external*. Berikut contoh penggunaan ketiga metode tersebut:

### a. *Inline*

Penulisan kode CSS dengan metode *inline* dapat dilakukan langsung pada *tag* yang ingin diberi *style* dengan menggunakan atribut *style*. Contoh:

```
<h1 style="color: red;">Mulan</h1>
```

Pada metode ini, kita tidak perlu menuliskan *selector* karena kita menuliskan CSS langsung pada *tag* yang ingin diberi *style*.

### b. *Internal*

Metode CSS *internal* ditulis di dalam *tag style* yang ditempatkan pada *tag head*. Contoh:

```
<head>
<title>Paragraph</title>
<style>
  #top {
    background-color: #F5F5F5;
    padding: 20px;
  }

  #judul {
    color: red;
  }

  .paragraf {
    color: blue;
    font-weight: bold;
  }
</style>
</head>
```

Metode ini dianjurkan untuk pengujian *style* atau ketika anda hanya memiliki satu halaman web.

### c. *External*

Metode ini dilakukan dengan membuat *file* CSS dan kemudian memanggil *file* tersebut di dalam *tag head*. *File* CSS memiliki **ekstensi (akhiran) .css**, misalnya *style.css*. pemanggilan *file* ini dilakukan dengan menggunakan *tag link* seperti contoh:

```
<head>
<title>Paragraph</title>
<link rel="stylesheet" href="namafile.css">
</head>
```

Metode *external* CSS menggunakan *class* yang berfungsi mendefinisikan *style* untuk elemen yang menggunakan *class* tersebut.

```
<style>
  .paragraf {
    color: blue;
    font-weight: bold;
  }
</style>
```

Dari *file* *namafile.css* di atas akan menghasilkan contoh implementasi *selector class* pada HTML sebagai berikut.

```
<body>
<p class="paragraf">Mulan akhirnya rilis pada 4 September kemarin
  melalui platform streaming Disney+.</p>
<p class="paragraf">Film ini merupakan live-action remake yang diadaptasi
  dari film animasi Disney, Mulan pada tahun 1998.</p>
</div>
</body>
```

## 2.3. Text Formatting

*Text formatting* merupakan modifikasi *property* pada teks dengan menggunakan CSS, seperti memberi warna dan jenis *font* pada teks.

### 2.3.1. Warna Teks

Warna teks dapat ditentukan oleh tiga nilai dalam CSS, yaitu:

Nilai	Warna	Kode
Nama	"yellow", "blue", "red"	<code>&lt;h1 style="color: red;"&gt;Mulan&lt;/h1&gt;</code>
HEX	"#F5F5F5", "#E6E6FA", "#00FFFF"	<code>&lt;h1 style="color: #00FFFF;"&gt;Mulan&lt;/h1&gt;</code>
RGB	"rgb(0,128,128)", "rgb(0,255,0)", "rgb(8,8,8)"	<code>&lt;h1 style="color: rgb(0,128,128);"&gt;Mulan&lt;/h1&gt;</code>

### 2.3.2. Text Alignment

*Property text-align* digunakan untuk mengatur perataan teks pada posisi horizontal dari sebuah kumpulan teks. Teks dapat diatur menjadi rata kiri (*left*), rata tengah (*center*), rata kanan (*right*), dan rata kanan kiri (*justify*).

```
<style>
  .kiri {
    text-align: left;
  }
</style>
</head>

<body>
  <div class="kiri">
    <h1>Ini Judul</h1>
    <p>Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri.
      Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri.
      Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri.
      Ini adalah paragraf rata kiri.</p>
  </div>
</body>
```

## Ini Judul

Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri. Ini adalah paragraf rata kiri.

### 2.3.3. CSS Font Family

Di dalam CSS kita dapat menggunakan berbagai jenis *font* sesuai dengan keinginan. Namun *font* yang akan ditampilkan pada web *browser* sepenuhnya berasal dari komputer pengguna. Jika kita memilih *font* yang tidak standar, maka tampilan web yang telah dirancang serapi mungkin bisa menjadi berantakan karena *font* yang telah kita pilih belum tentu tersedia di komputer pengguna.

Hal ini dapat diatasi dengan menggunakan *font* standar yang tersedia di berbagai sistem operasi seperti Windows, Linux, Mac OS, dll. Dengan menggunakan *font* standar, desain web yang telah dirancang akan dapat didukung oleh mayoritas web *browser*.

Berikut adalah contoh *font* standar yang dapat digunakan dalam desain web:

```
<style type="text/css">
  body {
    font-family: Arial;
  }

  h1 {
    font-family: Georgia, monospace;
  }

  div {
    font-family: "Duru Sans", Verdana, sans-serif;
  }
</style>
```

Catatan: jika nama *font* **lebih dari satu kata**, maka harus **ditulis dalam tanda kutip**, seperti **font-family: "Duru Sans"**. Jika ada lebih dari satu *font-family*, maka saat penulisan dapat dipisah dengan tanda koma (,).

