

DOKUMENTASI BELAJAR PHP

By Hilman Apriadi





Source: Grup FB Waktunya Memprogram

Mengapa Belajar PHP

- Tuntutan dari sekolah (Sbnrnya males belajar PHP wkwk)
- Masih banyak job yang membutuhkan PHP di Indonesia (saat ini)
- Bisa bikin web dinamis

Kelebihan PHP

- Open Source
- Populer dan Memiliki komunitas yang besar
- Koneksi ke Database dengan baik
- Bisa dapet Lambo

Kekurangan PHP

- Udah mulai kuno (katanya)
- Keamanan yang kurang jika tidak dikonfigurasi dengan baik
- Awas banyak haters-nya juga

Final Project

- Membuat to-do-list sederhana
- CMS Sederhana
- Sistem pemesanan tiket

Resource Belajar

- W3School (Referensi materi)
- ChatGPT (Buat nanya² kalo kebingungan)

Rincian Materi PHP

1. Pengenalan

- Sejarah dan kegunaan PHP
- Instalasi PHP dan web server lokal

2. Dasar-Dasar

- Variable dan Tipe data
- Operator dan Expression
- Struktur kendali (pengkondisian, perulangan)
- Function

3. Pengelolaan Forms

- Menerima input pengguna melalui forms HTML
- Validasi data masukan
- Menyimpan dan mengolah data masukan menggunakan PHP

4. Penanganan File

- Read & Write file teks
- Upload & Download file
- Manipulasi file (CRUD)

5. Koneksi ke Database

- Menghubungkan PHP dengan database MySQL
- Menjalankan query database (SELECT, INSERT, UPDATE, DELETE)
- Melakukan operasi CRUD pada data

6. Pengelolaan Session dan Cookies

- Membuat dan mengelola sesi pengguna
- Menyimpan data pengguna dalam cookies
- Menerapkan keamanan pada sesi dan cookies

7. Pengolahan Gambar

- Mengunggah dan memanipulasi gambar
- Mengubah ukuran, memotong, dan memodifikasi gambar menggunakan PHP

8. Framework PHP

- Memahami dan menggunakan framework PHP populer seperti Laravel, CodeIgniter, atau Symfony
- Menerapkan konsep MVC (Model-View-Controller) dalam pengembangan web menggunakan framework

1. Pengenalan

PHP (Hypertext Preprocessor) adalah bahasa pemrograman yang populer dan banyak digunakan untuk pengembangan aplikasi web.

A. Sejarah dan Kegunaan

PHP pertama kali dikembangkan pada tahun 1994 oleh Rasmus Lerdorf sebagai sekumpulan skrip untuk mengelola website pribadinya. Skrip-skrip tersebut kemudian diperluas dan berkembang menjadi bahasa pemrograman server-side yang fleksibel dan kuat. PHP menjadi open-source pada tahun 1995, memungkinkan pengembang lain untuk berkontribusi dan mengembangkan PHP lebih lanjut. Sejak itu, PHP telah mengalami banyak versi dan pembaruan untuk meningkatkan kinerja dan fitur-fiturnya.

PHP Memiliki beberapa kegunaan sebagai berikut:

- PHP digunakan untuk pengembangan aplikasi web dinamis.
- PHP memungkinkan interaksi dengan basis data, seperti MySQL, PostgreSQL, dan SQLite.
- PHP digunakan dalam pembuatan sistem manajemen konten (CMS) seperti WordPress dan Joomla.
- PHP mendukung pembangunan aplikasi e-commerce.
- Dll.

