

## DAFTAR ISI

<b>HTML .....</b>	<b>1</b>
A.    Teori Singkat .....	1
▪    Pengertian HTML .....	1
▪    Fungsi Web Browser .....	2
▪    Pengertian Tag, Elemen, dan Atribut pada HTML ..	3
B.    Kode Program .....	4
<b>PENGATURAN TEKS .....</b>	<b>8</b>
A.    Teori Singkat .....	8
▪    Menampilkan Tulisan / Teks .....	8
▪    Memasukkan Karakter Khusus ke dalam HTML ....	9
▪    Membuat Judul ( <i>heading</i> ) di HTML .....	10
▪    Membuat Daftar/List di HTML .....	12
▪    Membuat Link, Menyisipkan Gambar, Video .....	16
▪    Menambahkan Gambar Di Html (Tag Image) .....	18
▪    Menyisipkan Video .....	19
<b>TABEL dan FORM .....</b>	<b>20</b>
A.    Tabel .....	20
▪    Pengertian .....	20
▪    Tag .....	20

▪ Atribut.....	21
▪ Contoh Kode Program.....	23
B. Form.....	24
▪ Pengertian.....	24
▪ Elemen pada Form.....	25
▪ Contoh Kode Program.....	28
<b>CSS.....</b>	<b>30</b>
A. CSS.....	30
▪ Teori singkat.....	30
▪ Pendefinisian tag dengan kelas.....	31
▪ Penggunaan atribut ID dalam tag .....	32
▪ Mengatur font dan teks.....	32
▪ Mengatur Tepi.....	33
▪ Memberi Jarak .....	34
▪ Mengatur warna .....	34
▪ Hover .....	35
▪ Contoh Kode Program.....	35
B. Bootstrap.....	37
▪ Pengertian Bootstrap.....	37
▪ Pemanggilan Bootstrap .....	38

▪ Kolom.....	39
▪ Tabel.....	40
▪ Gambar .....	40
▪ Tombol .....	41
▪ Form.....	41
<b>PHP.....</b>	<b>43</b>
A. PHP Dasar .....	43
▪ Penjelasan singkat php .....	43
▪ Variabel.....	44
B. Form.....	45
C. Kondisi/ Percabangan .....	46
▪ Syntax .....	46
▪ Kode Program .....	47
D. Perulangan.....	48
▪ Syntax .....	48
▪ Kode Program .....	49
E. Array .....	49
▪ Array .....	50
▪ Array dan Perulangan .....	50
<b>CRUD.....</b>	<b>52</b>

A.	Pengertian CRUD.....	52
B.	Program dengan CRUD .....	52
▪	Membuat Database dan Tabel .....	52
▪	Membuat file koneksi.....	53
▪	Create/ Insert .....	54
▪	Read/ Select.....	58
▪	Delete.....	59
▪	Update .....	60
	<b>CODEIGNITER .....</b>	<b>67</b>
A.	Kelebihan Codeigniter .....	67
B.	Instalasi Codeigniter .....	69
C.	Pengaturan File Konfigurasi .....	70
▪	config.php .....	70
▪	autoload.php .....	71
D.	Halaman Utama Codeigniter .....	72
E.	Konsep MVC .....	73
▪	View .....	73
▪	Model .....	74
▪	Controller .....	74
F.	Penerapan Konsep MVC Database.....	75

▪	Konfigurasi Database .....	75
▪	Insert .....	76
▪	Select .....	80
▪	Delete .....	84
▪	Update .....	86
G.	File Handling .....	91
▪	Proses Upload pada Insert Data.....	91
▪	Proses Upload pada Update Data .....	95

# HTML

## A. Teori Singkat

### ▪ Pengertian HTML

HTML adalah singkatan dari Hypertext Markup Language. Disebut hypertext karena di dalam HTML sebuah text biasa dapat berfungsi lain, kita dapat membuatnya menjadi link yang dapat berpindah dari satu halaman ke halaman lainnya dengan hanya meng-klik text tersebut. Kemampuan text inilah yang dinamakan hypertext, walaupun pada implementasinya nanti tidak hanya text yang dapat dijadikan link.

Disebut Markup Language karena bahasa HTML menggunakan tanda (mark), untuk menandai bagian-bagian dari text. Misalnya, text yang berada di antara tanda tertentu akan menjadi tebal, dan di antara tanda lainnya akan tampak besar. Tanda ini akan kita kenal di HTML sebagai tag.

HTML merupakan bahasa dasar pembuatan web. Disebut dasar karena dalam membuat web, jika hanya menggunakan HTML, tampilan web kita akan terasa hambar. Terdapat banyak bahasa pemrograman web yang ditujukan untuk memanipulasi kode HTML, seperti JavaScript dan PHP. Namun sebelum anda belajar JavaScript maupun PHP, memahami HTML merupakan hal yang paling awal.

HTML bukan bahasa pemrograman (programming language), tetapi bahasa markup (markup language), hal ini terdengar sedikit aneh, tapi jika anda telah mengenal bahasa pemrograman lain, dalam HTML tidak akan ditemukan struktur yang biasa di temukan dalam bahasa pemrograman seperti IF, LOOP, maupun variabel. HTML hanya sebuah bahasa struktur yang fungsinya untuk menandai bagian-bagian dari sebuah halaman.

Selain HTML, dikenal juga xHTML yang merupakan singkatan dari eXtensible Hypertext Markup Language. xHTML merupakan versi tambahan dari HTML. Untuk tahap ini anda boleh menyebut bahwa xHTML itu juga HTML (Perbedaannya akan kita bahas materi Belajar HTML Lanjutan nanti)

### ▪ **Fungsi Web Browser**

Untuk mempelajari HTML, web browser adalah perangkat utama yang kita butuhkan. Ibarat bahasa pemrograman lainnya, Web Browser adalah *compiler* dan *intrepreter* HTML. Anda bebas menggunakan web browser yang disukai, berikut nama – nama web browser yang ada saat ini:

1. Mozila Firefox
2. Google Chrome
3. Internet Explorer
4. Opera
5. Apple Safari

Setelah menentukan web browser dan berhasil diinstal, selanjutnya membuat kode HTML, kita membutuhkan sebuah aplikasi text editor diataranya bisa menggunakan notepad ++, textpad dll.

## ▪ Pengertian Tag, Elemen, dan Atribut pada HTML

Tag yang ada dalam HTML antara lain

```
<html>
....
</html>
```

Bagian yang terdapat dalam tag HTML terdiri dari:

- Kepala

Bagian kepala dalam dokumen HTML ditandai dengan tag berikut :

```
<head>
...
</head>
```

Bagian kepala ini digunakan untuk membuat judul halaman web dengan menggunakan tag berikut :

```
<title>
...
</title>
```

- Badan

```
<body>
...
```



```
</body>
```

Bagian badan akan berisikan content yang akan ditampilkan dalam halaman web.



Aturan dalam menuliskan Tag-tag HTML yaitu :

- Tag ditulis dengan format diawali tanda < dan diakhiri tanda >, seperti <HTML> .
- Tidak boleh ada spasi setelah tanda <. Contoh : <TITLE> tidak boleh dituliskan dengan < TITLE> .
- Tag boleh ditulis dengan huruf kecil, huruf capital, ataupun kombinasi keduanya.
- Untuk menampilkan teks dihalaman web, dengan mengetikkan teks dalam tag <BODY>...</BODY>, dan untuk mengatur peletakan teks.

## B. Kode Program

1. Ketikkan script berikut pada notepad.

```
<html>
<head>
  <title>Ini      Title      Belajar
Web</title>
</head>
<body>
  Ini adalah paragraf pertama
  Ini adalah paragraf kedua
</body>

</html>
```

Simpan dengan praktek1.html dan jalankan di browser anda dengan mengklik dua kali file tersebut.

## 2. Cobalah script di bawah ini

```
<html>
<head>
<title>Ini Title Belajar Web</title>
</head>
<body>
<p>
  Ini adalah paragraf pertama
</p>
<p>
```

```
    Ini adalah paragraf kedua
</p>
<p>
    Ini adalah paragraf ketiga
</p>
</body>
</html>
```

Simpan dengan nama praktek2.html dan jalankan di browser anda.

3. Selanjutnya ketikkan script di bawah ini:

```
<html>
<head>
<title>Ini Title Belajar Web</title>
</head>
<body>
Dokumen ini yang akan di tampilkan di web
browser anda.

    - Web statis adalah web yang berisi
      informasi informasi yang bersifat
      statis (tetap).

    - Web dinamis adalah web yang
      menampilkan informasi yang bersifat
```

dinamis (berubah - ubah) dan dapat saling berinteraksi dengan user.

- Perhatikan perbedaan antara yang menggunakan tag `br` dan yang tidak menggunakan, selamat mencoba..

```
</body>
```

```
</html>
```

Simpan script dengan nama `praktek3.html`. Dan lihat hasilnya di browser anda.

4. Modifikasi dengan menambahkan tag `<br>` lalu bandingkan antara text yang menggunakan tag `<br>` dan tidak.

# PENGATURAN TEKS

## A. Teori Singkat

### ▪ Menampilkan Tulisan / Teks

Menampilkan teks dihalaman web, dengan mengetikkan teks dalam tag..., dan untuk mengatur peletakan teks, ada beberapa Tag yang digunakan antara lain :

No.	Tag	Fungsi
1.	B	Menebalkan cetakan teks
2.	SMALL	Menampilkan teks lebih kecil dari pada ukuran normal
3.	BIG	Menampilkan teks lebih besar dari pada ukuran normal
4.	U	Menambah garis bawah pada teks
5.	I	Menampilkan teks dengan bentuk tulisan miring
6.	SUB	Menampilkan teks sebagai subskrip

7.	SUP	Menampilkan teks sebagai superskrip
8.	TT	Menampilkan teks seperti ketikan menggunakan mesin
9.	EM	untuk menandai bagian text yang perlu di tekankan
10.	S	untuk membuat text dengan garis tercoret
11.	BLOCKQUOTE	untuk bagian text yang merupakan kutipan panjang
12.	CODE	untuk memasukkan kode ke dalam HTML

#### ▪ **Memasukkan Karakter Khusus ke dalam HTML**

Untuk memasukkan karakter-karakter khusus ke dalam HTML, kita harus men-‘escape’ karakter tersebut. Istilah escape maksudnya bahwa dari pada menulis sebuah karakter secara langsung, kita bisa menggantinya dengan angka atau referensi yang merujuk kepada karakter tersebut.

Dalam HTML, terdapat 2 cara untuk men-‘escape’ karakter, yaitu dengan menggunakan penomoran angka (numeric entity), dan menggunakan singkatan untuk karakter (named entity).

Kedua cara ini ditulis dengan karakter ‘&’ pada awal penulisan, dan diakhiri dengan karakter ‘;’.

Tabel Karakter Khusus dalam HTML

Character	Description	Name	Number
	Character space (nonbreaking space)	&nbsp;	&#160;
&	Ampersand	&amp;	&#038;
'	Apostrophe	&apos;	&#039;
<	Less-than symbol (useful for displaying markup on a web page)	&lt;	&#060;
>	Greater-than symbol (useful for displaying markup on a web page)	&gt;	&#062;
©	Copyright	&copy;	&#169;
®	Registered trademark	&reg;	&#174;
™	Trademark	&trade;	&#8482;
£	Pound	&pound;	&#163;
¥	Yen	&yen;	&#165;
€	Euro	&euro;	&#8364;
—	En-dash	&ndash;	&#8211;
—	Em-dash	&mdash;	&#8212;
'	Left curly single quote	&lquo;	&#8216;
'	Right curly single quote	&rquo;	&#8217;
“	Left curly double quote	&lquo;	&#8220;
”	Right curly double quote	&rdquo;	&#8221;
•	Bullet	&bull;	&#8226;
...	Horizontal ellipsis	&hellip;	&#8230;

## ▪ Membuat Judul (*heading*) di HTML

HTML menyediakan tag khusus untuk membuat judul atau di dalam HTML lebih di kenal dengan istilah: heading. Heading dirancang terpisah dari paragraf. Tag Heading biasanya

digunakan untuk judul dari paragraf, atau bagian dari text yang merupakan judul.

