



DIGITAL  
TALENT  
SCHOLARSHIP

# VOCATIONAL SCHOOL GRADUATE ACADEMY

## Junior Web Developer

Hari #3

Lokasi Pelatihan: Hotel Roditha Banjarbaru



#JADIJAGOANDIGITAL

Badan Penelitian dan Pengembangan Sumber Daya Manusia

# PROFIL PENGAJAR



**Ronny Mantala, S.Kom., MM**  
**Dosen Politeknik Negeri Banjarmasin**



**Meycelino Avisha Taguh, ST**  
**Programmer CV. Gunung Media**

## Course Definition

Peserta mempelajari pengetahuan dan keterampilan untuk membuat pemrograman web sederhana yang mampu mendemonstrasikan keterampilan membuat aplikasi web sederhana meliputi:

- Menggunakan teknologi web sederhana (HTML, CSS, Bootstrap, PHP scripting)
- Membuat Struktur menu program dengan HTML, CSS
- Membuat pengelolaan data dengan aksi berupa lihat, input, edit, hapus dengan scripting PHP dan Database Mysql

Pada akhir pelatihan, peserta diharapkan mampu menghasilkan aplikasi sederhana berbasis web sebagai **bukti kompetensi hasil pembelajaran**.

## Learning Objective

Peserta mampu memenuhi pengetahuan , keterampilan dan sikap kerja pada unit kompetensi sebagai berikut:

No	Kode Unit	Judul Unit
1	J.620100.017.02	Mengimplementasikan pemrograman terstruktur
2	J.620100.015.01	Menyusun fungsi, file atau sumber daya pemrograman yang lain dalam organisasi yang rapi
3	J.620100.016.01	Menulis kode dengan prinsip sesuai guidelines dan best practices
4	J.620100.019.02	Menggunakan library atau komponen pre-existing

sesuai dengan persyaratan tempat kerja dalam **lima dimensi kompetensi** (*task skills, task management skills, contingency management skills, jobs/roles environment skills, dan transfer skills*)

# Materi Pembelajaran

## 1. Membuat program menggunakan *array*

- 1.1 Dimensi *array* telah ditentukan.
- 1.2 Tipe data *array* telah ditentukan.
- 1.3 Panjang *array* telah ditentukan.
- 1.4 Pengurutan *array* telah digunakan.

# Materi Pembelajaran

2. Membuat program untuk akses *file*

2.1 Program untuk menulis data dalam media penyimpan telah dibuat.

2.2 Program untuk membaca data dari media penyimpan telah dibuat.

3. Mengkompilasi Program

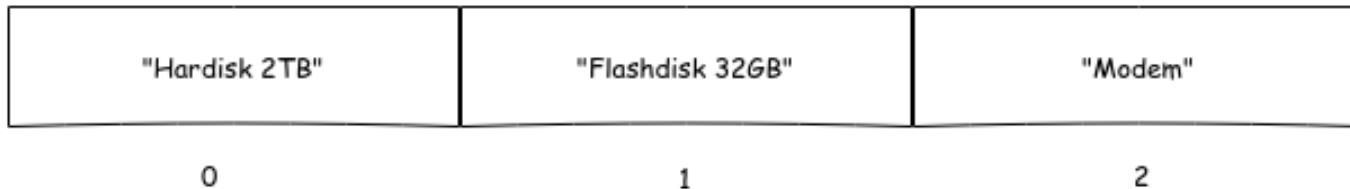
3.1 Kesalahan program telah dikoreksi.

3.2 Kesalahan *syntax* dalam program telah dibebaskan..

# Membuat program menggunakan *array*

## Apa Itu Array?

- Array adalah salah satu struktur data yang berisi sekumpulan data dan memiliki indeks. Indeks digunakan untuk mengakses nilai array.
- Indeks array selalu dimulai dari nol (0).



# Membuat Array di PHP

Array di PHP dapat kita buat dengan fungsi array() dan tanda kurung kotak [].

```
1 <?php
2
3 // membuat array kosong
4 $buah = array();
5 $hobi = [];
6
7 // membuat array sekaligus mengisinya
8 $minuman = array("Kopi", "Teh", "Jus Jeruk");
9 $makanan = ["Nasi Goreng", "Soto", "Bubur"];
10
11 // membuat array dengan mengisi indeks tertentu
12 $anggota[1] = "Komang";
13 $anggota[2] = "Medi";
14 $anggota[0] = "Usman";
15 ?>
```

## Membuat Array di PHP

Array dapat diisi dengan tipe data apa saja. Bahkan dicampur dengan data lain juga boleh.

```
1 <?php  
2  
3 $item = ["Bunga", 123, 39.12, true];  
4  
5 ?>
```

## Menampilkan isi Array

Untuk menampilkan isi array, kita bisa mengaksesnya melalui indeks.

```
1 <?php
2 // membuat array
3 $barang = ["Buku Tulis", "Penghapus", "Spidol"];
4
5 // menampilkan isi array
6 echo $barang[0]."<br>";
7 echo $barang[1]."<br>";
8 echo $barang[2]."<br>";
9 ?>
```

# Menampilkan isi Array

Contoh:

```
1 <?php
2 // membuat array
3 $barang = ["Buku Tulis", "Penghapus", "Spidol"];
4
5 // menampilkan isi array dengan perulangan for
6 for($i=0; $i < count($barang); $i++){
7     echo $barang[$i]."<br>";
8 }
9 ?>
```

```
1 <?php
2 // membuat array
3 $barang = ["Buku Tulis", "Penghapus", "Spidol"];
4
5 // menampilkan isi array dengan perulangan foreach
6 foreach($barang as $isi){
7     echo $isi."<br>";
8 }
9
10 echo "<hr>";
11
12 // menampilkan isi array dengan perulangan while
13 $i = 0;
14 while($i < count($barang)){
15     echo $barang[$i]."<br>";
16     $i++;
17 }
18 ?>
```

- Pergunakan loop untuk menampilkan sejumlah isi array
- Fungsi count() untuk menghitung banyaknya isi array.
  - Pada contoh di kiri, isi array sebanyak 3, maka perulangan akan dilakukan sebanyak 3x.

# Menghapus isi Array

- fungsi unset() untuk menghapus isi array. Fungsi ini juga dapat digunakan untuk menghapus variabel.

```
1 <?php
2 // membuat array
3 $hewan = [
4     "Burung",
5     "Kucing",
6     "Ikan"
7 ];
8 // menghapus kucing
9 unset($hewan[1]);
10
11 echo $hewan[0]."<br>";
12 echo $hewan[1]."<br>";
13 echo $hewan[2]."<br>";
14 echo "<hr>";
15 echo "<pre>";
16 print_r($hewan);
17 echo "</pre>";
18 ?>
```

fungsi print\_r() untuk menampilkan array secara mentah (raw).  
Biasanya untuk debugging.

# Menambahkan isi Array

Dua cara untuk menambah isi array:

1. Mengisi langsung ke nomer indeks yang ingin ditambahkan
2. Mengisi langsung ke indeks terakhir

```
1 <?php
2 // membuat array
3 $hobi = [
4     "Membaca",
5     "Menulis",
6     "Ngeblog"
7 ];
8 // menambahkan isi pada indeks ke-3
9 $hobi[3] = "Coding";
10
11 // menambahkan isi pada indeks terakhir
12 $hobi[] = "Olahraga";
13
14 // cetak array dengan perulangan
15 foreach($hobi as $hobiku){
16     echo $hobiku."<br>";
17 }
18 ?>
```

## Menambahkan isi Array

- Penambahan pada indeks yang sudah memiliki isi, maka isinya akan diganti/ditindih dengan yang baru.

```
1 <?php
2 // membuat array
3 $user = [
4     "Komang",
5     "Medi",
6     "Usman"
7 ];
8
9 // mengisi array pada indek ke-1 ("Evans")
10 $user[1] = "Evans";
11
12 // mencetak isi array
13 echo "<pre>";
14 print_r($user);
15 echo "</pre>";
16 ?>
```

## Array Asosiatif

- Array yang indeksnya tidak menggunakan nomer atau angka. tapi berbentuk kata kunci.

```
1 <?php
2 // membuat array asosiatif
3 $artikel = [
4     "judul" => "Belajar Pemrograman PHP",
5     "penulis" => "Digital Talent",
6     "view" => 128
7 ];
8
9 // mencetak isi array assosiatif
10 echo "<h2>".$artikel["judul"]."</h2>";
11 echo "<p>oleh: ".$artikel["penulis"]."</p>";
12 echo "<p>View: ".$artikel["view"]."</p>";
13 ?>
```

## Array Asosiatif

- Pada array asosiatif, kita menggunakan tanda => untuk mengasosiasikan sebuah kata kunci dengan isi array.
- Selain menggunakan tanda =>, kita juga bisa membuat array asosiatif seperti ini:

```
1 <?php
2 $email["subjek"] = "Apa Kabar?";
3 $email["pengirim"] = "komang@digitaltalent.com";
4 $email["isi"] = "Apa kabar? sudah lama tidak berjumpa";
5
6 echo "<pre>";
7 print_r($email);
8 echo "</pre>";
9 ?>
```



localhost/web-projects/array9.php

```
Array
(
    [subjek] => Apa Kabar?
    [pengirim] => komang@digitaltalent.com
    [isi] => Apa kabar? sudah lama tidak berjumpa
)
```

## Array Multi Dimensi

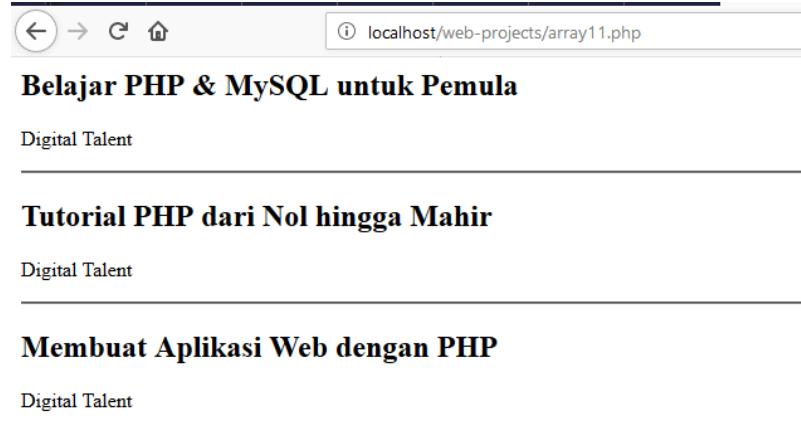
Array yang memiliki dimensi lebih dari satu, untuk membuat matrik, graph, dan struktur data rumit lainnya.

```
1 <?php
2 // ini adalah array dua dimensi
3 $matrik = [
4     [2,3,4],
5     [7,5,0],
6     [4,3,8],
7 ];
8
9 // cara mengakses isinya
10 echo $matrik[1][0]; //-> output: 7
11 ?>
```

# Array Multi Dimensi

Contoh lain:

```
1 <?php
2 // membuat array 2 dimensi yang berisi array asosiatif
3 $artikel = [
4 [
5     "judul" => "Belajar PHP & MySQL untuk Pemula",
6     "penulis" => "Digital Talent"
7 ],
8 [
9     "judul" => "Tutorial PHP dari Nol hingga Mahir",
10    "penulis" => "Digital Talent"
11 ],
12 [
13     "judul" => "Membuat Aplikasi Web dengan PHP",
14     "penulis" => "Digital Talent"
15 ]
16 ];
17 // menampilkan array
18 foreach($artikel as $post){
19     echo "<h2>".$post["judul"]."</h2>";
20     echo "<p>".$post["penulis"]."</p>";
21     echo "<hr>";
22 }
```



The screenshot shows a web browser window with the URL `localhost/web-projects/array11.php`. The page displays three articles, each consisting of a title and author information. The titles are bolded.