B. Instalasi PHP dan Web Server lokal

Karena posisi saya yang gapunya lepi/PC, alhasil pake Termux buat instalasi PHP-nya. Sedih sih tapi ya mau gimna lagi :(

Berikut merupakan langkah-langkahnya:

1. Download aplikasi Termux via F-Droid. Download Langsung:

https://f-droid.org/repo/com.termux_118.apk

2. Setelah Di-install, jalankan commands berikut:

➤ Ketik `pkg update -y` lalu enter

```
4:11 AM ...0.0KB/s 47
IRC channel: #termux on libera.chat
Working with packages:
* Search packages: pkg search <query>
* Install a package: pkg install <package>
* Upgrade packages: pkg upgrade
Subscribing to additional repositories:
* Root: pkg install root-repo
* Unstable: pkg install unstable-repo
* X11: pkg install x11-repo
Report issues at https://termux.com/issues
The Google Play version of the Termux app no longer
receives updates. For more information, visit:
https://wiki.termux.com/wiki/Termux_Google_Play
~ $ pkg update -y
```

➤ Jika muncul seperti ini, ketik `y`, lalu setelah itu enter, dan jika muncul lagi berikutnya, klik `y` lagi, hingga prosesnya selesai

```
on
The default action is to keep your current version.
*** openssl.cnf (Y/I/N/O/D/Z) [default=N] ? y
Progress: [ 36%] [#####.....]

ESC / - HOME ↑ END PGUP
= CTRL ALT ← ↓ → PGDN
```

- Jika proses sebelumnya sudah selesai, selanjutnya install PHP-nya dengan cara ketik command

pkg install php -y

```
Setting up lsof (4.98.0) ...
Setting up nano (7.2) ...
update-alternatives: using /data/data/com.termux/files/usr/bin/nano to provide /data/data/com.termux/files/usr/bin/editor (editor) in auto mode
Setting up debianutils (5.7-1) ...
Setting up dos2unix (7.5.0) ...
- $ pkg install php -y
```

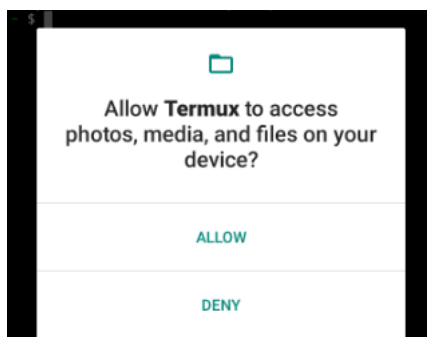
ESC / - HOME ↑ END PGUP
CTRL ALT ← ↓ → PGDN

- ketik **php -v** memastikan bahwa PHP telah terinstall dengan baik, dan kita bisa melihat versi PHP-nya

```
Setting up php (8.2.7-1) ...
- $ php -v
PHP 8.2.7 (cli) (built: Jul  2 2023 03:41:31) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.2.7, Copyright (c) Zend Technologies
- $
```

ESC / - HOME ↑ END PGUP
CTRL ALT ← ↓ → PGDN

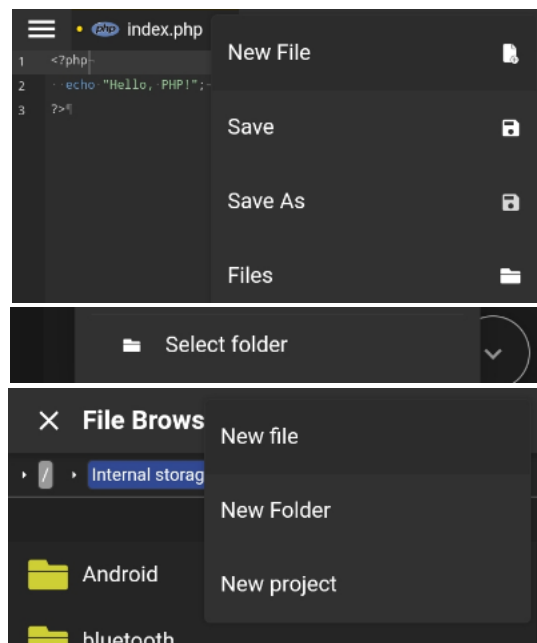
- Ketik **termux-setup-storage** untuk mengizinkan storage, lalu jika muncul pop up seperti ini, klik **ALLOW**