Tag heading di dalam HTML terdiri dari 6 tingkatan, yaitu <h1>, <h2>, <h3>, <h4>, <h5>, dan <h6>. Tag heading secara default akan ditampilkan oleh web browser dengan huruf tebal (bold). Tampilan dari tag header juga dibuat bertingkat. Tag header <h1> memiliki ukuran huruf paling besar, sedangkan <h6> terkecil.

Berikut adalah contoh penggunaan tag heading di dalam HTML:

```
<html>
<head>
<title>Belajar Web Dengan HTML </title>
</head>
<body>
<h1>contoh judul dengan h1 </h1>
<h2>contoh adalah judul dengan h2 </h2>
<h3>contoh adalah judul dengan h3 </h3>
<h4>contoh adalah judul dengan h4 </h4>
<h5>contoh adalah judul dengan h5 </h5>
<h6>contoh adalah judul dengan h6 </h6>
</body>
</html>
```



Save sebagai heading.html lalu jalankan di web browser anda.



## ▪ Membuat Daftar/List di HTML

Dalam HTML, tag list terdiri dari 2 jenis, ordered list (berurutan) dan unordered list (tidak berurutan). Ordered list akan ditampilkan dengan angka atau huruf, sedangkan unordered list dengan bulatan atau kotak.

Ordered list menggunakan tag `<ol>`, dan unordered list menggunakan tag `<ul>`, sedangkan untuk list sendiri menggunakan tag `<li>`. Penjelasan ini akan lebih mudah jika menggunakan contoh.

Berikut adalah contoh kode HTML untuk membuat ordered list menggunakan tag `<ol>`:

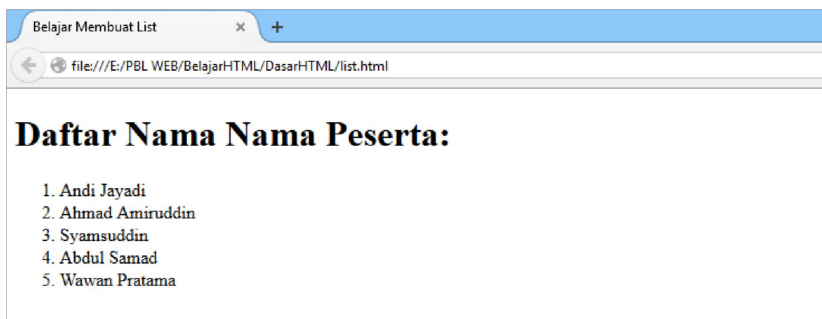
```
<html>
  <head>
    <title>TODO supply a title</title>
```

```

</head>
<body>
  <ol>
    <li>Andi Jayadi</li>
    <li>Ahmad Amirudin</li>
    <li>Syamsudin</li>
    <li>Abdul Samad</li>
    <li>Wawan Pratama</li>
  </ol>
</body>
</html>

```

Selanjutnya Save dengan list.html, lalu jalankan di web browser anda.



Untuk membuat unordered list, tinggal ganti tag <ol> menjadi <ul>. Tag yang digunakan antara lain :

No.	Tag	Fungsi
-----	-----	--------

1.	<code>&lt;UL&gt;....&lt;/UL&gt;</code>	Menampilkan daftar item (list) tanpa penomoran
2.	<code>&lt;LI&gt;</code>	Menambahkan tanda bullet didepan masing-masing item. Tag ini ditambahkan dalam daftar item yang ada didalam tag<UL>
3.	<code>&lt;OL&gt;.....&lt;/OL&gt;</code>	Menampilkan daftar item dengan no,or urut
4.	<code>&lt;DL&gt;....&lt;/DL&gt;</code>	Menampilkan daftar teks yang biasa untuk daftar definisi
5.	<code>&lt;DIR&gt;....&lt;/DIR&gt;</code>	Menampilkan daftar tanpa penomoran untuk menangani direktori
6.	<code>&lt;MENU&gt;.....&lt;MENU&gt;</code>	Menampilkan menu pilihan tanpa penomoran

## B. Praktek

1. Ketiklah script dibawah ini.

```
<html>
<head>
<title>Belajar HTML di LAUWBA</title>
```

```
</head>
<body>
<h1> Saya sedang Belajar HTML di &copy;
LAUWBA.COM&#174; &trade;</h1>
<h2>Pengertian Web Browser</h2>
</p>
<big>Web Browser</big> adalah sebuah
<i>software aplikasi</i>
untuk menerima, menampilkan, dan
menerjemahkan informasi dari world wide
web (wikipedia).
Dan salah satu informasi itu dibuat dalam
format HTML.
</p>
<h3>Nama - nama browser </h3>
<ol>
<li>Mozila Firefox</li>
<li>Chrome</li>
<li>Internet Explorer</li>
<li>Opera</li>
<li>Dan lain-lain</li>
</ol>
<h3>Devlop by&copy; lauwba.com&#174;
&trade;</h3>
```

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

Save dengan nama belajar.html lalu jalankan di web browser Anda.

### ▪ Membuat Link, Menyisipkan Gambar, Video

Tujuan kata Hypertext dari HTML adalah membuat sebuah text yang ketika di-klik akan pindah ke halaman lainnya. HTML menggunakan tag `<a>` untuk keperluan ini. Dimana **href** berisi alamat yang dituju (*href* adalah singkatan dari ***hypertext reference***).

Agar lebih jelas, kita akan lihat menggunakan contoh. Silahkan buka text editor dan buat kode seperti dibawah ini :

```
<html>  
<head>  
<title>Penggunaan Tag Link </title>  
</head>  
<body>  
<h1>Belajar Tag Link</h1>  
<p>Saya sedang belajar HTML di  
<a href="www.lauwba.com">Lauwba</a></p>
```

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

Hasil dari kode HTML diatas, diterjemahkan oleh browser menjadi:



Alamat absolute ditulis dengan lengkap, seperti <http://www.pbl-web.org>, atau jika kita merujuk kepada halaman tertentu, menjadi [http://www.pbl-web.org/nama\\_halaman.html](http://www.pbl-web.org/nama_halaman.html).

Alamat relatif maksudnya adalah relatif kepada file tempat kita memanggil link ini. Seandainya nama file kita adalah `belajar_link.html`, dan dalam folder yang sama terdapat halaman `belajar_list.html`, kita dapat menggunakan `href="belajar_list.html"` untuk membuat link ke halaman `belajar_list.html`.

Jika file tersebut berada di dalam folder `lagi_belajar`, maka alamat relatifnya menjadi `href="lagi_belajar/belajar_list.html"`. Namun bagaimana jika halaman tersebut berada 2 tingkat di

luar folder saat ini? Kita dapat menggunakan href="../../belajar\_list.html", untuk naik 2 folder diatasnya. Atribut tag <a>: target.

Atribut penting lainnya dari tag <a> adalah target. Atribut ini digunakan untuk menentukan apakah link yang dituju terbuka di jendela browser saat ini, atau terbuka di jendela baru.

Secara default, setiap link yang kita tulis akan terbuka pada jendela yang sama (menimpa halaman web saat ini). Tetapi apabila kita ingin halaman tersebut terbuka pada tab baru, gunakan atribut target="\_blank".

Contoh penggunaan tag link <a> dengan atribut target:

```
<a href="lauwba.html" target="_blank">Lauwba</a>
```

## ▪ Menambahkan Gambar Di Html (Tag Image)

Sulit membayangkan sebuah halaman HTML tanpa gambar untuk standard web saat ini. Pada materi belajar HTML cara menambahkan gambar di HTML ini kita akan mempelajari cara penggunaan tag image. Tag Image digunakan untuk menampilkan gambar kedalam halaman web, menggunakan <img>.

Contoh penggunaan tag <img> yaitu dengan penulisan script seperti dibawah ini :

```

```

Jika gambar digunakan sebagai background maka contohnya seperti dibawah ini :

```
<html>
<head>
<title>Penggunaan Tag Image</title>
</head>
<body background="nama_gambar.jpg">
</body>
</html>
```

#### ▪ Menyisipkan Video

```
<html>
  <head>
    <title>TODO supply a title</title>
  </head>
  <body>
    <center></center>
    <br>
    <iframe width="640" height="360"
src="https://www.youtube.com/embed/KJZsy_8Lyws"
frameborder="0" allow="autoplay; encrypted-
media" allowfullscreen></iframe>
  </body>
```



```
</html>
```

## TABEL dan FORM

### A. Tabel

#### ▪ Pengertian

Tabel merupakan cara untuk menampilkan informasi dalam bentuk sel yang terdiri atas baris dan kolom. Tabel dibuat dengan menggunakan tag `<table>`. Sebuah tabel dibagi menjadi baris-baris, dan tiap baris dibagi ke dalam cell-cell.

Tabel biasanya digunakan dalam halaman web untuk memperindah tampilan ataupun untuk mengatur agar informasi dapat disajikan dengan tampilan yang enak dibaca.

#### ▪ Tag

Untuk membuat table dalam halaman HTML, dapat menggunakan tag berikut:

No	Tag	Fungsi
1.	<code>&lt;table&gt; ... &lt;/table&gt;</code>	Membuat table dan sebagai tanda awal dan akhir table
2.	<code>&lt;tr&gt; ... &lt;/tr&gt;</code>	Membuat baris dalam table

3.	<code>&lt;td&gt; ... &lt;/td&gt;</code>	Membuat kolom (sel)
4.	<code>&lt;th&gt; ... &lt;/th&gt;</code>	Membuat judul kolom
5.	<code>&lt;caption&gt;...&lt;/caption&gt;</code>	Membuat judul pada tabel

### ▪ Atribut

Adapun atribut-atribut yang ada dalam tag `<table>` antara lain

No.	Atribut	Fungsi
1.	<code>border</code>	Menampilkan garis table dengan memberikan nilai untuk BORDER
2.	<code>bgcolor</code>	Membuat warna latar belakang table
3.	<code>height</code>	Mengatur tinggi table (tinggi sel)
4.	<code>width</code>	Mengatur lebar sel

5.	cellspacing	Mengatur jarak bagian sel terhadap tepi dalam bingkai table
6.	cellpadding	Mengatur jarak teks terhadap tepi kiri

Atribut-atribut untuk <th> <tr>, dan <td>.

No.	Atribut	Fungsi
1.	rowspan	Digunakan dalam <td>. Atribut ini digunakan untuk menggabungkan beberapa baris sel
2.	colspan	Digunakan dalam <th> atau <td>. Berfungsi untuk menggabungkan beberapa sel dalam satu baris
3.	align	Mengatur peletakan dengan memberikan nilai  LEFT, CENTER, atau RIGHT
4.	valign	Mengatur teks menurut posisi vertical dengan memberikan

		nilai TOP, MIDDLE, atau BOTTOM
--	--	-----------------------------------

## ▪ Contoh Kode Program

```
<html>
  <head>
    <title>
      TODO supply a title
    </title>
  </head>
  <body background="">
    <table border="2" style="width:80%"
bgcolor="grey">
      <tr>
        <th>Nama</th>
        <th>Usia</th>
      </tr>
      <tr>
        <td>Grey</td>
        <td>50</td>
      </tr>
      <tr>
        <td>Thomas</td>
        <td>20</td>
```

```
        </tr>
    </table>
</body>
</html>
```

## B. Form

### ▪ Pengertian

Form merupakan salah satu bentuk halaman web yang digunakan untuk menerima masukan dari pengguna, untuk selanjutnya masukan dari pengguna tersebut diolah menggunakan bahasa pemrograman web, baik secara server side scripting(misalkan PHP, JSP) ataupun client-side scripting (javascript).