Article Title	Author
<b>Belajar PHP &amp; MySQL untuk Pemula</b>	Digital Talent
<b>Tutorial PHP dari Nol hingga Mahir</b>	Digital Talent
<b>Membuat Aplikasi Web dengan PHP</b>	Digital Talent

# Membuat program untuk akses *file*

# Operasi File pada PHP

PHP memiliki beberapa fungsi untuk:

- ✓ Membuka file
- ✓ Menulis data ke dalam file
- ✓ Membaca data dari file
- ✓ Menutup file

## Membuka File

Sebelum melakukan operasi file, terlebih dahulu kita buka file yang akan dioperasi.

Gunakan perintah : fopen();

```
1 <?php  
2 $file = fopen("welcome.txt", "r");  
3 ?>
```

Asumsika

## Menutup File

Setelah selesai menggunakan file maka file harus ditutup kembali.  
Gunakan perintah: fclose();

```
1 <?php
2 //buka file
3 $file = fopen("welcome.txt", "r");
4 //tutup file
5 fclose($file);
6 ?>
```

# Operasi Membaca File

- Pergunakan perintah : fgets(String filename, int length);  
Untuk int length, itu hanya opsi saja, menandakan berapa panjang karakter yang kita baca dalam file itu.

```
1 <?php
2 //buka file
3 $file = fopen("welcome.txt","r");
4 //baca file
5 echo fgets($file);
6 //tutup file
7 fclose($file);
8 ?>
```

## Membaca file dengan tag HTML

Berikut contohnya (usahakan file anda terdapat tag HTML-nya), sebagai contoh pada file welcome2.txt terdapat tag HTML.

```
1 <?php
2 //buka file
3 $file = fopen("welcome2.txt","r");
4 //baca file dengan fgetss
5 echo fgetss($file);
6 //tutup file
7 fclose($file);
8 ?>
```

# Membaca file Tingkat Lanjut

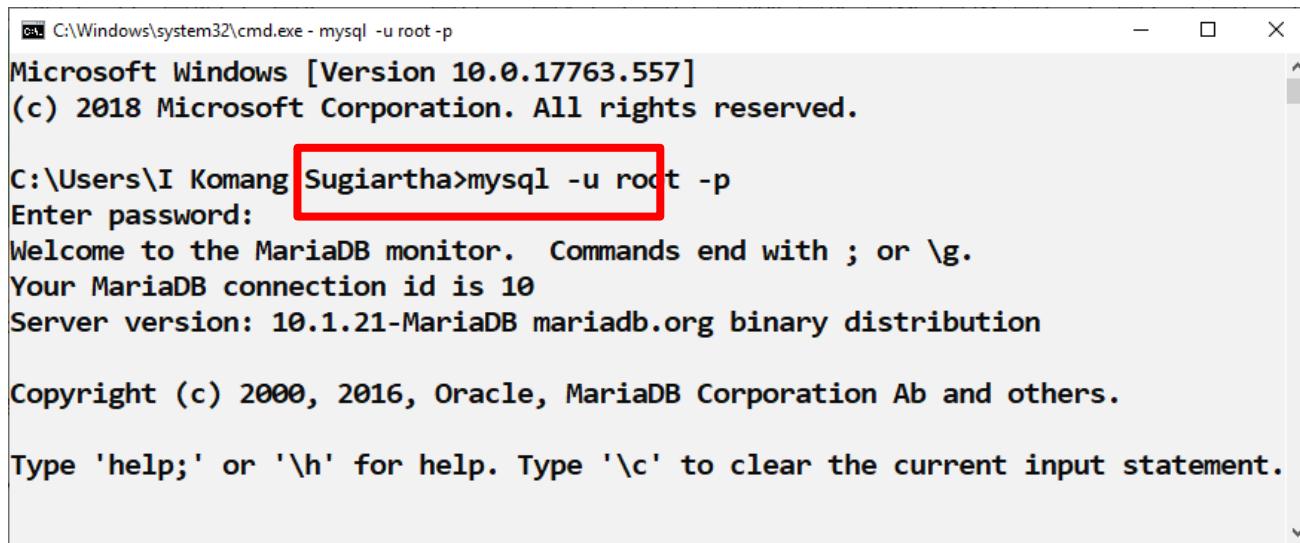
```
1 <?php
2 //membuka file
3 $data = "welcome.txt";
4 $bukafile = fopen($data, "r");
5 if (!$bukafile)
6 { print ("File $data gagal dibuka ! ");
7 exit;
8 }
9 while (!feof($bukafile))
10 { //membaca file
11 $data = fgets($bukafile,50);
12 //menampilkan data
13 print ("$data <br>");
14 }
15 //menutup file
16 fclose($bukafile);
17 ?>
```

## Akses Database pada PHP

- ❖ Membuat Koneksi dengan database MySQL
- ❖ Menjalankan query MySQL dari PHP
- ❖ Menampilkan Data
- ❖ Menambahkan Data
- ❖ Mengupdate Data
- ❖ Menghapus Data

# Membuat Database

Terlebih dahulu login ke database dengan perintah:



```
C:\Windows\system32\cmd.exe - mysql -u root -p
Microsoft Windows [Version 10.0.17763.557]
(c) 2018 Microsoft Corporation. All rights reserved.

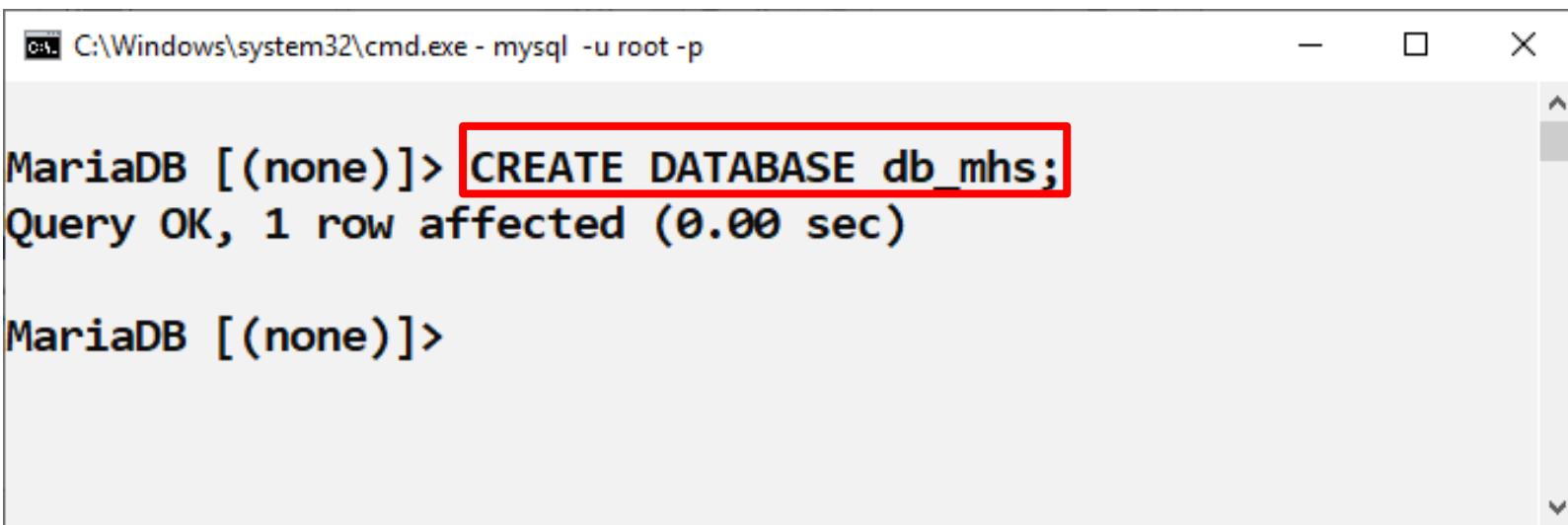
C:\Users\I Komang Sugiarta>mysql -u root -p
Enter password: [REDACTED]
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 10
Server version: 10.1.21-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

## Membuat Database

Buat database dengan nama db\_mhs;



A screenshot of a Windows Command Prompt window titled 'C:\Windows\system32\cmd.exe - mysql -u root -p'. The window contains the following text:

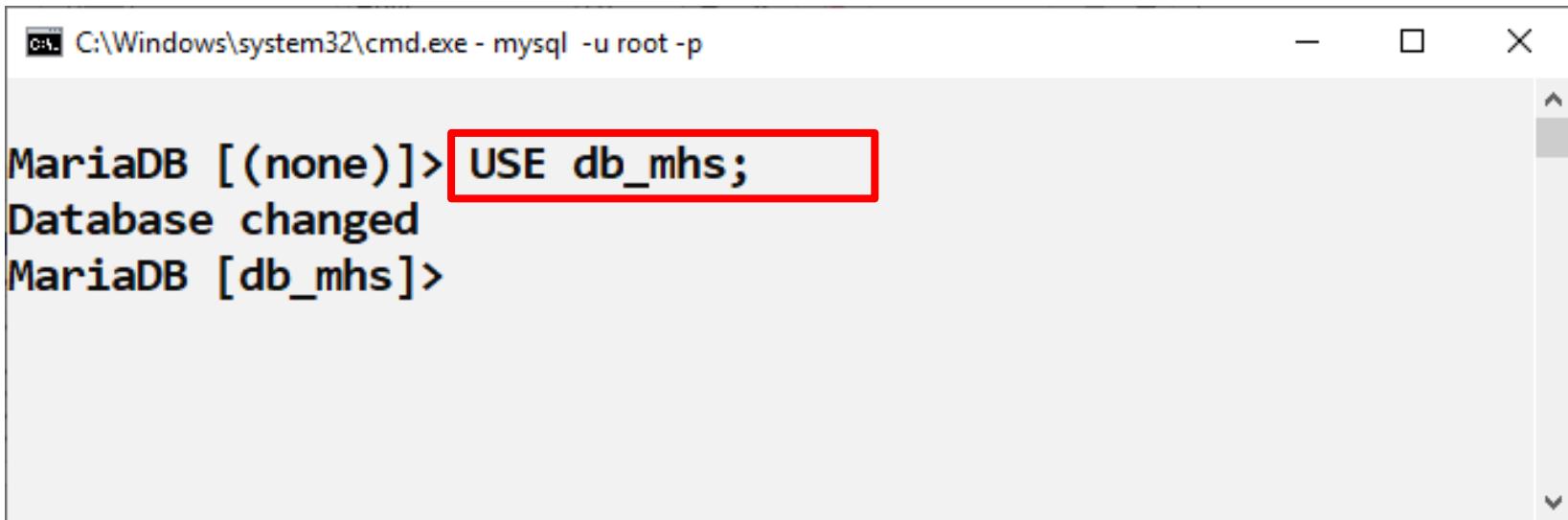
```
C:\Windows\system32\cmd.exe - mysql -u root -p
MariaDB [(none)]> CREATE DATABASE db_mhs;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]>
```

The command 'CREATE DATABASE db\_mhs;' is highlighted with a red rectangular box.

# Membuat Database

Gunakan database yang sudah dibuat.



A screenshot of a Windows Command Prompt window titled 'C:\Windows\system32\cmd.exe - mysql -u root -p'. The window contains the following text:

```
MariaDB [(none)]> USE db_mhs;
Database changed
MariaDB [db_mhs]>
```

The command 'USE db\_mhs;' is highlighted with a red rectangular box.

# Membuat Database

Buat Table dengan perintah

```
C:\Windows\system32\cmd.exe - mysql -u root -p
MariaDB [db_mhs]> CREATE TABLE mahasiswa (
    ->     nim varchar(13) NOT NULL,
    ->     id_mhs int(11) NOT NULL PRIMARY KEY AUTO_INCREMENT,
    ->     nama varchar(40) NOT NULL,
    ->     jenis_kelamin enum('P','L') NOT NULL,
    ->     jurusan varchar(30) NOT NULL,
    ->     alamat text NOT NULL);
Query OK, 0 rows affected (0.03 sec)

MariaDB [db_mhs]>
```

# Membuat Database

## Struktur Table mahasiswa

```
C:\Windows\system32\cmd.exe - mysql -u root -p
MariaDB [db_mhs]> DESC mahasiswa;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key  | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| nim   | varchar(13) | NO   |       | NULL    |                |
| id_mhs | int(11)      | NO   | PRI   | NULL    | auto_increment |
| nama  | varchar(40)  | NO   |       | NULL    |                |
| jenis_kelamin | enum('P','L') | NO   |       | NULL    |                |
| jurusan | varchar(30)  | NO   |       | NULL    |                |
| alamat | text        | NO   |       | NULL    |                |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)
```

## Insert Data

- Untuk menampilkan data tentunya kita butuh data yang akan ditampilkan, oleh karena itu kita membutuhkan beberapa sample data untuk di tampilkan nantinya.
- Silahkan jalankan SQL Query berikut ini untuk memasukan beberapa record kedalam tabel mahasiswa.