### 2.3.4. CSS Font Style

Properti CSS *Font Style* digunakan untuk menentukan gaya teks yang akan ditampilkan, misalnya menentukan apakah *font* ditampilkan dengan model miring atau normal. Untuk jenis *font* yang tidak memiliki varian *italic* dan *oblique*, maka *browser* akan tetap menampilkan gaya tersebut secara artifisial.

a. Normal

Normal adalah nilai *default* pada *font style* dan memberikan efek karakter yang tegak. Nilai normal dapat digunakan pada anak elemen HTML yang induk elemen HTML-nya menerapkan efek *italic* dan *oblique*.

b. *Italic*

Sebagian besar jenis font memiliki versi karakter khususnya, yaitu *italic*. Jika nilai *italic* ditentukan pada *property font-style*, maka *browser* akan menampilkan versi karakter *italic* pada font tersebut.

c. *Oblique*

Secara visual, *font style italic* dan *oblique* terlihat sama. Namun, nilai *oblique* dapat disebut juga sebagai *italic* buatan. Hal itu disebabkan oleh *browser* yang akan **memaksa font untuk ditampilkan ke dalam keadaan miring** meskipun jenis *font* yang digunakan tidak memiliki versi *italic*.

Berikut merupakan contoh penulisan kode untuk *font-style*.

```
<style type="text/css">
  body {
    font-family: Arial;
    font-style: normal;
  }

  h1 {
    font-family: Georgia, monospace;
    font-style: italic;
  }

  div {
    font-family: "Duru Sans", Verdana, sans-serif;
    font-style: normal;
  }
</style>
```

## 2.4. CSS Styling

### 2.4.1. Background

CSS dapat digunakan untuk mendefinisikan efek yang terjadi pada latar belakang dan elemennya. CSS properti yang digunakan untuk mengatur efek latar belakang (*background*) yaitu:

Nilai	Penjelasan
background-color	<i>Property</i> ini menentukan warna latar belakang suatu elemen
background-image	<i>Property</i> ini menetapkan gambar yang akan digunakan sebagai latar belakang suatu elemen
background-repeat	Secara <i>default</i> , <i>property</i> ini mengulangi gambar latar belakang baik secara horizontal dan vertikal
background-attachment	<i>Property</i> ini menentukan apakah gambar latar belakang harus <i>discroll</i> atau <i>fixed</i> (tidak akan <i>discroll</i> )
background-position	<i>Property</i> ini digunakan untuk menentukan posisi gambar latar belakang

*Property* latar belakang dapat menggunakan warna maupun gambar. Secara *default*, gambar diulang sehingga meliputi seluruh elemen. Gambar latar belakang untuk sebuah halaman dapat diatur seperti berikut ini.

```
<style type="text/css">
  body {
    width: 500px;
    height: 500px;
    background-image: url('bg1.jpg');
    background-attachment: fixed;
    background-repeat: no-repeat;
  }
</style>
```

#### 2.4.2. Link

CSS dapat mengatur *property* dari *link*, seperti warna *link*, *font-family*, latar belakang, dan lain-lain. Berikut contoh pengaturan *property link*:

Nilai	Penjelasan	Kode	Hasil
a:link	Normal, kondisi saat <i>link</i> <b>belum dikunjungi</b>	<pre>a:link {   color: blue; }</pre>	<a href="#">Ini Link</a>

a:visited	Kondisi saat <i>link telah dikunjungi</i>	<pre>a:visited { color: red; }</pre>	<u>Ini Link</u>
a:hover	Kondisi saat <i>mouse ada di atas link</i>	<pre>a:hover { color: hotpink; }</pre>	<u>Ini Link</u>
a:active	Kondisi <i>link ketika ditekan</i>	<pre>a:active { color: green; }</pre>	<u>Ini Link</u>

### 2.4.3. List

HTML memiliki dua jenis utama *lists*, yaitu:

Nama	Tag	Penjelasan
<i>Unordered Lists</i>	<ul>	<i>List items</i> ditandai dengan <b>bullets</b>
<i>Ordered Lists</i>	<ol>	<i>List items</i> ditandai dengan <b>angka atau huruf</b>

*Property lists* CSS memungkinkan untuk:

- Sebagai penanda *list item* untuk *ordered lists*
- Sebagai penanda *list item* untuk *unordered lists*
- Sebagai gambar untuk penanda *list item*
- Menambahkan warna latar belakang ke *lists* dan *list item*