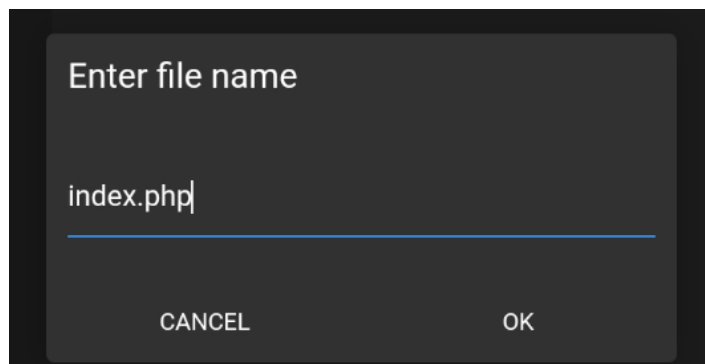
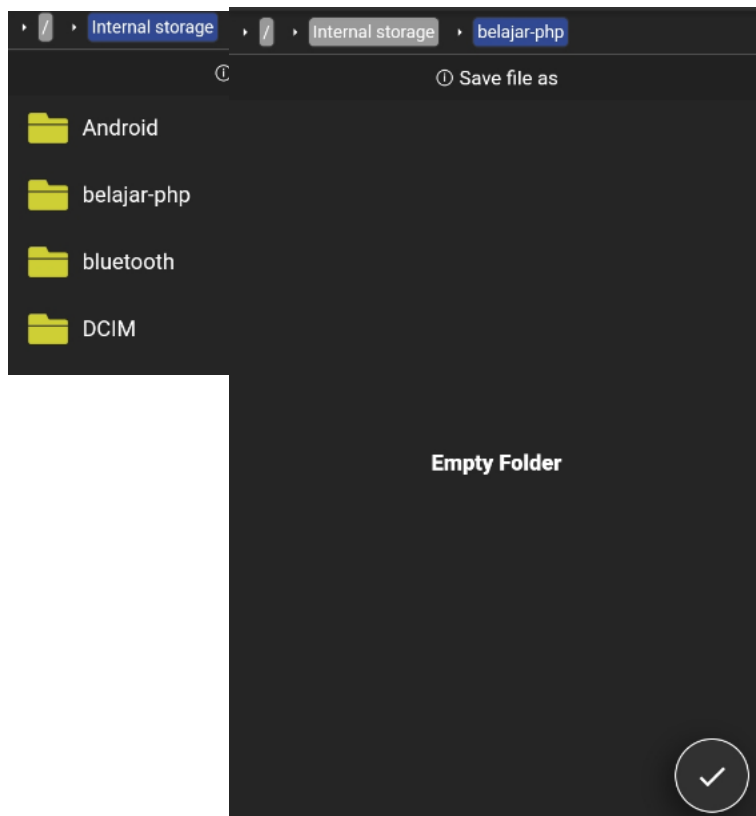


- Buka kode editor kalian (di sini saya pake aplikasi Acode btw) dan buat file PHP baru dengan nama **index.php** dan selanjutnya isi file-nya dengan kode sederhana seperti berikut

A screenshot of the Acode code editor interface. At the top, there's a tab labeled 'php index.php' with a close button. Below the tab, the editor shows three lines of PHP code: line 1: <?php, line 2: echo "Hello, PHP!";, and line 3: ?>. The editor has a dark theme and a blue vertical bar on the right side.

- Klik titik tiga pojok kanan atas => Save as => select folder => klik ikon (+) di kanan atas => new folder => nama folder barunya "belajar-php" => klik tombol ceklis di kanan bawah





- Buka termux lagi, lalu ketik command **ls**

```
~ $ ls
storage
```

- Untuk masuk ke folder "**belajar-php**" yang sudah dibuat sebelumnya, maka harus mengetik

```
cd storage/shared/belajar-php
```

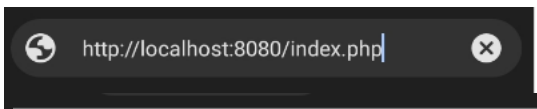
```
~ $ cd storage/shared/belajar-php
```

- Untuk menjalankan web server, gunakan command

```
php -S localhost:8080
```

```
~/.../shared/belajar-php $ php -S localhost:8080  
[Mon Jul 3 04:32:52 2023] PHP 8.2.7 Development Server (http://localhost:8080) started
```

- Buka browser, dan ketik **localhost:8080/index.php**



Hello, PHP!