```
C:\Windows\system32\cmd.exe - mysql -u root -p
MariaDB [db_mhs]> INSERT INTO mahasiswa VALUES ('TI102132', 1, 'Komang', 'L',
'Sistem Informasi', 'Depok');
Query OK, 1 row affected (0.00 sec)
```

## Membuat Koneksi PHP MySQL

- File ini berguna untuk menyimpan settingan konfigurasi dan perintah untuk koneksi ke database menggunakan bahasa pemrograman PHP.
- Silahkan buat sebuah file baru dengan nama koneksi.php dan ketikkan script berikut ini :

```
1 <?php
2 // konfigurasi database
3 $host      = "localhost";
4 $user      = "root";
5 $password   = "";
6 $database   = "db_mhs";
7 // perintah php untuk akses ke database
8 $koneksi = mysqli_connect($host, $user, $password, $database);
9 ?>
```

# Menampilkan Data dari MySQL Dengan PHP

Silahkan buat file index.php dan ketik program seperti dibawah ini :

```
1 <h2>List Mahasiswa</h2>
2 <table border="1">
3   <tr><th>NO</th><th>NIM</th><th>NAMA</th><th>GENDER</th><th>JURUSAN</th>
4   </tr>
5   <?php
6     include 'koneksi.php';
7     $mahasiswa = mysqli_query($koneksi, "SELECT * from mahasiswa");
8     $no=1;
9     foreach ($mahasiswa as $row){
10       $jenis_kelamin = $row['jenis_kelamin']=='P'?'Perempuan':'Laki laki';
11       echo "<tr>
12         <td>$no</td>
13         <td>".$row['nim']."</td>
14         <td>".$row['nama']."</td>
15         <td>".$jenis_kelamin."</td>
16         <td>".$row['jurusan']."</td>
17       </tr>";
18     }
19   ?>
20 </table>
```

# Membuat Form Inputan Data Dengan PHP

Pertama buat sebuah form (mis. file form-input.php), untuk digunakan sebagai interface untuk menginputkan data yang akan disimpan kedalam database.

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <title>Membuat Form Inputan Data</title>
5     </head>
6     <body>
7         <form method="post" action="simpan.php">
8             <table>
9                 <tr><td>NIM</td><td><input type="text" onkeyup="isi_otomatis()" name="nim"></td></tr>
10                <tr><td>NAMA</td><td><input type="text" name="nama"></td></tr>
11                <tr><td>JENIS KELAMIN</td><td>
12                    <input type="radio" name="jenis_kelamin" value="L">Laki Laki
13                    <input type="radio" name="jenis_kelamin" value="P">Perempuan
14                </td></tr>
15                <tr><td>JURUSAN</td><td>
16                    <select name="jurusan">
17                        <option value="TEKNIK INFORMATIKA">TEKNIK INFORMATIKA</option>
18                        <option value="TEKNIK MESIN">TEKNIK MESIN</option>
19                        <option value="TEKNIK KIMIA">TEKNIK KIMIA</option>
20                    </select>
21                </td></tr>
22                <tr><td>ALAMAT</td><td><input type="text" name="alamat"></td></tr>
23                <tr><td colspan="2"><button type="submit" value="simpan">SIMPAN</button></td></tr>
24            </table>
25        </form>
26    </body>
27 </html>
```

# Membuat Form Inputan Data Dengan PHP

Form ini menggunakan elemen textbox, radio button dan dropdown.

Jika sudah selesai menulis scriptnya, silahkan save dan akses form-input.php melalui web browser anda, maka akan tampil seperti disamping :

The screenshot shows a web browser window with the URL 'localhost/web-projects/form-input.php' in the address bar. The page displays a form with the following fields:

- NIM: An empty text input field.
- NAMA: An empty text input field.
- JENIS KELAMIN: A radio button group with two options: 'Laki Laki' and 'Perempuan'. 'Laki Laki' is selected.
- JURUSAN: A dropdown menu set to 'TEKNIK INFORMATIKA'.
- ALAMAT: An empty text input field.
- SIMPAN: A greyed-out button labeled 'SIMPAN'.

# Membuat Form Inputan Data Dengan PHP

Silahkan buat sebuah script baru dengan nama simpan.php dan ketik script berikut ini :

```
1 <?php
2 include 'koneksi.php';
3 // menyimpan data kedalam variabel
4 $nim          = $_POST['nim'];
5 $nama         = $_POST['nama'];
6 $jurusan      = $_POST['jurusan'];
7 $jenis_kelamin = $_POST['jenis_kelamin'];
8 $alamat       = $_POST['alamat'];
9 // query SQL untuk insert data
10 $query="INSERT INTO mahasiswa SET nim='$nim',nama='$nama',jurusan='$jurusan'
           ,jenis_kelamin='$jenis_kelamin',alamat='$alamat'";
11 mysqli_query($koneksi, $query);
12 // mengalihkan ke halaman index.php
13 header("location:index.php");
14 ?>
```

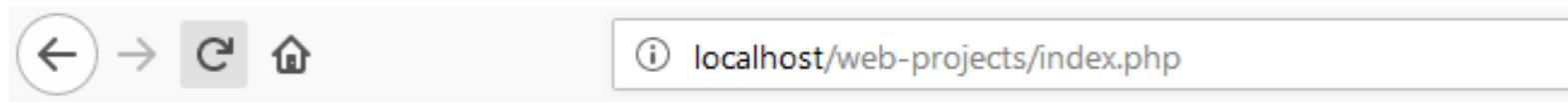
# Update dan Delete Data pada Database

modifikasi  
index.php

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Digital Talent</title>
5      </head>
6      <body>
7          <h2>List Mahasiswa</h2>
8          <table border="1">
9              <tr><th>NO</th><th>NIM</th><th>NAMA</th><th>GENDER</th><th>JURUSAN</th><th>ACTION</th></tr>
10             <?php
11                 include 'koneksi.php';
12                 $mahasiswa = mysqli_query($koneksi, "SELECT * from mahasiswa");
13                 $no = 1;
14                 foreach ($mahasiswa as $row) {
15                     $jenis_kelamin = $row['jenis_kelamin'] == 'P' ? 'Perempuan' : 'Laki laki';
16                     echo "<tr>
17                         <td>$no</td>
18                         <td>" . $row['nim'] . "</td>
19                         <td>" . $row['nama'] . "</td>
20                         <td>" . $jenis_kelamin . "</td>
21                         <td>" . $row['jurusan'] . "</td>
22                         <td><a href='form-edit.php?id_mhs=$row[id_mhs]'>Edit</a>
23                             <a href='delete.php?id_mhs=$row[id_mhs]'>Delete</a>
24                         </td>
25                         </tr>";
26                     $no++;
27                 }
28             ?>
29         </table>
30
31     </body>
32 </html>
```

## Update dan Delete Data pada Database

Sehingga output dari program diatas menjadi seperti ini:



### List Mahasiswa

NO	NIM	NAMA	GENDER	JURUSAN	ACTION
1	12345	Komang	Laki laki	TEKNIK INFORMATIKA	<a href="#">Edit</a> <a href="#">Delete</a>

# Membuat Form Update Data pada Database

Sekarang buatlah sebuah file baru dengan nama form-edit.php dan ketiklah script ini:

```
1 <?php
2 include 'koneksi.php';
3 $id      = $_GET['id'];
4 $mahasiswa = mysqli_query($koneksi, "select * from mahasiswa where id_mhs='$id'");
5 $row      = mysqli_fetch_array($mahasiswa);
6 // membuat data jurusan menjadi dinamis dalam bentuk array
7 $jurusan  = array('TEKNIK INFORMATIKA','TEKNIK ELEKTRO','REKAMEDIS');
8 // membuat function untuk set aktif radio button
9 function active_radio_button($value,$input){
10    // apabila value dari radio sama dengan yang di input
11    $result = $value==$input?'checked':'';
12    return $result;
13 }
14 ?>
```

# Membuat Form Update Data pada Database

## Lanjutan program form-edit.php

```
15  <!DOCTYPE html>
16  <html>
17      <head>
18          <title>Digital Talent</title>
19      </head>
20      <body>
21          <form method="post" action="update.php">
22              <input type="hidden" value=<?php echo $row['id_mhs'];?>" name="id_mhs">
23              <table>
24                  <tr><td>NIM</td><td><input type="text" value=<?php echo $row['nim'];?>" name="nim"></td></tr>
25                  <tr><td>NAMA</td><td><input type="text" value=<?php echo $row['nama'];?>" name="nama"></td></tr>
26                  <tr><td>JENIS KELAMIN</td><td>
27                      <input type="radio" name="jenis_kelamin" value="L" <?php echo active_radio_button("L", $row['jenis_kelamin']);?>>>Laki Laki
28                      <input type="radio" name="jenis_kelamin" value="P" <?php echo active_radio_button("P", $row['jenis_kelamin']);?>>>Perempuan
29                  </td></tr>
30                  <tr><td>JURUSAN <?php echo $row['jurusan'];?></td><td>
31                      <select name="jurusan">
32                          <?php
33                              foreach ($jurusan as $j){
34                                  echo "<option value='$j' ";
35                                  echo $row['jurusan']==$j?'selected="selected"':'';
36                                  echo ">$j</option>";
37                              }
38                          ?>
39                      </select>
40                  </td></tr>
41                  <tr><td>ALAMAT</td><td><input value=<?php echo $row['alamat'];?>" type="text" name="alamat"></td></tr>
42                  <tr><td colspan="2"><button type="submit" value="simpan">SIMPAN PERUBAHAN</button>
43                      <a href="index.php">Kembali</a></td></tr>
44                  </table>
45          </form>
46      </body>
47  </html>
```

# Membuat Proses Update

Sekarang kita akan membuat file edit.php dan ketiklah script :