Form dapat digunakan untuk berbagai keperluan seperti keperluan login, transaksi penjualan, mengumpulkan informasi atau meminta umpan balik dari pengguna, menawarkan barang/jasa secara on-line dan sebagainya. Form sangat sering kita jumpai dalam satu aplikasi sistem informasi berbasis web.

No.	Atribut	Fungsi
-----	---------	--------

1.	action	Menentukan URL yang akan dijalankan dan menerima semua masukkan dari formulir. Jika ACTION tidak disebutkan, informasi akan dikirim ke URL yang sama dengan Web itu sendiri.
2.	method	Menentukan cara pengiriman informasi, yaitu dengan nilai GET dan POST. GET digunakan jika informasi yang dikirim menjadi satu dengan URL, sedangkan POST digunakan jika informasi dikirim secara terpisah dengan URL

## ▪ Elemen pada Form

### - Input

Tag `<input>` digunakan untuk melakukan pemasukan data dan letaknya berada dalam pasangan tag `<form>` dan `</form>`. Ada bermacam-macam tipe pemasukan data, antara lain berupa kotak teks, kotak password, tombol radio, kotak cek, serta tombol submit dan reset. Tag `<input>` memiliki sejumlah atribut. Atribut-atributnya dapat dilihat pada tabel dibawah:

No.	Atribut	Fungsi
-----	---------	--------

1.	type	Menentukan tipe kotak masukan
2.	name	Menentukan nama data
3.	size	Menentukan ukuran kotak masukan
4.	maxlength	Menentukan jumlah karakter yang dapat dimasukkan dalam kotak isian
5	value	Menentukan nilai awal untuk kotak masukan
6	checked	Mengatur agar kotak cek dalam keadaan terpilih pada keadaan awal

Dan Tipe yang ada dalam atribut type antara lain :

No.	Tipe	Fungsi
1.	text	Sebagai kotak isian untuk memasukkan data
2.	submit	Membuat tombol. Jika tombol diklik maka URL yang disebutkan pada atribut

		ACTION dalam tag <FORM> akan dijalankan
3.	reset	Mengembalikan nilai field teks ke default
4.	password	Membuat teks yang diketikkan diganti dengan tanda *
5.	checkbox	Membuat kotak cek
6.	radio	Membuat tombol radio dalam bentuk 2 pilihan atau lebih yang mana user hanya dapat memilih satu pilihan.

#### - Select

Tag ini digunakan untuk menampilkan beberapa pilihan dengan menggunakan kotak kombo (drop-down). Untuk menampilkan daftar pilihannya dapat menggunakan tag <OPTION> dan menyertakan atribut VALUE untuk menyatakan nilai-nilai pilihan. Sedangkan untuk menyatakan nilai default (nilai bawaan pada saat halaman ditampilkan) dapat menggunakan atribut SELECTED.



- Textarea

Tag ini dibutuhkan untuk masukan teks yang panjang.  
Beberapa atribut yang terdapat dalam tag <TEXTAREA>

No.	Atribut	Fungsi
1.	name	Menentukan nama teksarea
2.	cols	Menentukan lebar teksarea
3.	wrap	Menentukan teks secara otomatis akan dilipat atau tidak pada area teks dengan memberikan nilai Hard, off, atau soft
4.	rows	Menentukan panjang teksarea

- **Contoh Kode Program**

```
<html>
  <head>
    <title>Form</title>
  </head>
  <body>
    <form>
<!--teks-->
      Form input teks:
      <input type="text" name="nama">
```

```

        <br><br>
<!--dropdown-->
        Agama :
        <select name="agama">
            <option value="">Pilih</option>
            <option
value="Islam">Islam</option>
        </select>
        <br><br>
<!--teksarea-->
        Komentar :
        <textarea name="komentar">textarea>
        <br><br>
        <button
type="button">Simpan</button>
        </form>
    </body>
</html>

```

# CSS

## A. CSS

### ▪ Teori singkat

CSS *Cascading Style Sheet* (CSS) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman.

Sama halnya *styles* dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas.

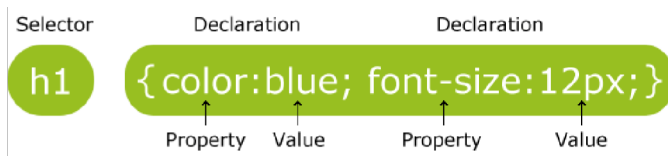
Mengenali kode Ccss:

- Pasangan tag <STYLE> dan </STYLE> digunakan untuk mendefinisikan style sheet menggunakan Ccss. Oleh karena itu, dalam tag <style> terdapat atribut TYPE dengan nilai berupa “teks/CSS”. Pasangan tag tersebut harus diletakkan dalam pasangan tag <HEAD> dan

</HEAD>. Umumnya setelah tag <TITLE> dan </TITLE>.

- Pasangan <!--dan -->merupakan komentar dalam HTML. Biasa digunakan dengan tujuan agar sekiranya browser tidak mendukung atau tidak mengenali style sheet maka yang terdapat dalam pasangan tersebut akan diabaikan, tetapi penulisan pasangan ini tidak diharuskan. Dan yang berada dalam pasangan ini adalah definisi style.

Sintaks Css terdiri dari tiga bagian yaitu selector, properti dan nilai, penulisannya seperti dibawah:



Selektor biasanya adalah elemen HTML / tag yang Anda inginkan untuk mendefinisikan, properti adalah atribut yang ingin Anda ubah, dan setiap properti dapat mengambil nilai. Properti dan nilai/value dipisahkan dengan tanda titik dua, dan dikelilingi oleh kurung kurawal seperti contoh diatas.

#### ▪ Pendefinisian tag dengan kelas

Dengan kelas selektor Anda dapat menetapkan berbagai gaya dalam elemen HTML, misalkan Anda ingin memiliki dua tipe paragraf dalam dokumen Anda, yang pertama paragraf

dengan format rata kanan, dan yang kedua paragraf rata tengah, maka penulisannya adalah sebagai berikut:

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

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

### ▪ **Penggunaan atribut ID dalam tag**

Dalam pasangan tag <STYLE> dan </STYLE>, Anda bisa mendefinisikan suatu nama ID dengan bentuk seperti berikut:

```
#nama_ID{properti_1:nilai_1;.....;properti_n:nilai_n;}
```

Pada pendefinisian ini, awalan # selalu digunakan

Cara memberi nama ID:

1. Dapat mengandung huruf, angka, atau karakter garis bawah
2. Karakter pertama harus berupa huruf atau karakter garis bawah

Contoh: membuat style bernama tebal yang mengatur teks agar dalam keadaan ditebalkan dan warnanya biru.

```
#tebal { font-weight:bold;color:blue;}
```

### ▪ **Mengatur font dan teks**

Berkaitan dengan pengaturan font dan teks, css mendukung sejumlah properti seperti font-size, font-family, text-decoration

- Properti font-size

Properti font-size berguna untuk menentukan ukuran font. Ada empat cara untuk memberikan nilai ke properti ini, yaitu melalui nilai absolut, relatif, ukuran plus satuan, dan presentase.

- Properti font-family

Properti font-family berguna untuk menentukan nama font.

Contoh: font-family: Arial;

Nama font yang disebutkan dalam properti ini bisa lebih dari satu nama. Dalam hal ini, antar nama font perlu dipisahkan oleh koma.

Contoh: font-family: Helvetica, Arial;

Maksud dari penulisan tersebut adalah font yang digunakan Helvetica, tetapi jika font ini tidak ada pada sisi klien maka font Arial yang akan digunakan.

## ▪ Mengatur Tepi

Untuk melakukan pengaturan tebal tepi (margin), bisa menggunakan properti bernama margin. ada dua cara yang bisa dilakukan.

Pertama, dengan memberikan suatu angka diikuti dengan satuan yang dapat berupa em, ex, px, pt, pc, in, mm, cm.

Contoh: margin:18px;

Artinya : semua tepi diatur dengan ketebalan 20 piksel

Contoh : margin : 20px 30px 40px 50px

Artinya : ketebalan tepi atas 20 piksel

Ketebalan tepi kanan 30 piksel

Ketebalan tepi bawah 40 piksel

Ketebalan tepi kiri 20 piksel

#### ▪ **Memberi Jarak**

Untuk memberikan jarak antara tepi dari suatu tag terhadap isi di dalam tag, anda bisa menggunakan properti padding.

Contoh: padding : 20px

Artinya : semua jarak untuk semua sisi sebesar 20 piksel

Contoh : padding: 20px 30px 40px 50px

Artinya : jarak dari tepi atas sebesar 20 piksel

Jarak dari tepi kanan sebesar 30 piksel

Jarak dari tepi bawah sebesar 40 piksel

Jarak dari tepi kiri sebesar 20 piksel

#### ▪ **Mengatur warna**

Fungsi rgb()

Berguna untuk membentuk warna didasarkan pada nilai komponen pembentuk warna. Sebagaimana diketahui, setiap

warna bisa dibentuk dengan mengatur komposisi warna merah, hijau, biru. Bentuk fungsi:

`rgb (komponen_merah, komponen_hijau, komponen_biru);`

Contoh pemanggilan fungsi `rgb()`

- `rgb(255,0,0)` = warna merah
- `rgb(0,255,0)` = warna hijau

### ▪ **Hover**

Fungsi dasar hover adalah untuk merubah tampilan sebuah link pada saat link tersebut disentuh oleh cursor, biasanya sebuah text link akan berubah warna pada saat link tersebut tersentuh oleh cursor. Image Hover memiliki fungsi yang sama yaitu merubah gambar pada saat gambar tersebut disentuh oleh cursor.

### ▪ **Contoh Kode Program**

Nama file : `tag_html.css`

```
/*selector tag html*/  
body  
{  
    background-color: grey;  
}  
h1  
{
```



```
    color: maroon;
    /*text-align: center;*/
}
p
{
    font-family: "Times New Roman";
    font-size: 20px;
    text-align: right;
}
```

Nama file : home.html

```
<html>
  <head>
    <title>Materi CSS</title>
    <!--CSS eksternal-->
    <link href="tag_html.css"
rel="stylesheet">

    <!--CSS Internal-->
    <style type="text/css">

        /*class*/
        .tengah
        {
            text-align: center;
```

```

        }
        /*ID*/
        #kanan
        {
            text-align: right;
        }
    </style>
</head>
<body>
    <h1 class="tengah">Materi CSS</h1>
    <p>Cascading Style Sheet</p>
    <p>Ini merupakan paragraf untuk Css</p>
    <div class="tengah">Pemanggilan Selector
class rata tengah</div>
    <div id="kanan">Selector ID</div>
</body>
</html>

```

## B. Bootstrap

### ▪ Pengertian Bootstrap

Bootstrap adalah sebuah library framework CSS yang dibuat khusus untuk bagian pengembangan front-end website. Bootstrap juga merupakan salah satu framework HTML, CSS dan javascript yang paling populer di kalangan web developer

yang digunakan untuk mengembangkan sebuah website yang responsive. Sehingga halaman website nantinya dapat menyesuaikan sesuai dengan ukuran monitor device (desktop, tablet, ponsel ) yang digunakan pengguna disaat mengakses website dari browser.