Berikut adalah contoh penulisan CSS dan HTML pada *list*:

```
<style type="text/css">
/* bullet pada unordered list berbentuk lingkaran */
.ul-a {
list-style-type: circle;
}

/* bullet pada unordered list berbentuk persegi */
.ul-b {
list-style-type: square;
}

/* urutan pada ordered list adalah romawi kapital
(I, II, III, IV, V, dst.) */
.ol-a {
list-style-type: upper-roman;
}

/* urutan pada ordered list adalah huruf alfabet kecil
(a, b, c, d, e, dst.) */
.ol-b {
list-style-type: lower-alpha;
}
```

```
<body>
<p>Daftar nama buah: </p>
<ul class="ul-a">
  <li>pisang</li>
  <li>mangga</li>
  <li>anggur</li>
</ul>

<p>Daftar nama sayuran: </p>
<ul class="ul-b">
  <li>bayam</li>
  <li>selada</li>
  <li>kangkung</li>
</ul>

<p>Daftar nama kendaraan darat: </p>
<ol class="ol-a">
  <li>motor</li>
  <li>mobil</li>
  <li>becak</li>
</ol>

<p>Daftar nama alat elektronik: </p>
<ol class="ol-b">
  <li>komputer</li>
  <li>microwave</li>
  <li>mesin cuci</li>
</ol>
</body>
```

Hasil dari kode di atas adalah sebagai berikut:

Daftar nama buah:

- o pisang
- o mangga
- o anggur

Daftar nama sayuran:

- bayam
- selada
- kangkung

Daftar nama kendaraan darat:

- I. motor
- II. mobil
- III. becak

Daftar nama alat elektronik:

- a. komputer
- b. microwave
- c. mesin cuci



### 3. Bootstrap

Bootstrap adalah kerangka kerja *front-end* yang digunakan dalam pengembangan web yang lebih efisien dan efektif sehingga *developer* hanya perlu memanggil *class* tertentu untuk membuat tombol, panel, tabel, pesan peringatan, dan lain sebagainya.

Bootstrap mencakup template desain berbasis HTML dan CSS untuk tipografi, formulir, tombol, tabel, navigasi, *modals*, *carousel* gambar, dan banyak lainnya, serta *library* JavaScript opsional. Selain itu, Bootstrap juga dapat digunakan untuk membuat desain responsif (mendukung segala jenis resolusi perangkat, mulai dari ponsel kecil hingga desktop besar) dengan mudah dan *simple*.

Bootstrap terdiri dari beberapa *file*. *File* pada Bootstrap berisi kumpulan baris kode tersusun dari CSS dan JavaScript yang berbentuk *class*. Jadi ketika akan menggunakan Bootstrap untuk mengembangkan *website*, dalam membuat satu tombol tidak perlu menyusun beberapa baris kode karena hanya akan memanggil salah satu *class* yang terdapat pada Bootstrap.

#### 3.1. Manfaat Menggunakan Bootstrap

1. **Mudah digunakan:** Siapa saja yang hanya memiliki pengetahuan dasar tentang HTML dan CSS dapat mulai menggunakan Bootstrap.
2. **Fitur responsif:** CSS responsif Bootstrap menyesuaikan dengan ponsel, tablet, dan desktop.
3. **Kompatibilitas *browser*:** Bootstrap kompatibel dengan semua *browser* modern (Chrome, Firefox, Internet Explorer, Edge, Safari, dan Opera).
4. **Dapat disesuaikan:** Bootstrap dapat memilih komponen yang diinginkan dan yang tidak diinginkan.
5. **Menghemat Waktu:** Bootstrap akan menghemat waktu dalam melakukan pengembangan suatu *website*, karena Bootstrap membuat kode yang sederhana dan singkat.

### 3.2. Cara Menggunakan Bootstrap

Dalam menggunakan Bootstrap ada beberapa opsi pilihan. Dapat dilakukan dengan Composer, Bower, atau NPM. Selain itu Bootstrap juga dapat digunakan secara *offline* dan *online*.

#### 1. Cara menggunakan Bootstrap dengan Composer

Jika ingin menggunakan dengan Composer, cukup dengan mengetikkan perintah berikut pada composer:

```
1 composer require twbs/bootstrap
```

#### 2. Cara menggunakan Bootstrap dengan Bower

Jika ingin menggunakan Bower, cukup dengan mengetikkan perintah berikut pada Bower:

```
1 bower install bootstrap
```

#### 3. Cara menggunakan Bootstrap dengan NPM

Jika ingin menggunakan NPM, cukup dengan mengetikkan perintah berikut pada NPM:

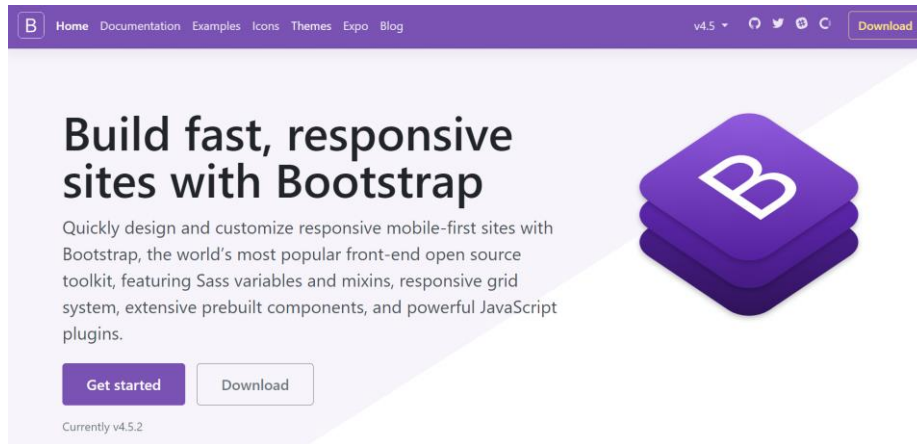
```
1 npm install bootstrap
```

#### 4. Cara menggunakan Bootstrap secara *offline*

Proses instalasi Bootstrap secara *offline* ini akan memudahkan dalam proses pengembangan *website*. Jadi, *website* tidak perlu mengambil *file (resource)* dari *website* lain, tapi langsung dari penyimpanan utama.

##### Langkah 1: Buat Direktori Baru Kemudian Download Bootstrap

Tambahkan folder baru ke dalam folder *root web server*. Kemudian download Bootstrap dari *website* resminya (<https://getbootstrap.com/>).



Di dalam folder ini terdapat beberapa *file* yang terkumpul ke dalam beberapa *assets* dan menyediakan beberapa fungsi dan *class* di dalamnya. *File* zip ini nantinya perlu dilakukan ekstrak terlebih dahulu ke dalam folder *project* agar dapat digunakan. Berikut merupakan susunan direktori dari Bootstrap, seperti di bawah ini:

```
bootstrap/
├── css/
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   ├── bootstrap.min.css.map
│   ├── bootstrap-theme.css
│   ├── bootstrap-theme.css.map
│   ├── bootstrap-theme.min.css
│   └── bootstrap-theme.min.css.map
├── js/
│   ├── bootstrap.js
│   └── bootstrap.min.js
└── fonts/
    ├── glyphs-halflings-regular.eot
    ├── glyphs-halflings-regular.svg
    ├── glyphs-halflings-regular.ttf
    ├── glyphs-halflings-regular.woff
    └── glyphs-halflings-regular.woff2
```

Selain itu, Bootstrap juga memerlukan *jquery* karena untuk bagian JavaScriptnya, Bootstrap menggunakan *jquery* pada folder JS Bootstrap atau dapat diletakkan sesuai keinginan asalkan direktori saat menghubungkan *file* CSS dan JS yang disesuaikan.

## Langkah 2: Buat Sebuah *File* Index.html

Agar dapat menggunakan Bootstrap dan membuat halaman *website*, bukalah teks editor dan buat *file* **index.html**. Di dalam *file* index.html isikan beberapa baris kode, tetapi jangan lupa untuk memanggil *file* Bootstrap dan *library* lainnya. Sebagai contoh, di bawah ini adalah tampilan sederhana menggunakan Bootstrap dan beberapa *library* lain.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Modul 1 Web Application Development</title>
5      <link rel="stylesheet" type="text/css" href="css/bootstrap.css">
6      <script type="text/javascript" src="js/jquery.js"></script>
7      <script type="text/javascript" src="js/bootstrap.js"></script>
8  </head>
9  <body>
10     <h1>Cara Menggunakan Bootstrap</h1>
11     <button class="btn btn-danger">Secara Offline</button>
12     <button class="btn btn-primary">Secara Online</button>
13 </body>
14 </html>
```

Baris kode di atas tidak hanya memanggil *file* **Bootstrap.min.css** yang berada di folder CSS melalui baris ke-5, yaitu:

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

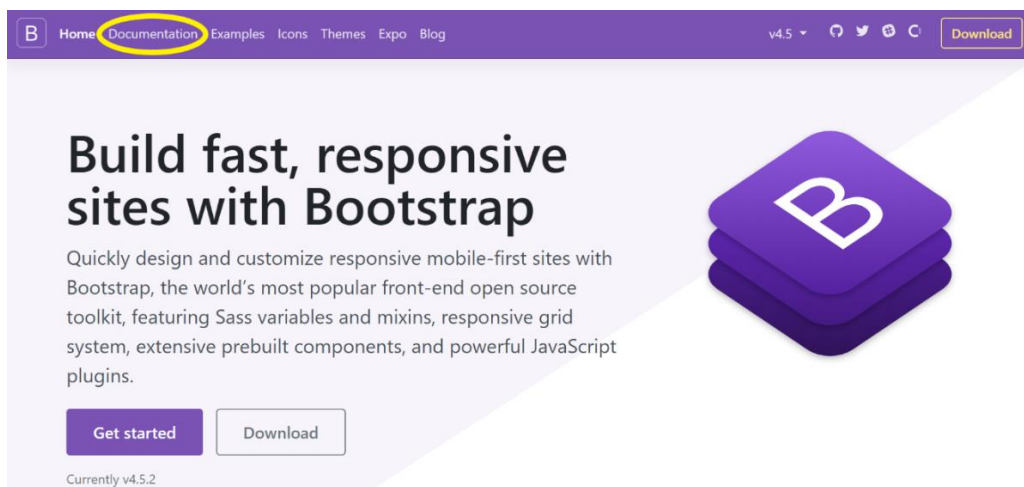
Kemudian untuk baris ke-11 dan ke-12 pada *class* "btn" pada elemen **<button>** merupakan *class* Bootstrap yang dapat digunakan untuk membuat tombol. "**btn-danger**" serta "**btn-primary**" digunakan dalam menentukan warna dari tombol yang akan dibuat untuk mendefinisikan kondisi tertentu.