```
1 <?php
2 include 'koneksi.php';
3 // menyimpan data kedalam variabel
4 $id_mhs    = $_POST['id_mhs'];
5 $nim        = $_POST['nim'];
6 $nama       = $_POST['nama'];
7 $jurusan    = $_POST['jurusan'];
8 $jenis_kelamin = $_POST['jenis_kelamin'];
9 $alamat     = $_POST['alamat'];
10 // query SQL untuk insert data
11 $query="UPDATE mahasiswa SET nim='$nim',nama='$nama',jurusan='$jurusan',jenis_kelamin='
           $jenis_kelamin',alamat='$alamat' where id_mhs='$id_mhs'";
12 mysqli_query($koneksi, $query);
13 // mengalihkan ke halaman index.php
14 header("location:index.php");
15 ?>
```

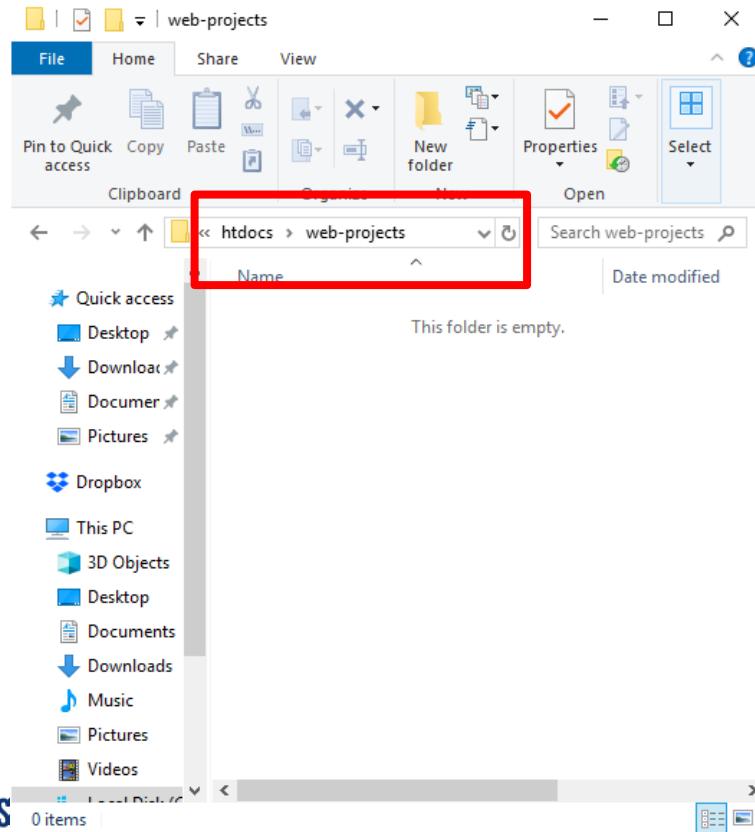
# **Menyusun fungsi, file atau sumber daya pemrograman yang lain dalam organisasi yang rapi**

J.620100.015.01

# Menyusun Struktur Folder dan File PHP

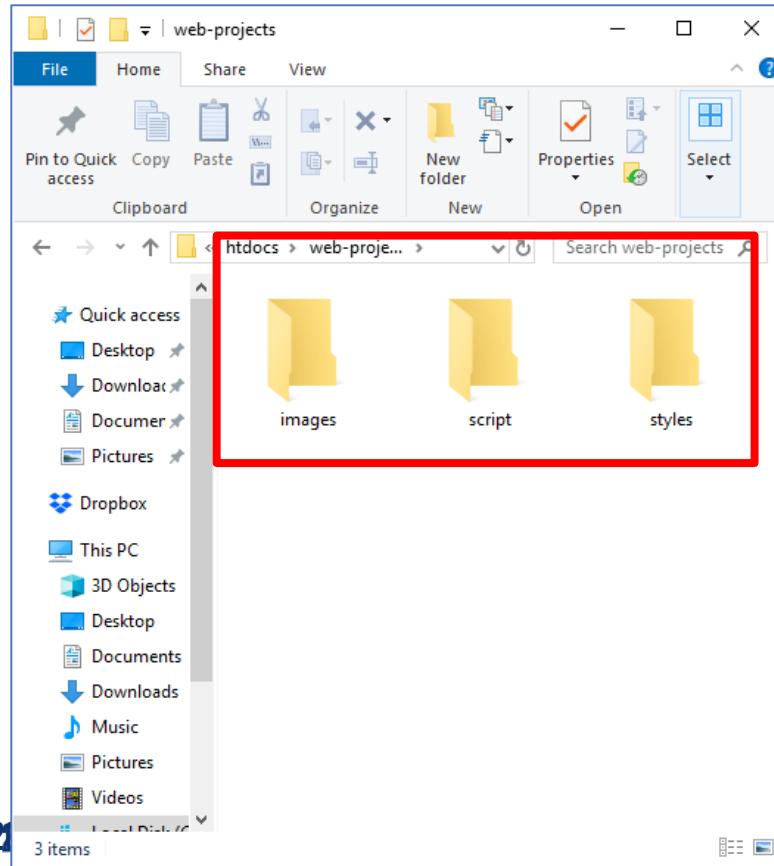
- ❖ Suatu website terdiri atas banyak file, seperti konten teks, kode, stylesheet, konten media, dan lain-lain.
- ❖ Ketika kamu sedang membuat sebuah website, kamu perlu merangkai file-file ini menjadi sebuah struktur yang sesuai pada komputermu.
- ❖ Memastikan file-file tersebut saling terhubung antara satu dengan lainnya, dan memastikan semua konten sudah benar sebelum akhirnya kamu unggah mereka ke suatu server.
- ❖ Ketika kamu sedang mengerjakan satu website lokal pada komputer pribadimu, kamu harus menyimpan semua file terkait pada satu folder sesuai dengan struktur website yang diunggah ke server.
- ❖ Folder ini harus disimpan di dalam folder htdocs agar pada saat server local diaktifkan, kita dapat melihat hasil website kita sebelum di upload ke server.

# Menyusun Struktur Folder dan File PHP



- ❖ Buatlah sebuah folder dengan nama web-projects (atau nama lain yang serupa) didalam folder htdocs.
- ❖ Di sinilah proyek-proyek websitemu akan disimpan.

# Menyusun Struktur Folder dan File PHP



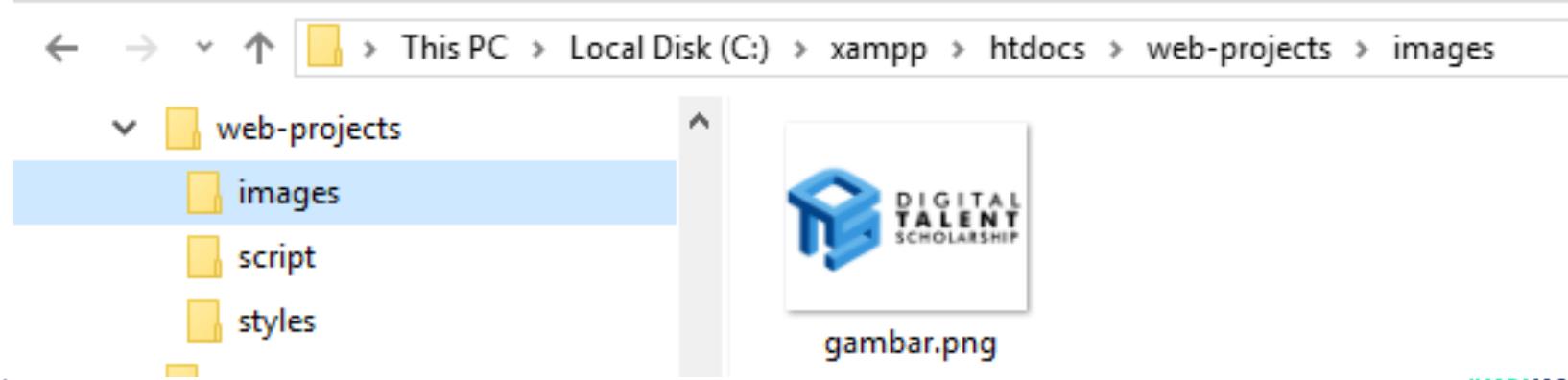
- ❖ Hal yang paling sering kamu temukan pada proyek website apapun yang kita buat adalah sebuah file indeks HTML/PHP dan folder untuk menyimpan gambar, file CSS, dan file script.
- ❖ Maka buat 3 folder didalam web-projects dengan nama sebagai berikut.

# Menyusun Struktur Folder dan File PHP

- ❖ Folder images: Folder ini berisi semua gambar yang akan digunakan pada websitemu.
- ❖ Folder styles: Folder ini berisi kode CSS yang digunakan untuk kontenmu (contoh: pengaturan teks dan warna latar belakang).
- ❖ Folder scripts: Folder ini berisi semua kode JavaScript yang digunakan untuk menambah fungsionalitas interaktif pada websitemu.

## Jalur/Path file

- ❖ Agar file-file dapat berkomunikasi antara satu dengan lainnya, kamu harus menyediakan sebuah path file — sebuah rute sehingga satu file dapat mengetahui di mana lokasi file yang lainnya.
- ❖ Untuk mendemonstrasikannya, kita akan membuat sebuah file dengan nama index.html, dan membuatnya menampilkan gambar.
- ❖ Salin/Copy gambar yang kamu pilih ke dalam folder image.



# Jalur/Path file

- ❖ Buat file index.html, dan tambahkan kode berikut ke dalam file tersebut.

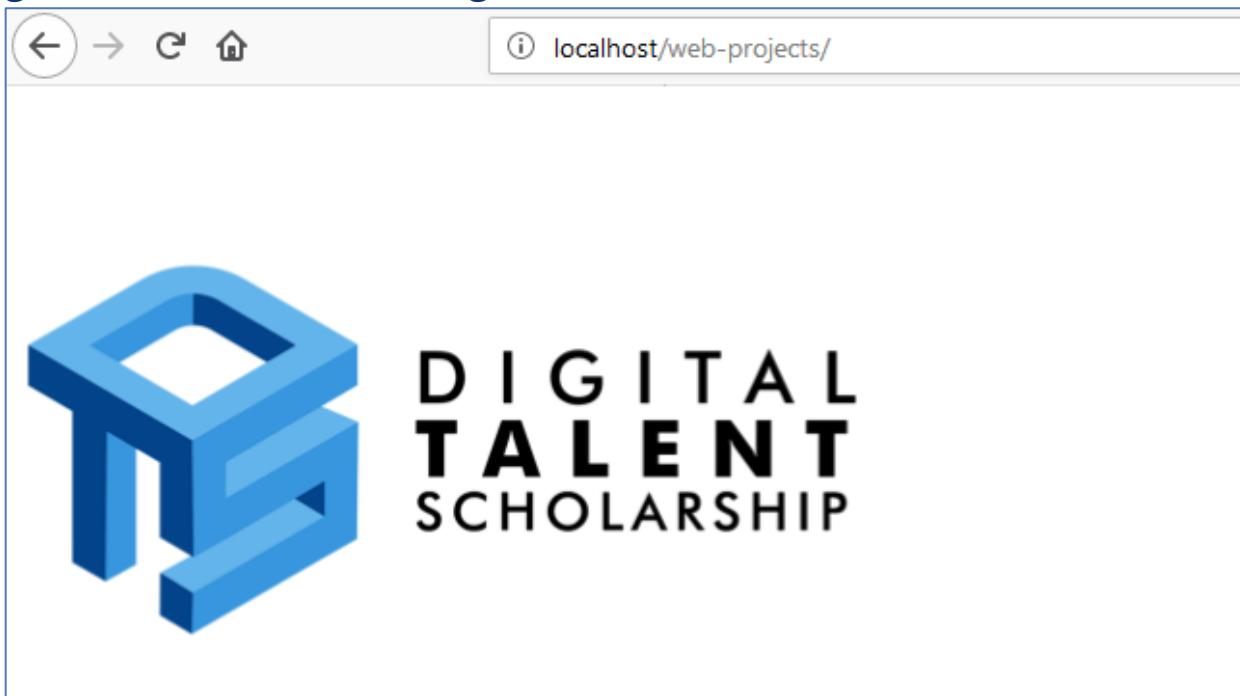
```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>Menampilkan Gambar</title>
5   </head>
6   <body>
7     
8   </body>
9 </html>
```

## Jalur/Path file

- ❖ Baris  adalah kode HTML yang menyisipkan gambar ke dalam halaman.
- ❖ Kita perlu memberitahu HTML di mana letak gambar tersebut.
- ❖ Gambar berada pada folder images yang sejajar dengan file index.html.
- ❖ Untuk menelusuri struktur file dari index.html ke gambar kita, nama file kita beri nama gambar.png, sehingga jalur file-nya menjadi images/gambar.png.
- ❖ Simpan file HTML, kemudian buka halaman tersebut pada web browser.

## Jalur/Path file

❖ Hasil dari program diatas adalah sebagai berikut:

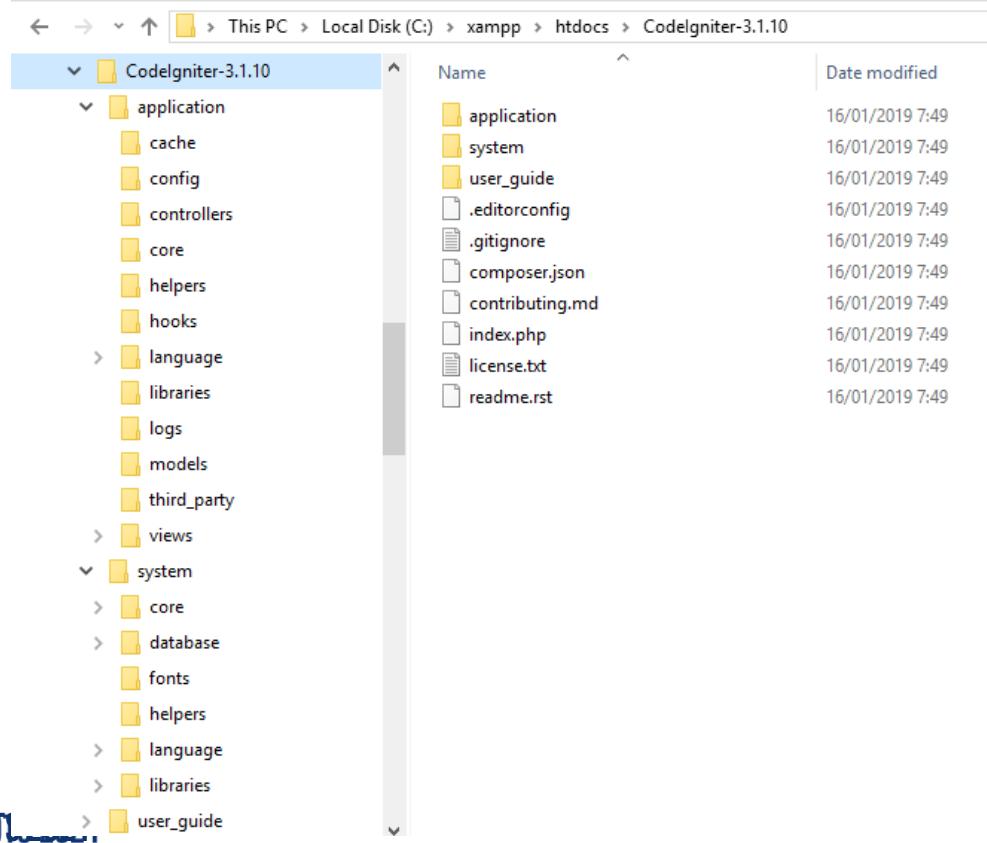


# Mengorganisasikan Sumberdaya Pemrograman

Yang Perlu diperhatikan dalam Mengorganisasikan Sumberdaya Pemrograman:

- ❖ Folder dan nama file diberi nama sesuai dengan fungsinya.
- ❖ Tempatkan File source code yang mempunyai fungsionalitas yang sama dalam satu folder
- ❖ Membuat file Readme.txt

# Contoh Pengorganisasian File PHP pada Framework Codeigniter



Folder config adalah folder yang digunakan untuk meletakkan file konfigurasi.

Folder controller adalah folder yang berfungsi untuk menyimpan file-file controller yang digunakan untuk mengatur alur proses aliran data atau fungsi yang dibuat.

# Membuat File Readme.txt

- ❖ README adalah sebuah berkas atau file yang berisi informasi tentang berkas lain di dalam direktori atau arsip dari perangkat lunak komputer.
- ❖ File Readme berisikan informasi tentang :
  - ✓ Petunjuk Konfigurasi
  - ✓ Petunjuk Instalasi
  - ✓ Penjelasan Struktur/Hirarki Program
  - ✓ Informasi hak cipta dan perizinan menggunakan perangkat lunak
  - ✓ ChangeLog SourceCode

## Coding-Standard HTML

- ❖ Semua kode HTML yang akan anda tulis harus valid dan terstruktur.
- ❖ Sebelum mulai menulis kode program, anda harus memeriksa spesifikasi proyek yang anda kerjakan dengan spesifikasi HTML yang anda gunakan.
- ❖ Spesifikasi yang sekarang biasa digunakan adalah HTML5 Document Type Definition.