## ▪ **Pemanggilan Bootstrap**

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap Example</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/
4.1.3/css/bootstrap.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquer
y/3.3.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4
.1.3/js/bootstrap.min.js"></script>
</head>
<body>
<div class="jumbotron text-center">
  <h1>My First Bootstrap Page</h1>
  <p>Resize this responsive page to see the
```

```
effect!</p>
</div>
</body>
</html>
```

## ▪ Kolom

```
<div class="container">
  <div class="row">
    <div class="col-sm-4">
      <h3>Column 1</h3>
      <p>Lorem ipsum dolor..</p>
      <p>Ut enim ad..</p>
    </div>
    <div class="col-sm-4">
      <h3>Column 2</h3>
      <p>Lorem ipsum dolor..</p>
      <p>Ut enim ad..</p>
    </div>
    <div class="col-sm-4">
      <h3>Column 3</h3>
      <p>Lorem ipsum dolor..</p>
      <p>Ut enim ad..</p>
    </div>
  </div>
</div>
```

## ▪ Tabel

```
<div class="container">
  <h2>Basic Table</h2>
  <table class="table">
    <thead>
      <tr>
        <th>Firstname</th>
        <th>Lastname</th>
        <th>Email</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>John</td>
        <td>Doe</td>
        <td>john@example.com</td>
      </tr>
    </tbody>
  </table>
</div>
```

## ▪ Gambar

```


```

```

```

## ▪ Tombol

```
<button type="button" class="btn">Basic</button>  
<button type="button" class="btn btn-  
primary">Primary</button>  
<button type="button" class="btn btn-  
secondary">Secondary</button>  
<button type="button" class="btn btn-  
success">Success</button>  
<button type="button" class="btn btn-  
info">Info</button>  
<button type="button" class="btn btn-  
warning">Warning</button>  
<button type="button" class="btn btn-  
danger">Danger</button>  
<button type="button" class="btn btn-  
dark">Dark</button>
```

## ▪ Form

```
<form action="#">  
  <div class="form-group">  
    <label for="email">Email address:</label>  
    <input type="email" class="form-  
control" id="email">  
  </div>
```

```
<div class="form-group">
  <label for="pwd">Password:</label>
  <input type="password" class="form-
control" id="pwd">
</div>
<div class="form-group form-check">
  <label class="form-check-label">
    <input class="form-check-
input" type="checkbox"> Remember me
  </label>
</div>
<button type="submit" class="btn btn-
primary">Submit</button>
</form>
```

# PHP

## A. PHP Dasar

- **Penjelasan singkat php**

PHP adalah bahasa pemrograman script server-side yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum (*wikipedia*). PHP dikembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group. Situs resmi PHP beralamat di <http://www.php.net>.

PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

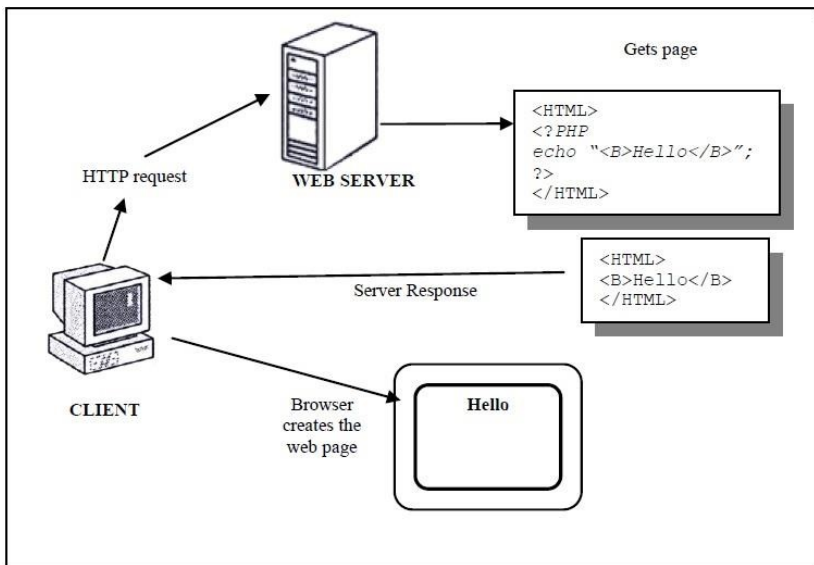
Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya,

PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya



digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll.

Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: Hypertext Preprocessor. Prinsip kerja php jika di gambarkan adalah seperti berikut:



## ▪ Variabel

Variabel diperlukan untuk menyimpan nilai atau informasi pada sistem memori komputer yang nilainya tidak diketahui maupun telah diketahui. Pada PHP penulisan syntax variabel tidak menggunakan pendeklarasian tipe data, cukup dengan

menggunakan tanda \$ saja, maka akan dianggap sebagai variabel.

Ini adalah contoh program dengan variable dalam php.

```
<?php
$a = 4;
$b= 6;
echo $a + $b;
?>
```

## B. Form

Dalam materi kali ini akan diberikan contoh kode program untuk mengambil nilai inputan dari form dengan menggunakan php.

- Nama file : form.php

```
<html>
  <title>Form dalam PHP</title>
  <body>
    <form method="post"
action="proses_form.php">
      Nilai :
      <input type="text" name="nilai">
```

```
        <button  
type="submit">Simpan</button>  
    </form>  
    </body>  
</html>
```

File form.php adalah file yang diakses, dan akan menampilkan form inputan yang mana user akan memasukkan nilai. Dan berikutnya yang harus dibuat adalah file action form nya.

- Nama file : proses\_form.php

File proses\_form.php ini berisi tentang bagaimana syntax php yang digunakan untuk mnegambil nilai dari inputan user.

```
<?php  
$a = $_POST['nilai'];  
echo "Nilai Anda adalah ". $a."<br>";  
?>
```

## C. Kondisi/ Percabangan

Percabangan adalah sebuah logika dalam pemrograman yang memungkinkan adanya lebih dari satu kemungkinan pada kondisi tertntu.

### ▪ Syntax

```
if(kondisi){  
    statement1
```

```
} else {  
    statement2  
}
```

### ▪ Kode Program

Berikut adalah contoh kode perogram yang menggunakan kondisi if.

```
<html>  
    <title>Form dalam PHP</title>  
    <body>  
        <form method="post"  
action="proses_form.php">  
            Nilai :  
            <input type="text" name="nilai">  
            <button  
type="submit">Simpan</button>  
        </form>  
    </body>  
</html>
```

```
<?php  
if($a > 80){  
    echo "Anda Lulus";  
} else if ($a <= 80) {
```

```
        echo "Anda Tidak Lulus";
    } else {
        echo "Nilai Tidak Valid";
    }
?>
```

## D. Perulangan

**Struktur perulangan** (atau dalam *bahasa inggris* disebut dengan **loop**) adalah instruksi program yang bertujuan untuk mengulang beberapa baris perintah. Dalam merancang perulangan kode program, kita setidaknya harus mengetahui 3 komponen, yaitu *kondisi awal dari perulangan, perintah program yang akan diulang, serta kondisi akhir dimana perulangan akan berhenti.*

- **Syntax**

```
//WHILE
$a = nilai awal;
while(kondisi){
    statement;
    perulangan;
}

//FOR
for(nilai awal;kondisi;perulangan){
```

```
        statement
    }
```

#### ▪ Kode Program

```
<?php
//WHILE
$a = 0;
while($a<=100){
    echo "WHILE ".$a."<br>";
    $a++;
}

//FOR
for($a = 0;$a<=100;$a++){
    echo "FOR ".$a."<br>";
}
?>
```

## E. Array

**Array** (atau *larik* dalam bahasa indonesia) bukanlah tipe data dasar seperti **integer** atau **boolean**, **Array** adalah sebuah tipe data bentukan yang terdiri dari kumpulan tipe data lainnya. Menggunakan array akan memudahkan dalam membuat kelompok data, serta menghemat penulisan dan penggunaan variabel.

- **Array**

Berikut adalah contoh kode program dalam array.

```
<?php
//deklarasi array
$a = [];
$a = array();

//contoh
$mobil = array("Toyota",
"Daihatsu","Mitsubishi","Honda");

//pemanggilan
echo $mobil[3];
?>
```

- **Array dan Perulangan**

Untuk mengefisienkan proses pemanggilan array digunakan perulangan.

```
<?php
$mobil = array("Toyota",
"Daihatsu","Mitsubishi","Honda");
$length = count($mobil); //batas
$a = 0;
while ($a < $length){
```

```
        echo $mobil[$a]."<br>";  
        $a++;  
    }  
  
    //perulangan khusus array  
    foreach ($mobil as $value){  
        echo $value."<br>";  
    }  
?>
```



# CRUD

## A. Pengertian CRUD

CRUD merupakan singkatan dari Create, Read, Update dan Delete. Jadi ketika belajar membuat CRUD artinya akan membuat 4 hal tersebut bersamaan dalam satu project.

- **Create** (Membuat, contohnya membuat sistem pendaftaran / membuat data baru di PHP yang dimasukkan ke database).
- **Read** (Membaca, menampilkan data yang ada di database dan dapat dibaca menggunakan pemrograman PHP).
- **Update** (Memperbarui, artinya memperbarui atau mengedit data yang ada di database via PHP).
- **Delete** (Menghapus, cara paling mudah dalam pembuatannya yaitu menghapus data yang ada di database lewat PHP).

## B. Program dengan CRUD

### ▪ Membuat Database dan Tabel

Nama database : db\_lauwba

Nama tabel : tb\_galeri

FIELD	TYPE(Length)	KET
-------	--------------	-----

id_galeri	INT	Auto Increament, Primary
judul_foto	VARCHAR(225)	
foto	Text	
ket_foto	Text	

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	id_galeri	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	judul_foto	varchar(225)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 3	foto	text	latin1_swedish_ci		No	None		
<input type="checkbox"/> 4	ket_foto	text	latin1_swedish_ci		No	None		

☐ Check all *With selected:* Browse Change Drop Primary Unique Ind

Print Preview table structure Track table Move columns Improve table structure

## ▪ Membuat file koneksi

Nama file : koneksi.php






```
<?php

//$cnn = new
//mysqli('host','username','pass','database');
$cnn = new
mysqli('localhost','root','','db_lauwba');

?>
```

## ▪ Create/ Insert

- Membuat folder untuk menyimpan data gambar galeri.

Name	Date modified	Type	Size
 foto_galeri	29/11/2018 6:32	File folder	
 galeri.php	24/11/2018 15:17	PHP File	20 KB
 galeri_form.php	24/11/2018 14:29	PHP File	19 KB
 galeri_proses_insert.php	24/11/2018 14:53	PHP File	1 KB
 koneksi.php	24/11/2018 14:25	PHP File	1 KB

- Modifikasi script pada inputan form (galeri\_insert.php), tambahkan atribut action dan method pada tag form.
- Pastikan semua element form sudah memiliki atribut name.
- Ubah atribut type pada button menjadi submit.

```
<form method="post"
action="galeri_proses_insert.php"
enctype="multipart/form-data">

    <div class="form-group">
    <label class="col-md-12">Judul</label>
    <div class="col-md-12">
        <input name="in_judul" type="text"
class="form-control form-control-line">
```

```

    </div>
</div>

<div class="form-group">
<label class="col-md-12">Gambar</label>
<div class="col-md-12">
    <input name="in_foto"
type="file" class="form-control form-control-
line">
</div>
</div>

<div class="form-group">
<label class="col-md-12">Keterangan</label>
<div class="col-md-12">
    <textarea name="in_ket" rows="5"
class="form-control form-control-
line"></textarea>
</div>
</div>

<button class="btn btn-success"
type="submit"> Simpan</button>

```

```
</form>
```

- Buat file baru dengan nama galeri\_proses\_insert.php untuk proses query insert ke database.