```
11     <button class="btn btn-danger">Secara Offline</button>
12     <button class="btn btn-primary">Secara Online</button>
```

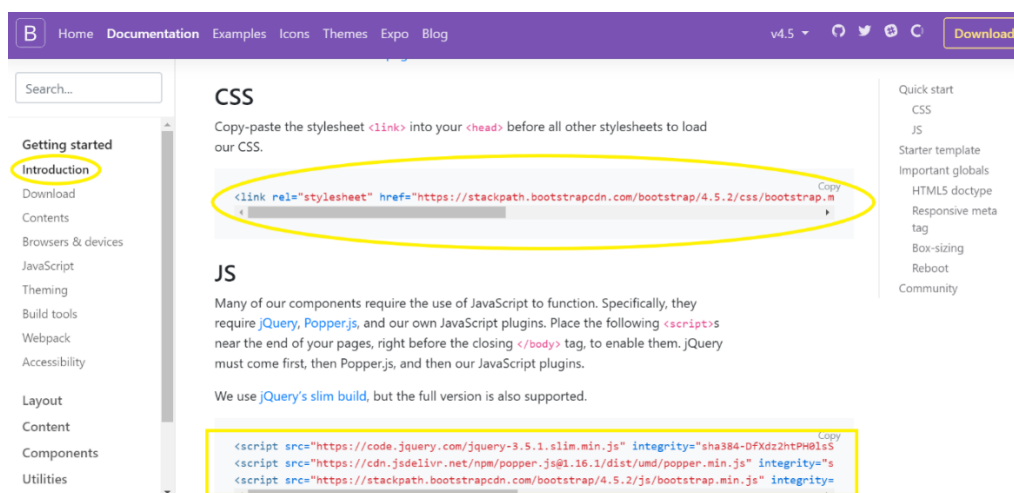
## 5. Cara menggunakan Bootstrap secara *online*

Penggunaan Bootstrap secara *online* digunakan jika tidak ingin menginstall Bootstrap. Cara yang dapat digunakan yaitu dengan memasukkannya dari CDN (Jaringan Pengiriman Konten). Namun terdapat kekurangan, yaitu ketika memanggil secara *online* yaitu akan mengurangi tingkat kecepatan *website*.

**Langkah 1:** Masuk ke dalam *website* Bootstrap, lalu pilih Documentation.



**Langkah 2:** Lalu pilih Introduction, dan *copy* pada semua CSS dan JS yang sudah ditandai seperti gambar di bawah. Setelah itu masukkan ke dalam *file* HTML yang akan digunakan. *Syntax* tersebut dapat diletakkan pada bagian **<head>** halaman HTML.



### 3.3. Cara Menggunakan Bootstrap

#### 3.3.1. Membuat Tabel Dengan Bootstrap

Bootstrap memberikan kemudahan untuk membuat tabel tanpa harus mengetikkan berbaris-baris kode program. Dengan hanya memanggil *class-class* yang sudah ada di dalam *file* Bootstrap. Untuk menggunakan *class* dengan kombinasi *class* lainnya, gunakan *class* **table** ditambah dengan *class* lain. Contoh penggunaannya seperti di bawah ini:

```
1  # untuk membuat tabel
2  <table class="table">
3  # untuk membuat tabel stripped
4  <table class="table table-stripped">
5  # untuk membuat tabel Bordered
6  <table class="table table-bordered">
7  # untuk membuat tabel hover
8  <table class="table table-hover">
```

Di dalam *file* Bootstrap sudah terdapat beberapa *class* yang telah didesain untuk kebutuhan pembuatan tabel, antara lain **".table"**, **".table-stripped"**, **".table-bordered"**, dan **".table-hover"**.

- **".table"**: *class* yang digunakan untuk membuat dan mendefinisikan tabel biasa/standar. *Class* ini hanya menambahkan lapisan yang kecil dan pembatas secara horizontal. *Class* ini menggunakan baris kode **<table class="table">**. Berikut merupakan tampilan dari penggunaan *class* **"table"**.

Nama Depan	Nama Belakang	Email
Sherly	Fratista	sherlyfratista@gmail.com
Annisa	Callista	annisacallista@gmail.com
Tomi	Mulhartono	tomimulhartono@gmail.com

- **".table-stripped"**: *class* yang digunakan untuk membuat *table* dengan gaya warna setiap baris berselang-seling. *Class* ini menggunakan baris kode **<table class="table table-stripped">**.

Berikut merupakan tampilan dari penggunaan *class* **“table-stripped”**.

Nama Depan	Nama Belakang	Email
Sherly	Fratista	sherlyfratista@gmail.com
Annisa	Callista	annisacallista@gmail.com
Tomi	Mulhartono	tomimulhartono@gmail.com

- **“table-bordered”**: merupakan *class* yang digunakan untuk menampilkan *border* di dalam *table*. Jadi keseluruhan *table* akan dipasang sebuah *border*. *Class* ini menggunakan baris kode **<table class=“table table-bordered”>**. Berikut merupakan tampilan dari penggunaan *class* **“table-bordered”**.

Nama Depan	Nama Belakang	Email
Sherly	Fratista	sherlyfratista@gmail.com
Annisa	Callista	annisacallista@gmail.com
Tomi	Mulhartono	tomimulhartono@gmail.com

- **“table-hover”**: *class* ini dapat digunakan untuk memberikan efek *hover* pada sebuah *row*. Efek ini akan berjalan ketika Anda meletakkan kursor di *row* yang diberikan *class* ini. *Class* ini menggunakan baris kode **<table class=“table table-hover”>**. Tampilan dari *table-hover* yaitu menambahkan efek *hover* (warna latar belakang abu-abu) pada baris tabel.

Selain kumpulan *class* untuk mendesain *table*, terdapat beberapa *class* pewarnaan *row* tabel dan data menggunakan Bootstrap, yaitu **“success”**, **“danger”**, **“info”**, **“warning”**, dan **“active”**. Dengan menggunakan kumpulan *class* ini, dapat digunakan dalam mendesain tabel yang menarik dan mudah dibaca.

### 3.3.2. Mengatur Tampilan Gambar Dengan Bootstrap

Dengan menggunakan Bootstrap juga dapat mendesain gambar. Membuat gambar yang *responsive*, berbentuk bulat, lingkaran, dan juga *thumbnail*. *Class* **“img-responsive”**, **“img-**

**rounded**", **img-circle**", dan **img-thumbnail**" adalah kumpulan *class* yang digunakan dalam memanipulasi gambar di Bootstrap. Contoh penggunaannya seperti di bawah ini:

```
1 
2 
3 
4 
```

- **".img-responsive"**: *class* ini memungkinkan untuk membuat gambar yang dapat dibuka di berbagai macam resolusi layar. *Class .img-responsive* membuka ukuran gambar dapat menyesuaikan dengan ukuran elemen inti. Dapat dilihat efeknya ketika melakukan *resize* pada *browser*. Berikut merupakan tampilan gambar dari penggunaan *class ".img-responsive"*



- **".img-rounded"**: ketika ingin membuat gambar dengan sisi melengkung, gunakan *class* ini, maka akan memiliki gambar yang melengkung di bagian sisinya. Berikut merupakan tampilan gambar dari penggunaan *class ".img-rounded"*



- **".img-circle"**: Pada *class* ini, gambar akan tampil dengan bentuk yang melingkar. Berikut merupakan tampilan gambar dari penggunaan *class ".img-circle"*





- **".img-thumbnail"**: *class* ini digunakan untuk membuat gambar *thumbnail* dengan Bootstrap. Berikut merupakan tampilan gambar dari penggunaan *class* **".img-thumbnail"**



### 3.3.3. Membuat Pesan Peringatan (*Alert*) Dengan Bootstrap

Bootstrap menyediakan *class* khusus untuk menampilkan *alert* dengan sangat sederhana dan mudah. Hanya dengan memanggil sebuah *class* di Bootstrap menggunakan satu perintah saja. Agar tampilan *alert* dapat muncul, panggil *class* **"alert"** untuk mendefinisikan tampilan *alert* di Bootstrap. Agar tampilannya lebih bervariasi, dapat menambahkan *class* lain.

Untuk membuat *class* ini yaitu buatlah tag **<div>** dengan isi peringatan yang ingin dibuat. Kemudian di tag pembuka **<div>** buat *class alert* untuk mendeklarasikan *alert*, berikut contohnya:

```

1 <div class="alert alert-success">
2 <strong>Success!</strong> Menunjukkan pemberitahuan sukses.
3 </div>
4
5 <div class="alert alert-info">
6 <strong>Info!</strong> Menunjukkan perubahan atau tindakan informatif.
7 </div>
8
9 <div class="alert alert-warning">
10 <strong>Warning!</strong> Menunjukkan pemberitahuan peringatan yang akan menimbulkan dampak.
11 </div>
12
13 <div class="alert alert-danger">
14 <strong>Danger!</strong> Menunjukkan pemberitahuan berbahaya atau berpotensi negatif.
15 </div>

```

Terdapat beberapa pilihan *class* untuk menampilkan pesan peringatan di Bootstrap, seperti **".alert-success"**, **".alert-info"**, **".alert-warning"**, atau **".alert-danger"**. Kumpulan *class* ini akan membuat tampilan *alert* akan menjadi lebih bervariasi.

- **.alert-success:** *Class* ini dapat digunakan ketika ada *input* yang berhasil. Misalnya ketika pengguna selesai mengisi data dan mengirimkannya. Untuk menggunakan *class* ini cukup masukkan kode `<div class="alert alert-success">`. Berikut merupakan tampilan dari penggunaan *class* ".alert-success"

Success! Menunjukkan pemberitahuan sukses.

- **.alert-info:** *Class* ini dapat digunakan ketika ingin memberikan informasi tambahan kepada pengguna. Misalnya memberikan informasi kegunaan tombol atau fungsi tertentu dan cara penggunaannya. Untuk menggunakan *class* ini cukup masukkan kode `<div class="alert alert-info">`. Berikut merupakan tampilan dari penggunaan *class* ".alert-info"

Info! Menunjukkan perubahan atau tindakan informatif.

- **.alert-warning:** *Class alert box* ini dapat untuk mengindikasikan pesan peringatan dan mungkin butuh perbaikan. *Class* ini dapat digunakan ketika ingin memberikan informasi yang penting. Misalnya ketika terjadi *error* pada suatu konfigurasi atau kesalahan pada saat pengisian data. Untuk menggunakan *class* ini cukup masukkan kode `<div class="alert alert-warning">`. Berikut merupakan tampilan dari penggunaan *class* ".alert-warning".

Warning! Menunjukkan pemberitahuan peringatan yang akan menimbulkan dampak.

- **.alert-danger:** *Class alert box* ini dapat mengindikasikan pesan penting dan kemungkinan berefek buruk. *Class* ini dapat digunakan ketika terjadi sesuatu yang sangat serius. Misalnya ketika terjadi

permasalahan pada server atau konfigurasi keamanan yang gagal. Penggunaan *alert* yang sesuai akan memberikan *user experience* yang menarik. Untuk menggunakan *class* ini cukup masukkan kode **<div class="alert alert-danger">**. Pengguna tidak akan kesulitan menemukan *error* yang terjadi sehingga *problem solving* pun dapat dilakukan dengan cepat. Berikut merupakan tampilan dari penggunaan *class* **"*alert-danger*"**.

**Danger!** Menunjukkan pemberitahuan berbahaya atau berpotensi negatif.

### 3.3.4. Membuat Tombol Dengan Bootstrap

Tombol adalah komponen yang umum dan sering kali digunakan dalam membuat *website*. Inilah yang membuat Bootstrap juga menyediakan *class* untuk membuat tombol yang menarik. Caranya yaitu dengan memanggil salah satu *class* tombol yang akan ditampilkan pada halaman *website*.

Agar dapat menggunakan tombol di dalam *website*, cukup dengan memanggil *class* **"btn"**. Selain itu, dapat menambahkan *class* lain untuk memberikan variasi tombol di dalamnya. Terdapat banyak variasi tombol yang disediakan oleh Bootstrap, seperti **"*btn*"**, **"*btn-default*"**, **"*btn-primary*"**, **"*btn-success*"**, **"*btn-info*"**, **"*btn-warning*"**, **"*btn-danger*"**, dan **"*btn-link*"**. Berikut merupakan salah satu contoh penggunaanya dengan membuat kode seperti di bawah ini:

```
1 <button type="button" class="btn">Basic</button>
2 <button type="button" class="btn btn-default">Default</button>
3 <button type="button" class="btn btn-primary">Primary</button>
4 <button type="button" class="btn btn-success">Success</button>
5 <button type="button" class="btn btn-info">Info</button>
6 <button type="button" class="btn btn-warning">Warning</button>
7 <button type="button" class="btn btn-danger">Danger</button>
8 <button type="button" class="btn btn-link">Link</button>
```

Berikut merupakan tampilan dari kode diatas :



Penjelasan dari setiap kode, yaitu:

- **.btn:** *Class* ini digunakan untuk membuat tombol dengan model standar berwarna abu kehitaman.
- **.btn-default:** *Class* ini digunakan untuk membuat tombol dengan model default berwarna abu.
- **.btn-primary:** *Class* ini digunakan untuk membuat tombol berwarna biru. Biasanya digunakan untuk tombol *login* dan masuk ke dalam menu tertentu.
- **.btn-success:** *Class* ini digunakan untuk membuat tombol ketika berhasil menginputkan sesuatu atau memproses suatu program.
- **.btn-info:** *Class* ini digunakan untuk membuat tombol berwarna biru muda.
- **.btn-warning:** *Class* ini digunakan untuk membuat tombol warning dengan warna kuning.
- **.btn-danger:** *Class* ini digunakan untuk membuat tombol warning berwarna merah tua.
- **.btn-link:** Selain tombol persegi, Bootstrap juga menyediakan tombol untuk link.

### 3.3.5. Membuat Panel Dengan Bootstrap

Contoh terakhir penggunaan Bootstrap adalah membuat panel. Panel adalah sebuah garis yang terbentuk dari garis dan *padding*. Pada Bootstrap, sebuah panel terdiri dari tiga bagian; *header*, *body*, dan *footer*. *Header* berada di bagian atas. *Body* di bagian tengah atau inti. Sedangkan *footer* di bagian bawah. Untuk membuat panel dengan *header*, *body*, dan *footer* harus dideklarasikan terlebih dahulu menggunakan kode seperti di bawah ini.

```
1 <div class="panel panel-default">
2 <div class="panel-heading">Judul konten panel - (Header)</div>
3 <div class="panel-body">isi konten panel - (Body)</div>
4 <div class="panel-footer">konten bagian bawah panel- (Footer)</div>
5 </div>
```

Berikut merupakan tampilan dari kode di atas:

Judul konten panel - (Header)
isi konten panel - (Body)
konten bagian bawah panel- (Footer)

Terdapat banyak variasi panel yang dapat dibuat menggunakan Bootstrap, seperti **“.panel-default”**, **“.panel-primary”**, **“.panel-success”**, **“.panel-info”**, **“.panel-warning”**, dan **“.panel-danger”**. Berikut merupakan salah satu contoh penggunaannya dengan membuat kode seperti di bawah ini:

```
1 <div class="panel panel-success">
2 <div class="panel-heading">Panel dengan class panel-success</div>
3 <div class="panel-body">isi konten panel</div>
4 </div>
```

Selain itu berikut merupakan tampilan dari variasi panel lainnya:

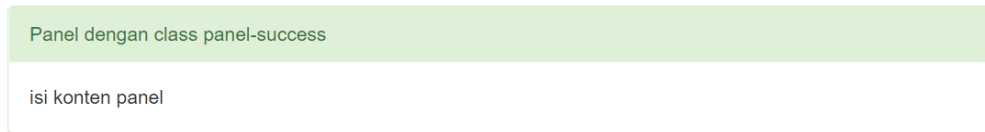
- **.panel-default:** *Class* ini digunakan untuk membuat sebuah panel dengan warna *header* biasa (abu).

Panel dengan class panel-default
Panel Content

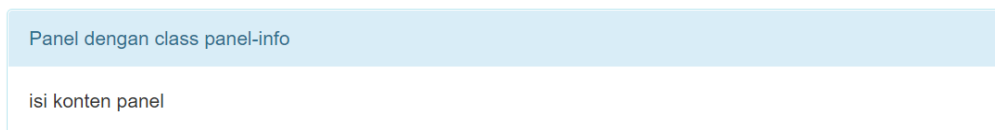
- **.panel-primary:** *Class* ini digunakan untuk membuat panel dengan warna *header* biru tua.

Panel dengan class panel-primary
isi konten panel

- **.panel-success:** *Class* ini digunakan untuk membuat panel dengan warna *header* hijau.



- **.panel-info:** *Class* ini digunakan untuk membuat panel dengan warna *header* biru muda.



- **.panel-warning:** *Class* ini digunakan untuk membuat panel dengan warna *header* kuning muda.



- **.panel-danger:** *Class* ini digunakan untuk membuat panel dengan warna *header* merah muda.



Kelima bagian di atas merupakan beberapa contoh cara menggunakan Bootstrap. Selain itu masih banyak *class* lain yang dapat digunakan pada Bootstrap, seperti *BS Dropdowns Collapse, Tabs/Pills, Navbar, Forms, Carousel, Modal, Tooltip, Popover, Scrollspy*, dan masih banyak lagi yang terdapat pada Bootstrap.

## DAFTAR PUSTAKA

<https://www.w3schools.com/Bootstrap/>

<https://getBootstrap.com/>