```
<!DOCTYPE HTML>
```

## Gunakan Lower Case untuk Nama

Elemen dan atribut harus ditulis menggunakan huruf lower case

```
<!-- Correct -->
<input name="name" type="text" />

<!-- Wrong -->
<input name="name" TYPE="text" />
```

## Tag Penutup

Baik elemen yang kosong maupun yang tidak kosong, harus memiliki tag penutup yang sesuai.

- ❖ Elemen yang tidak kosong

```
<h1>My title</h1>
<p>Some text</p>
```

- ❖ Elemen yang kosong

```
<span></span>
```

## Tag Penutup (2)

Khusus untuk elemen dengan single tag harus menggunakan > di akhir statement.

```
<br>
<hr>

```

## Elemen Bersarang

Elemen bersarang (nested element) harus ditulis dengan benar, sesuai dengan susunan tag-nya.

Penulisan yang benar

```
<!-- Correct -->
<div>
  <p>Some text</p>
</div>
```

Penulisan yang salah

```
<!-- Wrong -->
<div>
  <p>Some text</div>
</p>
```

## Nilai dari Atribut

Menuliskan nilai (value) dari sebuah atribut harus menggunakan tanda petik dua [“ ”], walaupun nilai tersebut berupa nilai numerik.

```
<!-- Correct -->
<input name="age" type="text" size="3" />

<!-- Wrong -->
<input name=age type=text size=3 />
```

## Indentasi

- ❖ Gunakan 2 spasi untuk indentasi kode.
- ❖ Gunakan indentasi secara konsisten untuk memudahkan dalam membaca kode.
- ❖ Ketika ada sebuah elemen yang didalamnya terdapat lebih dari satu baris kode, berikan indentasi pada konten elemen, antara tag awal dan tag akhir, untuk memudahkan dalam melihat di mana elemen dimulai dan berakhir.

```
<div class="container">
  <header class="header">
    <h1>Site Name<span></span></h1>
  </header>
  <!-- / header -->
  <hr>
  <nav class="navigation">
    <ul>
      <li><a href="#">Link</a></li>
      <li><a href="#">Link</a></li>
      <li><a href="#">Link</a></li>
      <li><a href="#">Link</a></li>
      <li><a href="#">Link</a></li>
    </ul>
  </nav>
  <!-- / navigation -->
</div>
<!-- / container -->
```

## Best-Practice untuk PHP

- ❖ Inisialisasi variable
- ❖ Inisialisasi/ Urutan deklarasi variable, method dan properties

## Inisialisasi variable

- ❖ Inisialisasi variable harus dilakukan pertama kali sebelum penggunaan variable tersebut

- ❖ Boleh saja:

```
<?php

if ($expr) {
    // ....
}

$movies = array();
$movies = get_movies();

// EOF
```

- ❖ Namun sebaiknya gunakan ini:

```
<?php

$movies = array();

if ($expr) {
    // ....
}

$movies = get_movies();

// EOF
```

## Inisialisasi/ Urutan deklarasi variable, method dan properties

- ❖ HARUS diawali dengan global, ikuti dengan konstanta, dan diakhiri dengan variabel local
- ❖ HARUS diawali dengan properti kemudian diikuti dengan method dalam class
- ❖ HARUS diawali dengan public, diikuti protected, dan diakhiri dengan private method dalam class
- ❖ HARUS sesuai dengan abjad dalam kelompoknya

## Inisialisasi/ Urutan deklarasi variable, method dan properties

```
<?php

global $app_config,
       $cache,
       $db_connection;

define('ENVIRONMENT', 'PRODUCTION');
define('MODE', 1);

$id = 0;
$firstname = '';
$lastname = '';

// EOF
```

```
<?php

namespace MyCompany\Model;

class Office
{
    private $id;
    private $name;
    private $status;

    public function get_id() {
        // ...
    }

    protected function get_status() {
        // ...
    }

    private function get_name() {
        // ...
    }
}

// EOF
```

## Coding Standard untuk CSS

- Format CSS
- Pemberian Nama
- Komentar

## Format CSS

- Semua dokumen CSS harus menggunakan dua spasi untuk indentasi dan file tidak boleh memiliki dua spasi tambahan (whitespace).
- Gunakan tanda kutip ganda.
- Gunakan notasi steno hanya jika diperlukan.
- Beri spasi setelah: dalam deklarasi properti.
- Letakkan spasi sebelum {dalam deklarasi aturan.
- Gunakan kode warna hex # 000 kecuali menggunakan rgba () .

## Format CSS (2)

- Selalu berikan properti fallback untuk browser lama.
- Gunakan satu baris per deklarasi properti.
- Selalu ikuti aturan dengan satu baris spasi.
- Selalu mengutip konten url () dan @import ().
- Jangan membuat indentasi blok.

## Format CSS (Contoh)

```
.media {  
    overflow: hidden;  
    color: #fff;  
    background-color: #000; /* Fallback value */  
    background-image: linear-gradient(black, grey);  
}  
  
.media .img {  
    float: left;  
    border: 1px solid #ccc;  
}  
  
.media .img img {  
    display: block;  
}  
  
.media .content {  
    background: #fff url("../images/media-background.png") no-repeat;  
}
```

## Pemberian Nama

- ❖ Semua nama yang digunakan untuk id, kelas dan atribut harus menggunakan huruf kecil, dan menggunakan tanda hubung apa bila diperlukan pemisahan.

```
/* GOOD */
.dataset-list {}

/* BAD */
.datasetlist {}
.datasetList {}
.dataset_list {}
```

## Komentar

- ❖ Gunakan komentar untuk menjelaskan apa pun yang memiliki makna ganda atau tidak mudah dipahami orang lain yang membaca kode CSS anda

```
.prose p {  
    font-size: 1.1666em /* 14px / 12px */;  
}  
  
.ie7 .search-form {  
    /*  
        Force the item to have layout in IE7 by setting display to block.  
        See: http://reference.sitepoint.com/css/haslayout  
    */  
    display: inline-block;  
}
```

## Coding Standard untuk PHP

- ❖ Tag PHP
- ❖ Namespace
- ❖ Komentar
- ❖ Keyword Include/Require Once
- ❖ Formatting
- ❖ Fungsi
- ❖ Struktur Kontrol
- ❖ Class

## Tag PHP

- ❖ Tag Pembuka HARUS berada pada barisnya sendiri dan HARUS diikuti oleh baris kosong.