```
<?php
//membaca file koneksi
include "koneksi.php";

//mengambil nilai input dari form
$judul = $_POST['in_judul'];
$ket = $_POST['in_ket'];

//mengambil nilai gambar
$foto = $_FILES['in_foto']['tmp_name'];

//membuat nama baru untuk gambar
$nama = time() . ".png";

//proses upload foto
move_uploaded_file($foto, "foto_galeri/" .
$nama);

//proses insert ke database
$sql = "INSERT INTO
tb_galeri(judul_foto,ket_foto,foto)
```

```
VALUES('$judul','$ket','$nama');"
mysqli_query($cnn, $sql);

header("location: galeri.php");
```

- Running program.

Judul

Logo

Gambar

Choose File logos.png

Keterangan

Sample foto untuk galeri

Simpan

- Pastikan saat insert data pada tabel galeri akan bertambah.

Options

	id_galeri	judul_foto	foto	ket_foto
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Logo	1544599844.png	Sample foto untuk galeri

☐ Check all
 With selected:
 ☐ Edit
 ☐ Copy
 ☐ Delete
 ☐ Export

- Dan pada folder foto\_galeri terdapat file gambar yang diupload

## ▪ Read/ Select

- Modifikasi scrip pada galeri.php dan tambahkan script query select menjadi seperti berikut:

```
<tbody>
  <?php
include "koneksi.php";
$no = 1;

$sql = mysqli_query($cnn, "Select * from
tb_galeri");

while ($row = mysqli_fetch_array($sql)) { ?>

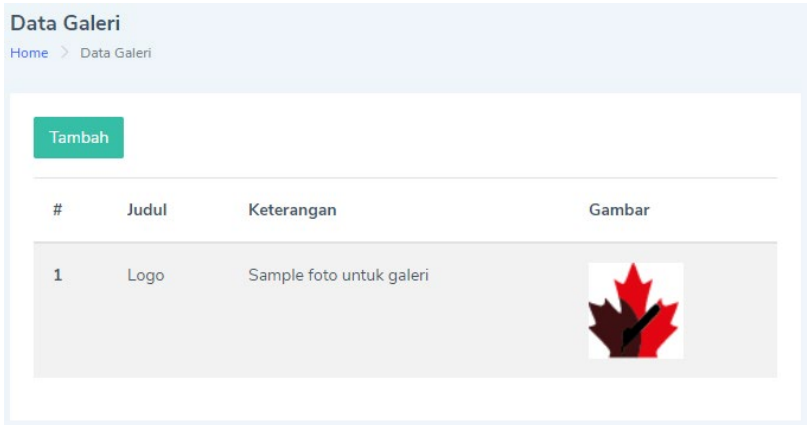
<tr>
<th scope="row"><?php echo $no++; ?></th>
<td><?php echo $row['judul_foto'];?></td>
<td><?php echo $row['ket_foto'];?></td>
<td>

</td>
</tr>

<?php } ?>
```

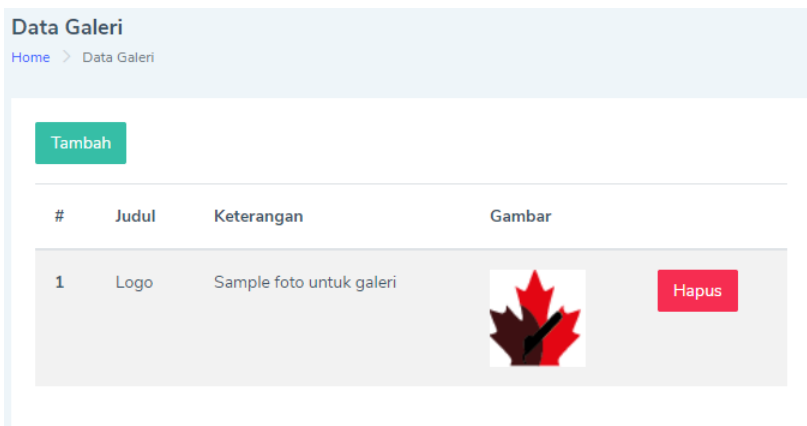
</tbody>

- Pastikan tidak ada error dan data dari tabel galeri akan tampil.



## ▪ Delete

- Tambahkan link hapus pada file galeri.php.



- Script link nya seperti berikut:



```

<a class="btn btn-danger"
href="galeri_proses_hapus.php?idgbr=<?php echo
$row['id_galeri']; ?>" onclick="return
confirm('Apakah anda yakin akan menghapus?')">
Hapus
</a>

```

- Membuat file baru untuk proses query hapus ke database.

Nama file : galeri\_proses\_hapus.php

```

<?php
include "koneksi.php";
$var_id = $_GET['idgbr'];

$query = "DELETE FROM tb_galeri WHERE id_galeri
= $var_id";

mysqli_query($cnn, $query);
header("location: galeri.php");
?>

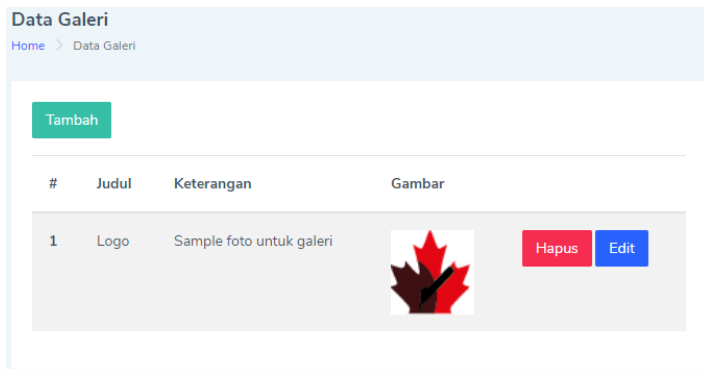
```

## ▪ Update

Pada manajemen data edit terdapat 2 proses query yaitu select data berdasarkan data yang dipilih lalu berikutnya adalah proses query update nya.

a. Select berdasarkan data yang dipilih

- Tambah link Edit pada file galeri.php untuk menuju ke form edit data galeri.



- Script yang ditambahkan adalah seperti berikut:

```
<a class="btn btn-info"
href="galeri_edit.php?idgbr=<?php echo
$row['id_galeri']; ?>">
    Edit
</a>
```

- Menampilkan data berdasarkan baris yang dipilih dengan membuat file baru untuk menampilkan form tapi berisi data dari database. Tampilan form ini sama persis dengan tampilan form pada galeri\_insert.php, maka bisa di copykan saja lalu di modifikasi dengan ditambahkan beberapa script.  
Nama file : galeri\_edit.php

```

<?php
include "koneksi.php";
$id = $_GET['idgbr'];
$sql = mysqli_query($cnn, "SELECT * FROM
tb_galeri where id_galeri=$id");
while($row = mysqli_fetch_array($sql)){
?>

<form method="post"
action="galeri_proses_edit.php"
enctype="multipart/form-data">

<input type="hidden" name="id"
value="<?php echo $row['id_galeri']?>">

<div class="form-group">
<label>Judul Gambar</label>
<input type="text" name="in_judul"
value="<?php echo $row['judul_foto']?>"
class="form-control" placeholder="Enter
text">
</div>

<div class="form-group">

```

```

<label> File Gambar</label>
<input type="file" name="in_foto"
class="form-control" placeholder="Enter
text">
<p></p>
</div>

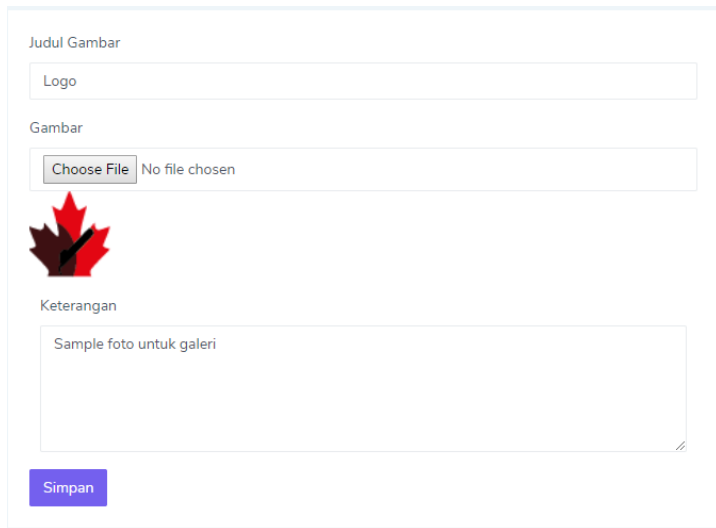
<div class="form-group">
<label class="col-md-
12">Keterangan</label>
<div class="col-md-12">
<textarea name="in_ket" rows="5"
class="form-control form-control-
line"><?php echo $row['ket_foto']
?></textarea>
</div>
</div>

<button type="submit" class="btn btn-
primary">Simpan</button>

</form>
<?php } ?>

```

- Pastikan tidak ada error dan tampilan menjadi seperti berikut ketika tombol Edit dipilih.




Judul Gambar

Logo

Gambar

Choose File No file chosen



Keterangan

Sample foto untuk galeri

Simpan

b. Update data

- Membuat file baru untuk proses Update data.  
Nama file : galeri\_proses\_edit.php

```
<?php
include "koneksi.php";

$judul = $_POST['in_judul'];
$ket = $_POST['in_ket'];
$var_id = $_POST['id'];

//mengambil nilai gambar
```

```
$foto = $_FILES['in_foto']['tmp_name'];

//membuat nama baru untuk gambar
$nama = time() . ".png";

//proses upload foto

if (!empty($foto)) {

move_uploaded_file($foto, "foto_galeri/" .
$nama);

$query = "UPDATE tb_galeri SET judul_foto=
'$judul', ket_foto='$ket', foto='$nama'
WHERE id_galeri = $var_id";

} else {

$query = "UPDATE tb_galeri SET
judul_foto='$judul', ket_foto='$ket' WHERE
id_galeri = $var_id";

}
```

```
mysqli_query($conn, $query);  
header("location: galeri.php");  
  
?>
```

- Pastikan tidak ada error dan data berhasil di edit.

# CODEIGNITER

CodeIgniter adalah sebuah web application network yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis.

CodeIgniter menjadi sebuah framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai framework pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh CodeIgniter, pembuat PHP Rasmus Lerdorf memuji CodeIgniter di frOSCon (Agustus 2008) dengan mengatakan bahwa dia menyukai CodeIgniter karena “it is faster, lighter and the least like a framework.”

## A. Kelebihan Codeigniter

- Gratis, ringan dan cepat

CodeIgniter dilisensikan di bawah lisensi MIT sehingga Anda dapat menggunakannya sesuka hati Anda. Framework ini benar-benar ringan. Sistem inti hanya memerlukan beberapa library yang sangat kecil. Hal ini



kontras dengan banyak framework yang membutuhkan lebih banyak sumber daya secara signifikan. Library tambahan dimuat secara dinamis atau sesuai permintaan, berdasarkan kebutuhan Anda untuk proses tertentu, sehingga sistem dasar sangat ramping dan cukup cepat.

- Menggunakan MVC

CodeIgniter menggunakan pendekatan Model-View-Controller, yang memungkinkan pemisahan antara logika dan presentasi. Ini sangat baik untuk proyek yang dimana desainer bekerja dengan file template Anda, sehingga kode di file ini menjadi minimal

- Menghasilkan URL yang bersih

URL yang dihasilkan oleh CodeIgniter bersih dan search-engine friendly. Daripada menggunakan standar “*query string*” pendekatan ke URL yang identik dengan sistem dinamis, CodeIgniter menggunakan pendekatan *segment-based*:

`example.com/news/article/345`

Catatan: Secara default file `index.php` disertakan dalam URL tetapi bisa dihapus menggunakan file `.htaccess` yang sederhana.

- Packs a Punch

CodeIgniter dilengkapi dengan library yang lengkap yang umumnya paling diperlukan untuk tugas pengembangan

web, seperti mengakses database, mengirim email, memvalidasi data form, menjaga session, memanipulasi gambar, bekerja dengan XML-RPC data dan masih banyak lagi.

- Extensible

Sistem ini dapat dengan mudah diperluas dengan menggunakan library Anda sendiri, helper, atau melalui class extensions atau system hooks.

- Didokumentasikan secara menyeluruh

## B. Instalasi Codeigniter

Untuk memiliki Framework Codeigniter harus download melalui

<http://codeigniter.com/download.php/>

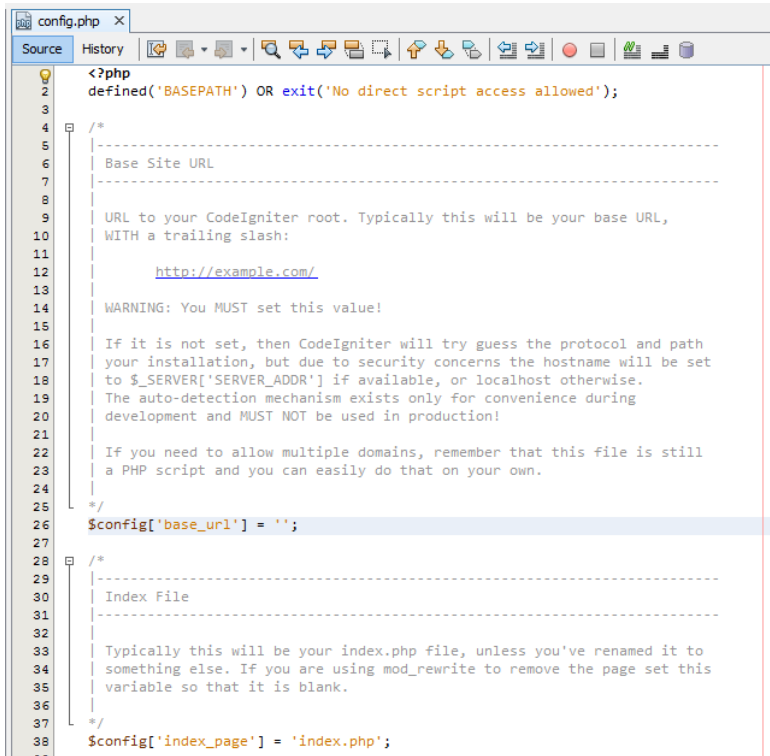
File hasil download tersebut diekstrak di folder Htdocs dan Xampp dan beri nama folder tersebut Codeigniter seperti gambar berikut ini.

PC > New Volume (D:) > htdocs > CodeIgniter			
Name	Date modified	Type	Size
application	12/06/2018 6:45	File folder	
system	12/06/2018 6:45	File folder	
tests	12/06/2018 6:45	File folder	
user_guide	12/06/2018 6:45	File folder	
.editorconfig	12/06/2018 6:45	EDITORCONFIG File	1 KB
.gitignore	12/06/2018 6:45	GITIGNORE File	1 KB
composer.json	12/06/2018 6:45	JSON File	1 KB
contributing.md	12/06/2018 6:45	MD File	7 KB
index.php	12/06/2018 6:45	PHP File	11 KB
license	12/06/2018 6:45	Text Document	2 KB
readme.rst	12/06/2018 6:45	RST File	3 KB

## C. Pengaturan File Konfigurasi

- **config.php**

File config.php berada pada folder application/config. Berikut adalah tampilan source code dari file config.php.