```
< ? php print_welcome_message ();
```

Salah karena <?php tidak pada jalurnya sendiri.

```
<? php  
print_welcome_message ();
```

Salah karena <?php tidak diikuti oleh baris kosong

```
<? php  
  
print_welcome_message ();
```

BENAR

## Tag Penutup

### ❖ Benar

Tidak menggunakan ?>

```
<? php  
  
print_welcome_message ();
```

### ❖ Salah

Salah karena ?> digunakan.

```
<? php  
  
print_welcome_message ();  
? >
```

- ❖ Tag Penutup TIDAK HARUS digunakan dalam file PHP, namun digunakan apabila ada kode PHP yang dituliskan dalam HTML (inline).

## Tag Pembuka dan Penutup (inline PHP)

❖ Benar

Karena tag <?php dan ?> berada dalam satu baris

```
< div >
    < h1 > <? php print_welcome_message (); ?> </ h1 >
</ div >
```

❖ Salah

Salah karena tag <?php dan ?> tidak dalam satu baris

```
< div >
    < h1 > <? php
print_welcome_message ();
?> </ h1 >
</ div >
```

## Tag Pembuka Pendek (short open tag)

Tag terbuka pendek sebaiknya TIDAK digunakan.

❖ Benar

```
<? php  
print_welcome_message ();
```

❖ Salah

```
< ?  
print_welcome_message ();
```

## Tag Echo Pendek (short echo tag)

Tag gema pendek sebaiknya digunakan dalam file PHP yang berada di dalam HTML (inline).

Masih dapat diterima,

```
< div >
    < p > <?php echo get_welcome_message(); ?> </ p >
</ div >
```

Namun sebaiknya:

```
< div >
    < p > <?= get_welcome_message(); ?> </ p >
</ div >
```

## Akhir dari File

- Bagian ini menjelaskan bagaimana setiap file PHP harus diakhiri.
- Cara penulisan komentar akhir file:
  - HARUS dimasukkan di akhir file
  - HARUS berada di jalurnya sendiri
  - HARUS didahului dan diakhiri oleh garis-garis kosong

```
<? php  
  
print_welcome_message ();  
  
// EOF
```

## Penanganan Galat / Error

- ❖ Penanganan error adalah proses menangkap kesalahan pada program untuk diambil tindakan yang sesuai.
- ❖ Metode Penanganan Error
  - Menggunakan Function "die()"
  - Menulis function Penanganan Error sendiri
  - Error reporting

## Function "die()"

Misalkan terdapat kode sbb:

```
<?php  
$file=fopen("hello.txt", "r");  
?>
```

Jika file tersebut tidak ditemukan, maka akan keluar pesan berikut:

```
Warning: fopen(hello.txt) [function.fopen]: failed to open stream:  
No such file or directory in /var/www/webfolder/test.php on line 2
```

## Penggunaan Function die();

Memeriksa keberadaan file terlebih dahulu

```
<?php
if(!file_exists("hello.txt")) {
    die("File tidak ditemukan");
} else {
    $file=fopen("hello.txt", "r");
}
?>
```

Jika file tidak ditemukan, maka akan muncul pesan:

```
File tidak ditemukan
```

Program akan terhenti ketika memanggil function die(); dan menampilkan pesan yang dapat dipahami oleh pengguna

## Metode Menulis Function Penanganan Error Sendiri

- ❖ Kita dapat menulis fungsi sendiri untuk menangani kesalahan apa pun

```
<?php
//function penanganan error
function customError($errno, $errstr) {
    echo "<b>Error:</b> [$errno] $errstr";
}

//set error handler
set_error_handler("customError");

//picu error
echo($test);
?>
```

Error: [8] Undefined  
variable: test

## Metode Menulis Function Penanganan Error Sendiri

- ❖ Fungsi yang dibuat harus mampu menangani minimal dua parameter (tingkat kesalahan dan pesan kesalahan) tetapi dapat menerima hingga lima parameter (opsional: file, nomor baris, dan konteks kesalahan):
- ❖ Syntax:

```
error_function(error_level,error_message,error_file,error_line, error_context)
```

## Metode Menulis Function Penanganan Error Sendiri

❖ Parameter Kesalahan:

Parameter	Deskripsi
error_level	Wajib. Menentukan tingkat laporan kesalahan untuk kesalahan yang ditentukan pengguna. Harus berupa angka nilai. Lihat tabel di slide berikutnya untuk daftar tingkat laporan kesalahan.
error_message	Wajib. Pesan error yang disampaikan ke pengguna.
error_file	Tidak wajib. Menampilkan nama file dimana error terjadi
error_line	Tidak wajib. Menampilkan nomor baris dimana error terjadi.
error_context	Tidak Wajib. Menampilkan sebuah array yang mengandung setiap variabel, dan nilainya yang digunakan ketika terjadi error.

## Metode Menulis Function Penanganan Error Sendiri

### ❖ Level Kesalahan:

Nilai	Konstanta	Deskripsi
2	E_WARNING	Kesalahan yang tidak fatal. Eksekusi program tidak terhenti.
8	E_NOTICE	Pemberitahuan Run-time. Dalam kode mungkin terdapat sebuah kesalahan, tetapi dapat juga terjadi ketika kode berjalan normal
256	E_USER_ERROR	Kesalahan fatal yang dibuat oleh programmer. Ini seperti sebuah E_ERROR yang di set oleh programmer menggunakan function trigger_error()
512	E_USER_WARNING	Peringatan yang tidak fatal yang dibuat oleh programmer. Ini seperti sebuah E_WARNING yang di set oleh programmer menggunakan function trigger_error()
1024	E_USER_NOTICE	Pemberitahuan yang dibuat oleh programmer. Ini seperti sebuah E_NOTICE yang di set oleh programmer menggunakan function trigger_error()
4096	E_RECOVERABLE_ERROR	Kesalahan fatal yg dapat ditangkap. Ini seperti sebuah E_ERROR Tapi yang ditangkap oleh programmer dg mendefinisikan handle (set_error_handler())
8191	E_ALL	Semua kesalahan dan peringatan

## Penanganan Pengecualian (Exception)

- ❖ Pengecualian digunakan untuk mengubah alur program yg normal jika terjadi kesalahan (error) tertentu. Kondisi tersebut disebut pengecualian.
- ❖ Pada dasarnya yang terjadi saat exception dipicu adalah:
  1. Keadaan kode yang paling baru (sebelum terjadi kesalahan) disimpan
  2. Eksekusi kode akan beralih ke fungsi handler exception (custom) yang telah ditentukan
  3. Bergantung pada situasinya, handler kemudian dapat melanjutkan eksekusi dari status kode yang disimpan, mengakhiri eksekusi program atau melanjutkan program dari lokasi yang berbeda dalam kode.

## Penggunaan Exception dasar

Penulisan Exception yang standard harus mengandung:

- ❖ try - Fungsi yang menggunakan exception harus berada di dalam blok "try". Jika exception tidak terpicu, kode akan dilanjutkan seperti biasa. Namun jika exception terpicu, exception akan di-"throw".
- ❖ throw - Ini adalah bagaimana Anda memicu exception. Setiap "throw" harus memiliki setidaknya satu "catch".
- ❖ catch - Blok "catch" mengambil exception dan membuat objek yang berisi informasi dari execption.

## Contoh Penggunaan Exception

```
<?php
//create function with an exception
function checkNum($number) {
    if($number>1) {
        throw new Exception("Value must be 1 or below");
    }
    return true;
}

//trigger exception in a "try" block
try {
    checkNum(2);
    //If the exception is thrown, this text will not be shown
    echo 'If you see this, the number is 1 or below';
}

//catch exception
catch(Exception $e) {
    echo 'Message: ' . $e->getMessage();
}
?>
```

Apabila dijalankan akan menghasilkan

Message: Value must be 1 or below

## Error Reporting

- Fungsi error\_reporting () menentukan kesalahan mana yang dilaporkan.
- PHP memiliki banyak level kesalahan, dan menggunakan fungsi ini menetapkan level kesalahan tersebut untuk kode program yang sedang dijalankan.
- Syntax:

---

```
error_reporting(Level);
```

---

## Error Reporting (Contoh)

```
<?php
    // Turn off error reporting
    error_reporting(0);

    // Report runtime errors
    error_reporting(E_ERROR | E_WARNING | E_PARSE);

    // Report all errors
    error_reporting(E_ALL);

    // Same as error_reporting(E_ALL);
    ini_set("error_reporting", E_ALL);

    // Report all errors except E_NOTICE
    error_reporting(E_ALL & ~E_NOTICE);
?>
```

## Kinerja Situs Web

- Ketika berbicara tentang kinerja aplikasi web, kita berbicara tentang kinerja web atau kinerja runtime.
- Kinerja web sebagai ukuran waktu mulai dari saat pengguna akhir meminta konten hingga kapan konten itu tersedia di perangkat pengguna.
- Kinerja runtime sebagai indikasi seberapa responsif aplikasi terhadap input pengguna saat runtime.

## Menggunakan Ukuran Performansi dalam Menuliskan kode sumber

- Mengukur Kinerja Program (kecepatan)

```
<?php
$awal = microtime(true);

// Berhenti untuk sesaat atau berisi blok kode program yg diukur
usleep(1000);

$akhir = microtime(true);
$waktu = $akhir - $awal;

echo "Program berjalan selama $waktu detik\n";
?>
```

## Menggunakan Ukuran Performansi dalam Menuliskan kode sumber

- Mengukur Kinerja Program (penggunaan memory)

```
<?php
/*
Blok kode program yang akan diukur penggunaan memorinya
*/
$penggunaan_memory = memory_get_usage();
echo "Program menggunakan memory sebesar
$penggunaan_memory bytes\n";
?>
```

## Menggunakan Ukuran Performansi dalam Menuliskan kode sumber

- Menghitung efisiensi program

$$Efisiensi = \left( \frac{\text{Kinerja setelah efisiensi}}{\text{(Kinerja sebelum efisiensi)}} \right) \times 100\%$$

## Menulis kode PHP dengan Efisien

- **Menggunakan Fungsi bawaan PHP**

Menggunakan fungsi bawaan PHP lebih cepat dibanding menulis fungsi sendiri.

Referensi fungsi bawaan PHP dapat dilihat di manual PHP <https://www.php.net/manual/en/funcref.php>

- **Membuat Kelas hanya jika diperlukan**

Kelas dan method dibuat jika memang benar-benar diperlukan (digunakan kembali di banyak kode).

- **Menutup Koneksi**

Menutup koneksi basisdata setelah tidak dibutuhkan akan menghemat memory.

## Menulis kode PHP dengan Efisien

- **Menggunakan petik tunggal**

Ketika menggunakan string, penggunaan petik tunggal ( ‘ ’ ) lebih cepat dari pada petik ganda ( “ ” ). Petik ganda akan memeriksa keberadaan variabel di dalamnya, sehingga butuh waktu lebih lama.

### Lebih lambat

```
$awal = microtime(true);  
$i = 0;  
while($i < 1000) {  
    $tmp[] = "";  
    ++$i;  
  
}  
  
$akhir = microtime(true);  
$waktu = $akhir - $awal;  
echo "Membutuhkan $waktu detik\n";
```

### Lebih cepat

```
$awal = microtime(true);  
$i = 0;  
while($i < 1000) {  
    $tmp[] = '';  
    ++$i;  
  
}  
  
$akhir = microtime(true);  
$waktu = $akhir - $awal;  
echo "Membutuhkan $waktu detik\n";
```

## Menulis kode PHP dengan Efisien

- Mengurangi perhitungan yang tidak perlu

Menghitung dan memberikan nilai ke variabel di awal, lebih cepat dari pada menghitungnya beberapa kali dimana perhitungan tersebut diperlukan.

### Lebih lambat