```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
|-----
| Base Site URL
|-----
|
| URL to your CodeIgniter root. Typically this will be your base URL,
| WITH a trailing slash:
|
| http://example.com/
|
| WARNING: You MUST set this value!
|
| If it is not set, then CodeIgniter will try guess the protocol and path
| your installation, but due to security concerns the hostname will be set
| to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
| The auto-detection mechanism exists only for convenience during
| development and MUST NOT be used in production!
|
| If you need to allow multiple domains, remember that this file is still
| a PHP script and you can easily do that on your own.
|-----
*/
$config['base_url'] = '';

/*
|-----
| Index File
|-----
|
| Typically this will be your index.php file, unless you've renamed it to
| something else. If you are using mod_rewrite to remove the page set this
| variable so that it is blank.
|-----
*/
$config['index_page'] = 'index.php';
```

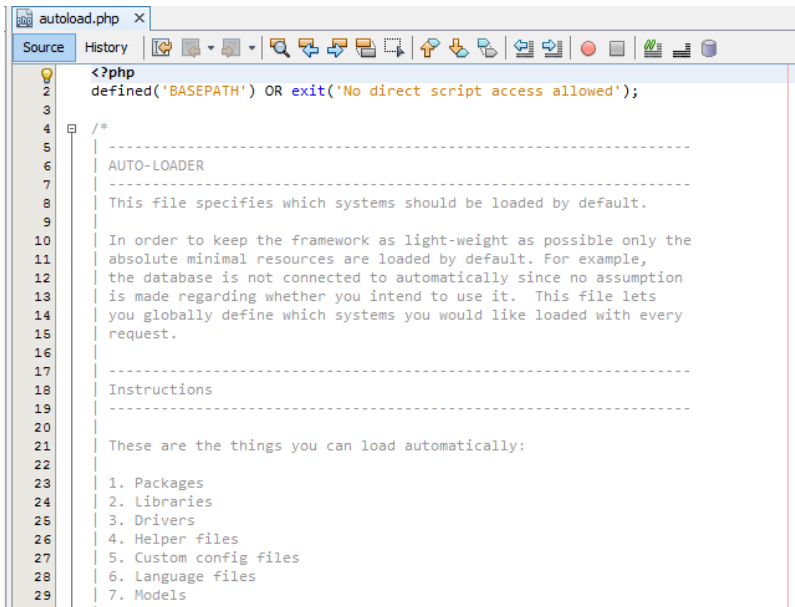
Cari bagian `$config['base_url'] = ''`. Pada gambar diatas berada pada baris ke 26 di dalam tanda kutip kosong digantikan dnegan url project kita.

```
$config['base_url'] = 'http://localhost/Codeigniter/';
```

Lalu simpan perubahan file. Base URL adalah alamat default dari website yang dibuat, secara default URL tersebut akan selalu digunakan untuk pembuatan link du halaman website.

### ▪ **autoload.php**

File autoload.php terletak di folder application/config. File ini berisi konfigurasi bagian apa saja dari framework Codeigniter yang dijalankan secara otomatis saat website pertama kali diakses. Yang diotomatiskan dapat berupa librant(kode pustaka), helper, lugin, model (file akses ke database) dan setting bahasa. Berikut adalah contoh tampilan file autoload.php:



```
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 /*
5 |-----
6 | AUTO-LOADER
7 |-----
8 | This file specifies which systems should be loaded by default.
9 |
10 | In order to keep the framework as light-weight as possible only the
11 | absolute minimal resources are loaded by default. For example,
12 | the database is not connected to automatically since no assumption
13 | is made regarding whether you intend to use it. This file lets
14 | you globally define which systems you would like loaded with every
15 | request.
16 |
17 |-----
18 | Instructions
19 |-----
20 |
21 | These are the things you can load automatically:
22 |
23 | 1. Packages
24 | 2. Libraries
25 | 3. Drivers
26 | 4. Helper files
27 | 5. Custom config files
28 | 6. Language files
29 | 7. Models
--
```

Komponen yang akan kita isikan adalah:

#### a. Library

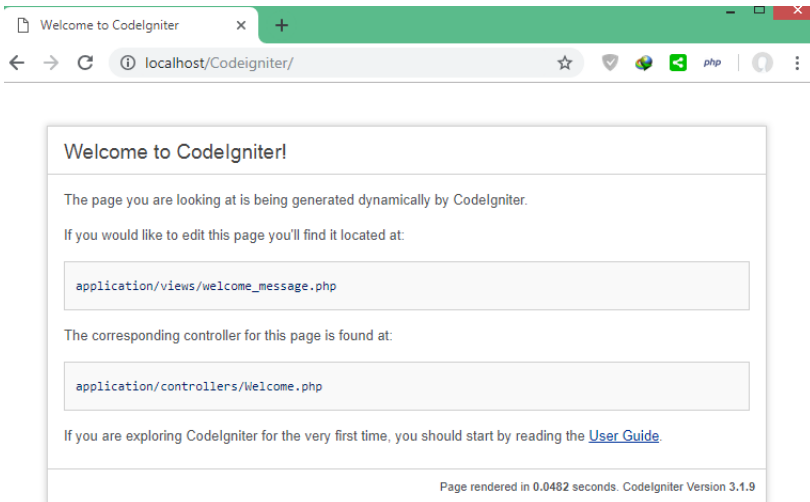
```
$autoload['libraries'] =  
array('database','session','form_validation','uplo  
ad');
```

#### b. Helper

```
$autoload['helper'] = array('url','file','form');
```

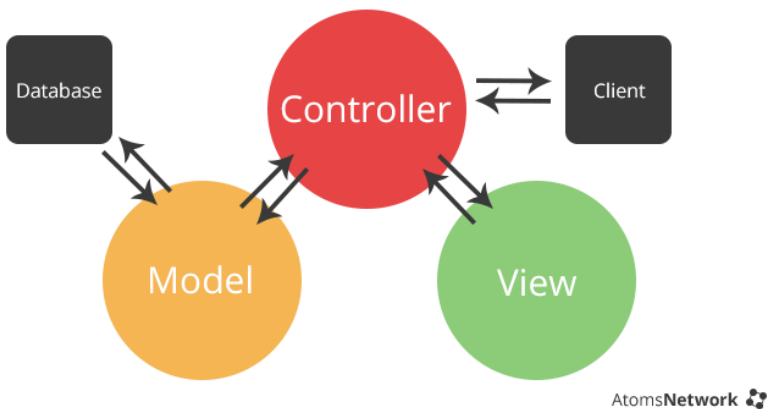
## D. Halaman Utama Codeigniter

Setelah menyelesaikan konfigurasi maka untuk menjalankan project web ini harus mengakses melalui browser dengan url <http://localhost/Codeigniter/>, dan akan tampil halaman utama dari Framework Codeigniter yang telah berhasil diinstall seperti berikut:



## E. Konsep MVC

MVC adalah sebuah pola pemrograman yang bertujuan memisahkan logika bisnis, logika data dan logika tampilan atau interface, atau secara sederhana memisahkan antara proses, data dan tampilan. MVC mengatur arsitektur sebuah aplikasi. Umumnya aplikasi yang dibangun dengan konsep MVC adalah aplikasi yang cukup besar, karena salah satu keuntungan dari MVC itu adalah kemudahan maintenance dan pengembangan aplikasi tersebut. Codeigniter menggunakan konsep ini, Anda harus memisahkan kode database ke folder model, kode proses ke folder Controller dan kode tampilan ke folder View. Alurnya seperti berikut :



### ▪ View

View adalah komponen dari MVC yang bertugas menampilkan apa yang harus ditampilkan ke pengunjung

website. Isinya dapat berupa form, table, gambar, animasi ataupun lainnya yang boleh dilihat oleh user atau pengunjung. Jadi, view mengatur bagaimana suatu data yang diperoleh dari controller ditampilkan untuk user. View mencakup semua proses yang terkait layout output. Tempat menaruh template interface website atau aplikasi. View merupakan informasi yang ditampilkan kepada pengunjung dari website.

- **Model**

Model adalah komponen MVC yang bertugas mengambil data dari database dan memasukkan data ke database. Isi utamanya berupa perintah SQL. Hasilnya dikirimkan ke Controller.

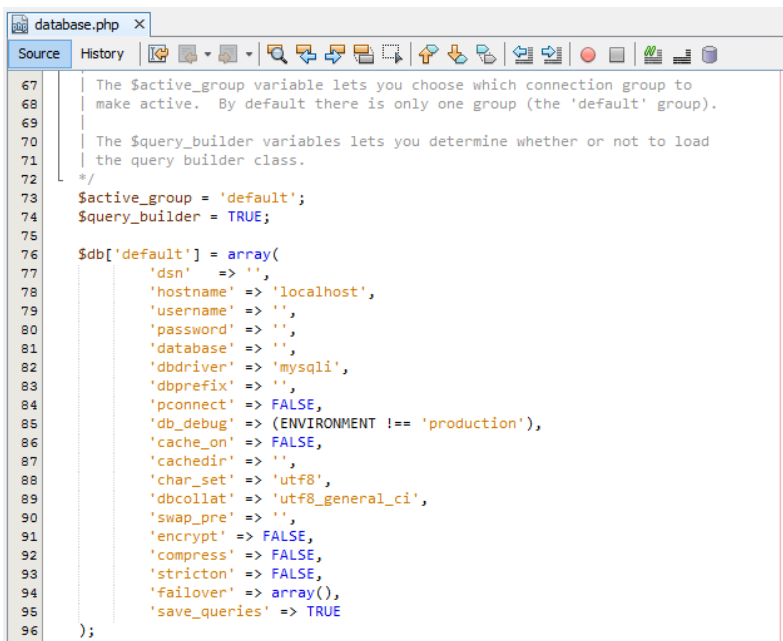
- **Controller**

Controller adalah komponen MVC yang bertugas mengirim perintah ke model untuk mendapatkan data yang diinginkan. Controller tidak mengetahui bagaimana data tersebut diambil dari database, karena Controller tidak berisi kode perintah SQL. Karena itu adalah tugas modal. Controller mengolah data dari inputan user dan data dari modal kemudian data olahan tersebut dikirimkan ke view untuk ditampilkan sesuai aturan controller. Controller merupakan penghubung antara model dan view dan mengatur hubungan tersebut.

## F. Penerapan Konsep MVC Database

### ▪ Konfigurasi Database

Konfigurasi database pada Framework Codeigniter ditambahkan pada file database.php yang terletak di folder application/config. Tampilan utama file database.php adalah sebagai berikut:



```
67 | The $active_group variable lets you choose which connection group to
68 | make active. By default there is only one group (the 'default' group).
69 |
70 | The $query_builder variables lets you determine whether or not to load
71 | the query builder class.
72 |
73 | $active_group = 'default';
74 | $query_builder = TRUE;
75 |
76 | $db['default'] = array(
77 |     'dsn' => '',
78 |     'hostname' => 'localhost',
79 |     'username' => '',
80 |     'password' => '',
81 |     'database' => '',
82 |     'dbdriver' => 'mysqli',
83 |     'dbprefix' => '',
84 |     'pconnect' => FALSE,
85 |     'db_debug' => (ENVIRONMENT !== 'production'),
86 |     'cache_on' => FALSE,
87 |     'cachedir' => '',
88 |     'char_set' => 'utf8',
89 |     'dbcollat' => 'utf8_general_ci',
90 |     'swap_pre' => '',
91 |     'encrypt' => FALSE,
92 |     'compress' => FALSE,
93 |     'stricton' => FALSE,
94 |     'failover' => array(),
95 |     'save_queries' => TRUE
96 | );
```

Dengan menggunakan database yang sudah dibuat pada project php native sebelumnya maka pada tampilan diatas pada line 79, 80, 81 diubah menjadi :

```
'username' => 'root',
'password' => '',
```



```
'database' => 'db_lauwba',
```

## ▪ Insert

Untuk melakukan proses insert data maka file pertama yang dibuat adalah query ke database yaitu membuat file model untuk query insert ke tb\_galeri. Nama file adalah Galeri\_m.php dan harus disimpan ke dalam folder application/models dan membuat class untuk model seperti berikut:

```
<?php
class Galeri_m extends CI_Model{

}
```

Setelah class dibuat lalu tambahkan sebuah fungsi untuk query insert.

```
function insert_db($data) {
    $this->db->insert('tb_galeri', $data);
}
```

Setiap penggunaan file model harus didaftarkan ke konfigurasi autoload.php. Tambahkan nama file model yang dibuat ke dalam indeks model pada file autoload.php.

```
$autoload['model'] = array('Galeri_m');
```

Setelah membuat model yang berikutnya dibuat adalah view. Pada file view kali ini diperlukan form inputan untuk user, maka kodenya adalah dibawah ini. File view ini akan diberi nama galeri\_form\_v.php dan disimpan pada folder application/views.

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Input Galeri</title>
        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/
3.3.7/css/bootstrap.min.css">
        <script
src="https://ajax.googleapis.com/ajax/libs/jquer
y/3.3.1/jquery.min.js"></script>
        <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3
.3.7/js/bootstrap.min.js"></script>
    </head>
    <body>
        <div class="container">

<p><a class="btn btn-info" href="<?php echo
site_url('Galeri');?>">Data Galeri</a></p>

<form method="post" action="">

<div class="form-group">

```

```
<label for="usr">Judul:</label>
<input type="text" class="form-control"
name="in_judul">
</div>

<div class="form-group">
<label for="comment">Keterangan Foto:</label>
<textarea class="form-control" rows="5"
name="in_ket"></textarea>
</div>

<button class="btn btn-success"
type="submit">Simpan</button>

</form>

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

Setelah website memiliki form maka proses insert nya adalah membuat file controller untuk menampilkan halaman view dan melakukan proses insert data ke database.

Pertama membuat file controller dengan nama Galeri.php dan membuat class seperti berikut:

```
<?php

class Galeri extends CI_Controller{

}
```

Di dalam class Galeri.php tambahkan fungsi baru untuk proses menampilkan halaman view yang menampilkan form untuk inputan user.

```
function form() {
    $this->load->view('galeri_form_v');
}
```

Setelah memanggil view, hal yang dilakukan berikutnya adlaah membuat fungsi lagi untuk proses insert data ke database yaitu melakukan pemanggilan terhadap model.

```
function insert() {
    $data = array(
        'judul_foto' => $this->input-
>post('in_judul'),
        'ket_foto' => $this->input->post('in_ket')
    );

    $this->Galeri_m->insert_db($data);
    redirect('Galeri/form');
}
```

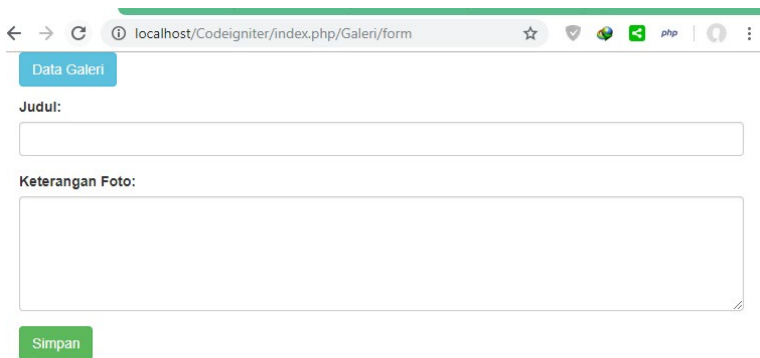
Modifikasi kode pada file galeri\_form\_v.php untuk mengisi action pada form.

```
<form method="post" action="<?php echo  
site_url('Galeri/insert');?>">
```

Running program dengan url :

<http://localhost/Codeigniter/index.php/Galeri/form/>

Dan pastikan tidak ada error dan tampilannya seperti berikut:



The screenshot shows a web browser window with the address bar displaying 'localhost/Codeigniter/index.php/Galeri/form'. The page content includes a blue button labeled 'Data Galeri'. Below it, there is a label 'Judul:' followed by a text input field. Underneath that is a label 'Keterangan Foto:' followed by a larger text area. At the bottom of the form is a green button labeled 'Simpan'.

## ▪ Select

Proses select adalah proses menampilkan data dari database ke halaman website. Maka dari itu modifikasi pada file Galeri\_m.php pada folder models dan tambahkan sebuah fungsi baru yang berfungsi untuk query select ke database.

```
function select_db() {
```

```
return $this->db->query("Select * FROM  
tb_galeri")->result();  
  
}
```

File yang berikutnya perlu dibuat adalah file view untuk tampilan data. Berikut adalah contoh tampilan sederhana untuk view data, nama file ini adalah galeri\_v.php.

```
<!DOCTYPE html>  
<html>  
    <head>  
        <meta charset="UTF-8">  
        <title></title>  
        <link rel="stylesheet"  
href="https://maxcdn.bootstrapcdn.com/bootstrap/  
3.3.7/css/bootstrap.min.css">  
        <script  
src="https://ajax.googleapis.com/ajax/libs/jquer  
y/3.3.1/jquery.min.js"></script>  
        <script  
src="https://maxcdn.bootstrapcdn.com/bootstrap/3  
.3.7/js/bootstrap.min.js"></script>  
    </head>  
    <body>  
        <div class="container">
```

```
<p><a class="btn btn-success" href="<?php echo  
site_url('Galeri/form'); ?>">Tambah</a></p>
```

```
<div class="table-responsive">  
<table class="table table-striped">
```

```
<thead>  
<tr>  
<th scope="col">#</th>  
<th scope="col">Judul</th>  
<th scope="col">Keterangan</th>  
<th scope="col"></th>  
</tr>  
</thead>
```

```
<tbody>  
<?php  
$galeri = $this->Galeri_m->select_db();  
$no = 1;  
foreach ($galeri as $row) {  
?>  
  
<tr>
```

```

<th scope="row"><?php echo $no++; ?></th>
<td><?php echo $row->judul_foto; ?></td>
<td><?php echo $row->ket_foto; ?></td>
<td></td>
</tr>

<?php } ?>
</tbody>

</table>
</div>

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

```

Setelah file view dan model maka untuk menjalankan semua proses diatas menjadi tugas controller, tambahkan fungsi baru pada controller Galeri.php yang berfungsi untuk meload file view untuk menampilkan data dari database.

```

function index() {
    $this->load->view('galeri_v');
}

```

Pada file routes.php di folder application/config ubah kode pada index default\_controller ke controllers Galeri.php menjadi seperti dibawah ini.



```

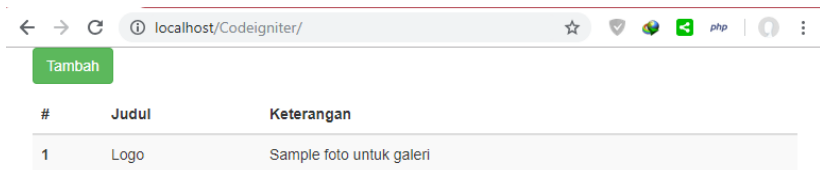
| | my-controller/my-method -> my_controller/my_method
| */
$route['default_controller'] = 'Galeri';
$route['404_override'] = '';
$route['translate_uri_dashes'] = FALSE;

```

Untuk running program akses url:

<http://localhost/Codeigniter/>

Pastikan tidak ada error dan tampilan contoh program akan seperti berikut:



#	Judul	Keterangan
1	Logo	Sample foto untuk galeri

## ▪ Delete

Proses hapus data mrnggunakan syntax SQL Delete, seperti porses sebelumnya kita perlu membuat fungsi untuk query hapus ke dalam class model Galer\_i\_m.php.

```

function delete_db($id) {
    $this->db->where('id_galeri', $id);
    $this->db->delete('tb_galeri');
}

```

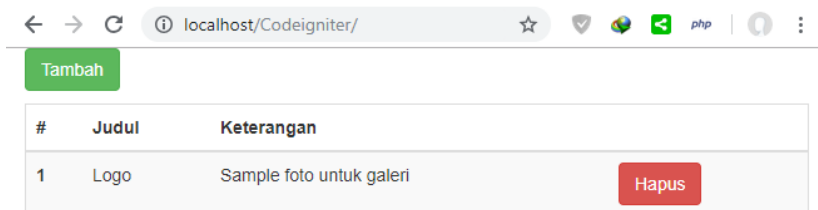
Setelah kode untuk query sudah dibuat maka hal berikutnya adalah membuat fungsi di controller Galeri.php untuk menjalankan proses hapusnya.

```
function delete($id) {  
    $this->Galeri_m->delete_db($id);  
    redirect('Galeri');  
}
```

Setelah controller dan model sudah dibuat maka ditambahkan link untuk memanggil controller hapus pada halaman tampilan galeri\_v.php.

```
<a class="btn btn-danger" href="<?php echo  
site_url('Galeri/delete/'.$row->id_galeri); ?>"  
onclick="return confirm('Apakah anda yakin akan  
menghapus?')">  
Hapus  
</a>
```

Dan tampilannya adalah seperti gambar dibawah ini.



Running program dengan mengakses url <http://localhost/Codeigniter/> dan klik tombol Hapus, pastikan data berhasil terhapus dan tidak ada error.

## ▪ Update

Update adalah proses untuk melakukan perubahan pada data yang sudah ada. Proses update biasanya diawali dengan menampilkan data terpilih berdasarkan id-nya.

Dalam file Galeri\_m.php kita perlu menambahkan fungsi untuk menampilkan data berdasarkan id nya.

```
function select_by_db($id) {  
  
    return $this->db->query("Select * FROM tb_galeri  
    WHERE id_galeri='$id'")->result();  
  
}
```

Langkah berikutnya adalah membuat file view untuk tampilan form edit untuk data yang dipilih dengan nama file galeri\_form\_edit\_v.php.

```
<!DOCTYPE html>  
<html>  
    <head>  
        <meta charset="UTF-8">  
        <title>Input Galeri</title>
```

```

        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/
3.3.7/css/bootstrap.min.css">
        <script
src="https://ajax.googleapis.com/ajax/libs/jquer
y/3.3.1/jquery.min.js"></script>
        <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3
.3.7/js/bootstrap.min.js"></script>
    </head>
    <body>
        <div class="container">

<?php
foreach ($galeri as $row) {
?>

<form method="post" action="">

<input type="hidden" name="id" value="<?php echo
$row->id_galeri; ?>">

<div class="form-group">
<label for="usr">Judul:</label>

```

```

<input type="text" class="form-control"
name="in_judul" value="<?php echo $row-
>judul_foto; ?>">
</div>

<div class="form-group">
<label for="comment">Keterangan Foto:</label>
<textarea class="form-control" rows="5"
name="in_ket"><?php echo $row->ket_foto;
?></textarea>
</div>

<button class="btn btn-success"
type="submit">Simpan</button>

</form>

<?php } ?>
    </div>
</body>
</html>

```

Tambahkan fungsi juga pada controller Galeri.php untuk menjalankan prosesnya dengan memanggil model dan viewnya.

```
function select_by($id) {

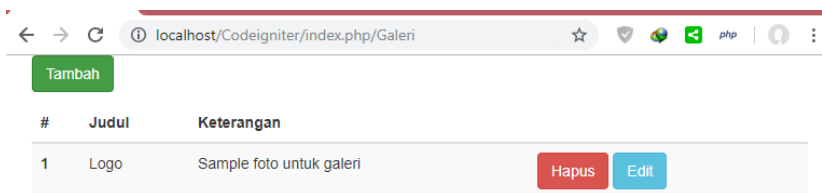
$data['galeri'] = $this->Galeri_m-
>select_by_db($id);
$this->load->view('galeri_form_edit_v', $data);

}
```

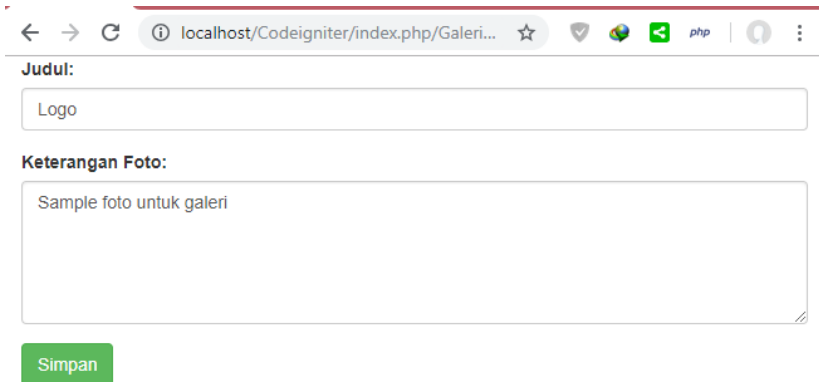
Dan modifikasi kode pada file view galeri\_v.php untuk menambahkan link Edit yang mengarahkan ke class select\_by() yang baru saja dibuat.

```
<a class="btn btn-info" href="<?php echo
site_url('Galeri/select_by/'.$row->id_galeri);
?>">Edit</a>
```

Tampilan pada galeri\_v menjadi seperti berikut:



Dan setelah dipilih link Edit maka akan muncul seperti berikut:



← → ↻ ⓘ localhost/Codeigniter/index.php/Galeri... ☆ 🔒 🌐 php ⓘ ⋮

**Judul:**

Logo

**Keterangan Foto:**

Sample foto untuk galeri

Simpan

Setelah menampilkan data berdasarkan idnya sudah ditambahkan maka selanjutnya adalah membuat proses editnya. Kembali ke file `Galeri_m.php` untuk membuat fungsi query editnya.

```
function edit_db($id, $data) {  
    $this->db->where('id_galeri', $id);  
    $this->db->update('tb_galeri', $data);  
}
```

Controller `Galeri.php` ditambahkan sebuah fungsi untuk menjalankan proses editnya.

```
function edit() {  
  
    $id = $this->input->post('id');  
  
    $data = array(  

```

```

        'judul_foto' => $this->input-
>post('in_judul'),
        'ket_foto' => $this->input->post('in_ket')
    );

    $this->Galeri_m->edit_db($id, $data);
    redirect('Galeri');

}

```

Lalu modifikasi pada tampilan galeri\_form\_edit\_v.php untuk menambahkan kode pada tag form di atribut action.

```

<form method="post" action="<?php echo
site_url('Galeri/edit');?>">

```

Running program dengan mengubah data yang dipilih, pastikan tidak ada error dan data berhasil diubah.

## G. File Handling

Pada materi Framwork Codeigniter sebelumnya proses manajemen data yang dilakukan tidak menggunakan file handling atau proses upload gambar, maka pada materi ini akan dibahas bagaimana mengupload file gambar menggunakan Framework Codeigniter.

- **Proses Upload pada Insert Data**
  - Menentukan Direktori Tujuan



Hal yang pertama kali perlu diperhatikan jika ada proses upload file adalah menentukan direktori tujuan atau tempat dimana file akan disimpan. Kita akan menyimpan data file upload ke dalam folder assets/upload/:

```
htdocs\  
  
    Codeigniter\  
  
        application\  
  
            system\  
  
                assets\  
  
                    upload\
```

- Form Input File

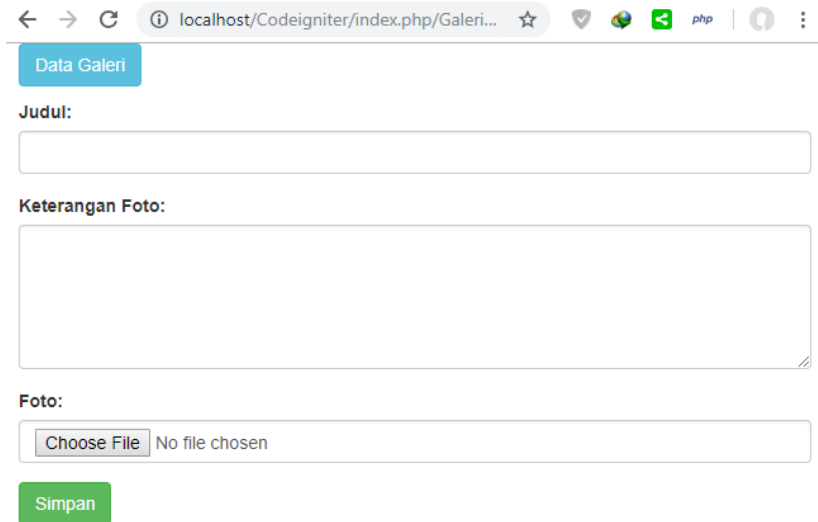
Untuk melakukan upload gambar form yang dibutuhkan pastinya adalah input file, maka dari itu dari program yang sudah dibuat sebelumnya perlu ditambahkan elemen form baru yaitu input file. Tambahkan kode pada file galeri\_form\_v.php.

```
<div class="form-group">  
  <label for="usr">Foto:</label>  
  <input type="file" class="form-control"  
  name="in_foto">  
</div>
```

Dan pastikan tambahkan atribut enctype pada tag form:

```
<form method="post" action="<?php echo  
site_url('Galeri/insert');?>"  
enctype="multipart/form-data">
```

Dari kode yang ditambahkan pastikan untuk tampilannya akan seperti berikut:



The screenshot shows a web browser window with the address bar displaying 'localhost/Codeigniter/index.php/Galeri...'. The page has a blue header bar with the text 'Data Galeri'. Below the header, there is a form with the following elements:

- A label 'Judul:' followed by a text input field.
- A label 'Keterangan Foto:' followed by a large text area.
- A label 'Foto:' followed by a file upload button labeled 'Choose File' and the text 'No file chosen'.
- A green button labeled 'Simpan' at the bottom.

### • Konfigurasi Upload

Setelah file form input dibuat maka modifikasi pada fungsi insert untuk konfigurasi dan proses upload file gambar. Ubah fungsi insert() pada controller Galeri.php menjadi seperti ini:

```
function insert() {  
    $nm_file = time() . '.png';  
    $config['upload_path'] = './assets/upload/';
```

```

$config['allowed_types'] = 'jpg|jpeg|png';
$config['file_name'] = $nm_file;
$config['overwrite'] = TRUE;
$this->upload->initialize($config);
if ($this->upload->do_upload('in_foto')) {
    $foto = $this->upload->data();
    $data = array(
        'judul_foto' => $this->input-
>post('in_judul'),
        'ket_foto' => $this->input->post('in_ket'),
        'foto' => $foto['file_name']
    );
    $this->Galeri_m->insert_db($data);
} else {
    $error = array(
        'error' => $this->upload->display_errors()
    );
    echo json_encode($error);
}
redirect('Galeri');
}

```

- View Data Upload

Setelah konfigurasi selesai maka berikutnya adalah menampilkan hasil file yang sudah diupload, modifikasi tampilan pada file galeri\_v.php untuk membaca file gambar:

```
<td>

</td>
```

Running Program, coba upload sebuah file, pastikan untuk tidak terjadi error dan hasil tampilan akan seperti berikut pada galeri\_v.php:



## ▪ Proses Upload pada Update Data

- Form Input File Update

Sama seperti pada proses insert untuk membuat file upload pada proses edit juga perlu ditambahkan file inputan untuk file dan tambahkan kode pada file view galeri\_form\_edit\_v.php dan juga atribut tambahan pada tag form.

```
<div class="form-group">
<label for="usr">Foto:</label>
<input type="file" class="form-control"
name="in_foto">
</div>

<div class="form-group">

</div>

<input type="hidden" name="nm_foto" value="<?php
echo $row->foto; ?>">
```

Dan hasil tampilannya akan seperti berikut:

← → ↻ ⓘ localhost/Codeigniter/index.php/Galeri... ☆ 🛡️ 🌐 📄 php 🔍 ⋮


---

**Judul:**

**Keterangan Foto:**

**Foto:**

No file chosen



- Konfigurasi Upload Update

Setelah file form sudah dibuat maka tambahkan kode untuk proses upload update dengan memodifikasi fungsi edit() pada controller Galeri.php.

```
function edit() {  
    $id = $this->input->post('id');  
  
    $nm_file = $this->input->post('nm_foto');  
    $config['upload_path'] = './assets/upload/';  
    $config['allowed_types'] = 'jpg|jpeg|png';  
    $config['file_name'] = $nm_file;
```

```
$config['overwrite'] = TRUE;
$this->upload->initialize($config);

if ($this->upload->do_upload('in_foto')) {

    $foto = $this->upload->data();
    $data = array(
        'judul_foto' => $this->input->post('in_judul'),
        'ket_foto' => $this->input->post('in_ket'),
        'foto' => $foto['file_name']
    );

} else {

    $data = array(

        'judul_foto' => $this->input->post('in_judul'),
        'ket_foto' => $this->input->post('in_ket'),
    );

}

$this->Galeri_m->edit_db($id, $data);
redirect('Galeri');
```

```
}
```

Running program dan coba edit lalu pastikan file dan data berhasil diubah tanpa terjadi error.