```
$awal = microtime(true);  
$arrA = array(1,2,3,4,5,6,7,8,9);  
for( $i=0; i< count($arrA); $i++) {  
    echo count($arrA);  
}  
  
$akhir = microtime(true);  
$waktu = $akhir - $awal;  
echo "Membutuhkan $waktu detik\n";
```

### Lebih cepat

```
$awal = microtime(true);  
$arrA = array(1,2,3,4,5,6,7,8,9);  
$len = count($arrA);  
for( $i=0; i< $len; $i++) {  
    echo $len;  
}  
  
$akhir = microtime(true);  
$waktu = $akhir - $awal;  
echo "Membutuhkan $waktu detik\n";
```

## Menulis kode PHP dengan Efisien

- Menggunakan function `isset()`

Gunakan `isset()` jika memungkinkan untuk memeriksa lebih besar dari 0. Hindari menggunakan `count()`, `strlen()`, `sizeof()`.

Lebih lambat

```
if(count($returnValue) > 0) {  
    // lakukan sesuatu  
}
```

Lebih cepat

```
if(isset($returnValue)) {  
    // lakukan sesuatu  
}
```

## Menulis kode PHP dengan Efisien

- Menggunakan Swith Case dibanding If

Menggunakan pernyataan case lebih efisien dibandingkan struktur if/else untuk mengerjakan pekerjaan yang sama.

### Lebih lambat

```
if($a == 0){  
    // lakukan tugas a  
}else if($a == 1){  
    // lakukan tugas b  
}else{  
    // lakukan tugas c  
}
```

### Lebih cepat

```
Switch ($a) {  
    case 0:  
        // lakukan tugas a  
    case 1:  
        // lakukan tugas b  
    default:  
        // lakukan tugas c  
}
```

## Membuat Proses Delete

Silahkan buat script baru dengan nama delete.php dan ketik script ini :

```
1 <?php
2 include 'koneksi.php';
3 // menyimpan data id kedalam variabel
4 $id_mhs = $_GET['id_mhs'];
5 // query SQL untuk insert data
6 $query="DELETE from mahasiswa where id_mhs='$id_mhs'";
7 mysqli_query($koneksi, $query);
8 // mengalihkan ke halaman index.php
9 header("location:index.php");
10 ?>
```

## Menggunakan Library atau Komponen Pre-Existing

### Deskripsi Singkat mengenai Topik:

Pelatihan ini berhubungan dengan penggunaan komponen-komponen *reuse* (yang dapat dipergunakan secara berulang) untuk mendukung pengembangan aplikasi yang efisien.

### Tujuan Pelatihan:

Peserta pelatihan memiliki sikap, pengetahuan dan keterampilan yang dibutuhkan untuk dapat menerapkan pengelolaan *log* aplikasi dalam membuat program/aplikasi.

### Materi yang akan disampaikan:

1. Pemilihan library
2. Integrasi Library
3. Pembaharuan Library

### Tugas:

Memasang library dan menggunakan dalam program

### Outcome/Capaian Pelatihan:

1. Peserta dapat melakukan pemilihan unit-unit *reuse* yang potensial
2. Peserta dapat melakukan integrasi library atau komponen pre-existing dengan source code yang ada
3. Peserta dapat melakukan pembaharuan library atau komponen pre-existing yang digunakan

## Library atau Komponen Pre-Existing

- ❖ **Reuse** adalah penggunaan kembali suatu kesatuan kode tanpa menulis ulang atau mengubah kode tersebut.
- ❖ **Library** adalah pemaketan kode yang dapat digunakan untuk spesifikasi tertentu.
- ❖ **Pre-Existing** merupakan istilah untuk library atau komponen yang sudah ada dari sebelumnya.

## Keuntungan menggunakan Library

- ❖ Mengurangi redundansi(pengulangan) penulisan kode.
- ❖ Menghemat waktu.
- ❖ Menghemat sumber daya.

## Jenis-jenis Library

- ❖ Pembuatan User interface (Bootstrap, DataTables, JQueryUI)
- ❖ Image editing (Imagine, PHP Graphics Works, Zebra Image)
- ❖ impor/ekspor ke bentuk dokumen tertentu (TCPDF, PHPOffice)
- ❖ algoritma kompleks (Artificial Neural Network Library, Artificial Neural Network Library)
- ❖ Dan lain-lain.

## Lisensi Library

Hak yang diberikan	Public domain	Permissive FOSS license	CopyleftFOSS license	Freeware/ Shareware/ Freemium	Proprietary license	Rahasia dagang
Contoh lisensi		BSD License, MIT License	GPL, LGPL			
Hak Cipta Dilindungi	Tidak	Iya	Iya	Iya	Iya	Sangat Ketat
Hak untuk ditampilkan	Iya	Iya	Iya	Iya	Iya	Tidak
Hak untuk menyalin	Iya	Iya	Iya	Seringkali Iya	Tidak	Tidak
Hak untuk memodifikasi	Iya	Iya	Iya	Tidak	Tidak	Tidak
Hak untuk mendistribusikan	Iya	Ya, di bawah lisensi yang sama	Ya, di bawah lisensi yang sama	Seringkali Iya	Tidak	Tidak
hak untuk sublisensi	Iya	Iya	Tidak	Tidak	Tidak	Tidak
Contoh software	SQLite, Image J	Apache web server, ToyBox	Linux kernel, GIMP, OBS	Winamp, League of Legends	Windows, Spotify	Online Games PlayStation Network

Lisensi library biasanya terdapat dalam direktori library disimpan dalam file “LICENSE”

## Mendapatkan Library

- ❖ Mengunduh file langsung dari website penyedia library
- ❖ Menggunakan library manager (Composer)

## Mengunduh file langsung

- ❖ Ketergantungan antar library atau unit dilakukan manual.
- ❖ Memperbarui library dilakukan dengan memeriksa di website/repository.
- ❖ Tidak membutuhkan software tambahan.

## Library manager

- ❖ Ketergantungan antar library atau unit dapat dilakukan secara otomatis
- ❖ Memperbarui library dapat dilakukan dengan mudah
- ❖ Manajer Ketergantungan (Dependency Manager) untuk PHP Composer (<https://getcomposer.org>)

## Instalasi Composer

### ❖ Instalasi di Windows

1. Pastikan terlebih dahulu PHP atau paket XAMPP sudah terinstall.
2. Download file installer Composer di <https://getcomposer.org/Composer-Setup.exe>
3. Ikuti petunjuk instalasi.
4. Composer akan meminta lokasi file PHP berada. Biasanya ada dalam folder C:\XAMPP\php\php.exe
5. Setelah proses instalasi selesai, composer dapat dijalankan melalui command prompt (CMD).

## Contoh Kasus: Php Spreadsheet Library

- ❖ <https://phpspreadsheet.readthedocs.io>
- ❖ PhpSpreadsheet merupakan bagian dari **PhpOffice**.
- ❖ Digunakan untuk membuat dan membaca file spreadsheet (Microsoft Excell, OpenOffice Calc)
- ❖ Dokumentasi API: <https://phpoffice.github.io/PhpSpreadsheet/master/>

## Instalasi Library dengan Composer

1.Buka command prompt/terminal.

2.Pindah ke direktori tempat project PHP dibuat. Misal di Windows:

```
# cd C:\xampp\htdocs\dts
```

3.Jalankan perintah:

```
# composer require phoffice/phpspreadsheet
```

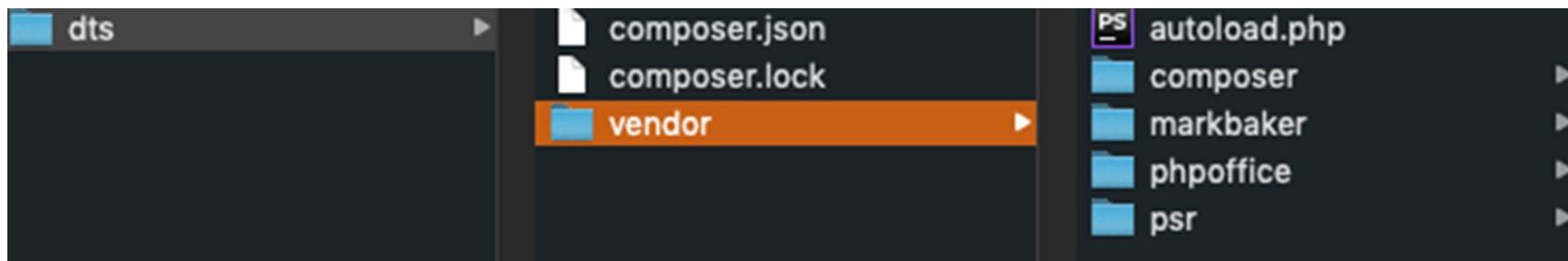
## Instalasi Library dengan Composer

## 4. Proses instalasi

```
Using version ^1.6 for phpoffice/phpspreadsheet
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 4 installs, 0 updates, 0 removals
- Installing markbaker/matrix (1.1.4): Loading from cache
- Installing markbaker/complex (1.4.7): Loading from cache
- Installing psr/simple-cache (1.0.1): Loading from cache
- Installing phpoffice/phpspreadsheet (1.6.0): Loading from cache
phpoffice/phpspreadsheet suggests installing mpdf/mpdf (Option for :)
phpoffice/phpspreadsheet suggests installing dompdf/dompdf (Option :)
phpoffice/phpspreadsheet suggests installing tecnickcom/tcpdf (Option :)
phpoffice/phpspreadsheet suggests installing jpgraph/jpgraph (Option :)
Writing lock file
Generating autoload files
```

## Instalasi Library dengan Composer

### 5. Hasil instalasi



## Memperbarui Library

- ❖ Memperbarui library dengan composer, dapat dilakukan dengan perintah yang dijalankan dalam direktori project:

```
# composer update
```

- ❖ Proses update composer

```
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 7 installs, 82 updates, 6 removals
- Removing npm-asset/floatthead (2.1.1)
- Removing composer/installers (v1.5.0)
- Removing bower-asset/bootstrap3-dialog (v1.35.4)
- Removing phpunit/phpunit-mock-objects (6.0.1)
- Removing kartik-v/yii2-sortable (v1.2.0)
- Removing kartik-v/yii2-dynagrid (v1.4.8)
- Updating yiisoft/yii2-composer (2.0.5 => 2.0.7): Loading from cache
- Updating bower-asset/jquery (3.2.1 => 3.3.1): Loading from cache
- Updating cebe/markdown (1.1.2 => 1.2.1): Loading from cache
  813 more files modified, choose "v" to view the full list
Discard changes [y,n,v,d,s,?]?
```

## Memperbarui Library

- ❖ Untuk menghindari penggunaan komponen yang sudah kadaluarsa (obsolete), library perlu diperbarui.
- ❖ Pembaharuan library yang diunduh secara langsung, proses pembaharuan dilakukan dengan memeriksa langsung ke website/repository library terkait.

## Pertimbangan?

- ❖ Apakah itu memperkenalkan perubahan yang anda butuhkan?
- ❖ Apa manfaat yang Anda dapatkan dari peningkatan?
- ❖ Apa yang terjadi apabila Anda tidak melakukan update?
- ❖ Apakah manfaatnya lebih besar daripada risiko atau investasi waktu?

Pertimbangan...

## ❖ Apakah itu memperkenalkan perubahan yang anda butuhkan?

- Perubahan pada Bootstrap biasanya untuk memperbaiki bug di versi sebelumnya.
- Biasanya juga terdapat fitur-fitur baru.
- Tampilan dan performa yang lebih baik, mengikuti perkembangan teknologi dan gadget.

Pertimbangan...

## ❖ Apa manfaat yang Anda dapatkan dari peningkatan?

- Perbaikan pada bug-bug di versi sebelumnya
- Fitur-fitur baru yang lebih stabil dan fleksibel
- Perbaikan pada system yang digunakan (grid system, js, css, dll)
- Mendapatkan modul kode yang lebih rapi

Pertimbangan...

## ❖ Apa yang terjadi apabila Anda tidak melakukan update?

- Periksa dependencies dari website anda
- Periksa perkembangan teknologi yang digunakan oleh website anda
- Periksa permintaan dari klien yang menggunakan jasa anda

## Pertimbangan...

### ❖ Apakah manfaatnya lebih besar daripada risiko atau investasi waktu?



- Bug diperbaiki
- Fitur-fitur yang ditawarkan sangat menarik dan menjadi nilai tambah bagi perusahaan
- Biaya yang dibutuhkan sesuai atau dibawah budget
- Bila tidak diupdate ada beberapa script yang menampilkan error



- Tidak ada bug di versi lama
- Fiturnya tidak terlalu dipakai
- Tidak ada nilai tambah bagi perusahaan
- Biaya yang dibutuhkan berada diatas budget
- Tidak ada dependency yang berarti

# Menghubungkan program dengan Library PhpSpreadSheets

```
<?php
// impor library PhpSpreadsheet
require 'vendor/autoload.php';

// menggunakan kelas
use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

// membuat objek baru dari kelas Spreadsheet
$spreadsheet = new Spreadsheet();

// mendapatkan sheet aktif
$sheet = $spreadsheet->getActiveSheet();

// memberikan nilai ke kolom A baris ke-1
$sheet->setCellValue('A1', 'Hello World !');

// menulis file spreadsheet
$writer = new Xlsx($spreadsheet);
DTS $writer->save('hello world.xlsx');
```

Panduan lengkap di:

<https://phpspreadsheet.readthedocs.io/en/latest/>

# Library Bootstrap

## Pendahuluan



- ❖ Bootstrap merupakan framework untuk membangun desain web secara responsif dan cepat.
- ❖ Bootstrap menyediakan HTML, CSS dan Javascript siap pakai dan mudah untuk dikembangkan.

## Kelebihan Bootstrap

- ❖ Penggunaan bootstrap sangat menghemat waktu
- ❖ Tampilan bootstrap yang sudah cukup terlihat modern
- ❖ Mobile Friendly yang maksudnya tampilan bootstrap sudah sangat responsive, yaitu tampilan bootstrap sudah mendukung segala jenis resolusi, baik itu pc, laptop, tablet dan smartphone
- ❖ Sangat ringan karena bootstrap dibuat dengan sangat terstruktur

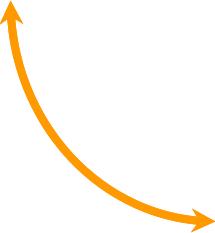
## Persiapan

- ❖ Anda Harus memiliki file distribusi Bootstrap
- ❖ Download jQuery Library Untuk Bekerja Offline
- ❖ Anda harus memiliki text editor.

[Download Bootstrap 4 \(v4.2.1\)](#)[Download jQuery \(v3.3.1\)](#)

## Mulai

- ❖ Buat Folder Baru
- ❖ Copy seluruh file Bootstrap ke dalam Folder Baru Tersebut
- ❖ Copy file jQuery ke dalam folder js
- ❖ Buat file HTML



```
<!DOCTYPE html>
<html>
<head>
    <title>Belajar Bootstrap</title>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css">
    <script type="text/javascript" src="js/jquery.js"></script>
    <script type="text/javascript" src="js/bootstrap.js"></script>
</head>
<body>
    <h1>Cara Menggunakan Bootstrap | DigiTalent</h1>
    <button class="btn btn-danger">TOMBOL MERAH</button>
    <button class="btn btn-primary">TOMBOL BIRU</button>
</body>
</html>
```

## Struktur File

```
Belajar_Bootstrap/
├── index.html
└── css/
    ├── bootstrap.css
    ├── bootstrap.css.map
    ├── bootstrap.min.css
    ├── bootstrap.min.css.map
    ├── bootstrap-grid.css
    ├── bootstrap-grid.css.map
    ├── bootstrap-grid.min.css
    └── bootstrap-grid.min.css.map
    ├── bootstrap-reboot.css
    ├── bootstrap-reboot.css.map
    ├── bootstrap-reboot.min.css
    └── bootstrap-reboot.min.css.map
└── js/
    ├── bootstrap.bundle.js
    ├── bootstrap.bundle.js.map
    ├── bootstrap.bundle.min.js
    └── bootstrap.bundle.min.js.map
    ├── bootstrap.js
    ├── bootstrap.js.map
    └── bootstrap.min.js
    └── bootstrap.min.js.map
    └── jquery-3.3.1.min.js
```

## Penjelasan

```
<!DOCTYPE html>
<html>
<head>
    <title>Belajar Bootstrap</title>
    <link rel="stylesheet" type="text/css"
href="css/bootstrap.css">
        <script type="text/javascript"
src="js/jquery.js"></script>
        <script type="text/javascript"
src="js/bootstrap.js"></script>
</head>
<body>
    <h1>Cara Menggunakan Bootstrap | Digitalent</h1>
    <button class="btn btn-danger">TOMBOL MERAH</button>
    <button class="btn btn-primary">TOMBOL BIRU</button>
</body>
</html>
```

Memanggil file bootstrap dari html

Memanggil file jquery.js dan bootstrap.js dari html

Ingat, letakkan jquery.js pada bagian atas karena bootstrap bekerja menggunakan bantuan jquery bukan jquery menggunakan bantuan bootstrap

## Membuat Navigasi Tabs

- ❖ Adalah class yang disediakan oleh bootstrap untuk membuat navigasi yang berbentuk tabs

```
<body>  
  
<div class="container">  
  <h1> Membuat Navigasi Tabs </h1> <br/>  
  
  <ul class="nav nav-tabs">  
    <li class="active"><a href="#">Home</a></li>  
    <li><a href="#">Menu 1</a></li>  
    <li><a href="#">Menu 2</a></li>  
    <li><a href="#">Menu 3</a></li>  
  </ul>  
  
</div>  
  
</body>
```

**Navigasi tabs** dibuat menggunakan class **nav nav-tabs**; **class nav** -> untuk mendefinisikan bahwa yang kita buat adalah navigasi. dan **nav-tabs** -> digunakan untuk membuat navigasi dengan bentuk tabs.



## Membuat Navigasi Pills

- ❖ Adalah class yang disediakan oleh bootstrap untuk membuat navigasi yang berbentuk pil.

```
<body>
<div class="container">
<h1>Membuat Navigasi Pills Bootstrap </h1>
<br/>
<ul class="nav nav-pills">
<li class="active"><a href="#">Home</a></li>
<li><a href="#">Menu 1</a></li>
<li><a href="#">Menu 2</a></li>
<li><a href="#">Menu 3</a></li>
</ul>
</div>
</body>
```

Navigasi tabs dibuat menggunakan class **nav nav-tabs**; class **nav** -> untuk mendefinisikan bahwa yang kita buat adalah navigasi. dan **nav-pills** -> digunakan untuk membuat navigasi dengan bentuk pil.



## Membuat Form

- ❖ Bootstrap menyediakan class khusus lagi untuk mendesain form yang terlihat sangat modern dan rapi.
- ❖ Ada tiga macam model form;
  1. **Form dengan model vertical:** form dengan model vertikal ini merupakan form default bootstrap
  2. **Form dengan model inline:** form dengan model inline adalah sebuah form yang label dan form inputnya terletak sebaris.
  3. **Form dengan model horizontal:** form dengan model horizontal ini merupakan form yang memiliki bentuk horizontal. artinya posisi label dan input form terletak horizontal.

## Form Vertical

```
<body>

    <div class="container">
        <h1>Membuat Form dengan Bootstrap </h1>
    <form>
        <div class="form-group">
            <label for="nama">Nama Anda:</label>
            <input type="text" class="form-control">
        </div>
        <div class="form-group">
            <label for="alamat">Alamat anda:</label>
            <input type="text" class="form-control" id="alamat">
        </div>
        <button type="submit" class="btn btn-default">Submit</button>
    </form>
</div>
```

DTS 2021

Untuk membuat form dengan bootstrap yang pertama harus di perhatikan adalah : **class form-group** -> untuk membalut element sebuah form seperti label dan formnya, dan kemudian pada <form> berikan class form-control

## Form Vertical

# Membuat Form dengan Bootstrap |

Nama Anda:

Alamat anda:

Submit

## Form Inline

```
<body>

<div class="container">
<h1>Membuat Form dengan Bootstrap | <h1>
<form class="form-inline">
<div class="form-group">
<label for="nama">Nama Anda:</label>
<input type="text" class="form-control">
</div>
<div class="form-group">
<label for="alamat">Alamat anda:</label>
<input type="text" class="form-control" id="alamat">
</div>
<button type="submit" class="btn btn-primary">Simpan</button>
</form>
</div>
</body>
```

Untuk membuat form inline dengan bootstrap harus di perhatikan adalah: class form-group -> untuk membalut element sebuah form seperti label dan formnya, dan kemudian pada <form> berikan class form-inline

## Form Inline

# Membuat Form dengan Bootstrap |

Nama Anda:

Alamat anda:

**Simpan**

## Form Horizontal

```
<body>
<div class="container">
  <h1>Membuat Form dengan Bootstrap |</h1>
  <form class="form-horizontal">
    <div class="form-group">
      <label class="control-label col-sm-2" for="nama">Nama
      Anda:</label>
      <div class="col-sm-10">
        <input type="text" class="form-control" />
      </div>
    </div>
    <div class="form-group">
      <label class="control-label col-sm-2" for="alamat">Alamat
      anda:</label>
      <div class="col-sm-10">
        <input type="text" class="form-control" id="alamat" />
      </div>
    </div>
    <button type="submit" class="btn btn danger">Simpan</button>
  </form>
</div>
</body>
```

Untuk membuat form inline dengan bootstrap harus di perhatikan adalah :

class form-group -> untuk membalut element sebuah form seperti label dan formnya,  
dan kemudian pada <form> berikan class form-horizontal

## Form Horizontal

### Membuat Form dengan Bootstrap |

Nama Anda:

Alamat anda:

Simpan

## Reference

1. PHP Tutorial, diakses dari laman <https://www.w3schools.com/php/>, pada 24 Februari 2021
2. Tutorial Belajar PHP DuniaIlkom, diakses dari laman <https://www.duniaIlkom.com/tutorial-belajar-php-dan-index-artikel-php/>, pada 24 Februari 2021
3. Anonymous. (n.d.). *MySQL Reference Manual*. Retrieved from <http://downloads.mysql.com/docs/>.
4. <http://getbootstrap.com>
5. <http://getcomposer.org>

## Tools / Lab Online

1. Sublime Text
2. Web Browser (Google Chrome/Mozilla Firefox)
3. XAMPP

## Summary

- ❖ PHP digunakan untuk membuat website dinamis
- ❖ PHP dapat digabungkan dengan database server MySQL untuk penyimpanan data
- ❖ Penggunaan library memudahkan untuk membuat program

## Team Teaching

## Quiz / Games

# #JADIJAGOANDIGITAL TERIMA KASIH



[digitalent.kominfo](#)



[DTS\\_kominfo](#)



[digitalent.kominfo](#)



[digital talent scholarship](#)