Selamattt kode PHP pertama berhasil dijalankan

Note: "Kalo ada lepi/PC mending pake XAMPP aja coy"

2. Dasar-Dasar

Skrip PHP dimulai dengan `<?php` dan diakhir dengan `?>`

A. Variable

Variabel adalah tempat untuk menyimpan nilai atau data.

Syarat penulisan variable yang benar di PHP adalah:

- Nama variabel harus diawali dengan tanda dollar "\$" diikuti oleh karakter huruf atau garis bawah "_".
- Nama variabel bersifat case-sensitive, artinya huruf besar dan huruf kecil dianggap berbeda.
- Nama variabel tidak boleh dimulai dengan angka, tetapi bisa mengandung angka setelah karakter pertama.
- Nama variabel tidak boleh mengandung spasi atau karakter khusus lainnya, kecuali garis bawah "_".

Contoh:

```
<?php
    $nama = "John Doe";
    $umur = 25;
    $_alamat = "Jalan Raya No. 123";
    $nilai = 90;
?>
```

B. Tipe Data

PHP Memiliki beberapa Tipe Data sebagai berikut:

- | | | |
|------------|------------|-------------|
| 1) String | 4) Boolean | 7) NULL |
| 2) Integer | 5) Array | 8) Resource |
| 3) Float | 6) Object | |

1) String

Teks atau urutan karakter, diapit oleh tanda kutip (" ") atau tanda kutip tunggal (' ').

```
$nama = "John Doe";  
$alamat = 'Jalan Raya No. 123';  
$pesan = "Halo, apa kabar?";
```

2) Integer

Angka bilangan bulat (bilangan tanpa desimal) seperti 1, 10, atau -5.

```
$umur = 25;  
$jumlah_barang = 10;  
$nilai = -5;
```

3) Float

Angka dengan desimal (floating-point number) seperti 3.14, 2.5, atau -0.75.

```
$pi = 3.14;  
$harga = 2.5;  
$nilai_decimal = -0.75;
```

4) Boolean

Tipe data untuk kebenaran, yang hanya memiliki dua nilai, yaitu true (benar) dan false (salah).

```
$is_valid = true;  
$is_logged_in = false;.
```

5) Array

Untuk menyimpan banyak nilai dalam satu variabel. Array dapat berisi tipe data apa pun, termasuk tipe data lainnya, seperti integer, string, boolean, dll.

```
$nilai_ujian = array(80, 90, 75, 85);  
$hewan = array("kucing", "anjing", "burung");  
$daftar_buah = ["apel", "mangga", "pisang"];
```

6) Object

Tipe data yang menggambarkan sebuah objek, yaitu sebuah entitas yang memiliki properti dan metode tertentu.

```
class Mobil {  
    public $warna;  
    public $kecepatan;  
}  
  
$mobil = new Mobil();  
$mobil->warna = "merah";  
$mobil->kecepatan = 100;
```

7) Null

Tipe data yang hanya memiliki satu nilai, yaitu null. Digunakan untuk menunjukkan bahwa suatu variabel tidak memiliki nilai atau tidak ditetapkan.

```
$nama = null;
```

```
$alamat = null;
```

8) Resource

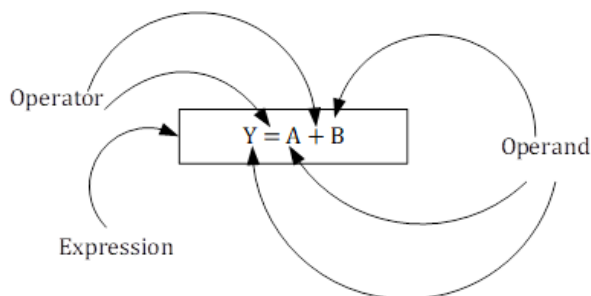
Tipe data khusus yang digunakan untuk merepresentasikan sumber daya eksternal, seperti koneksi database atau file yang dibuka.

c. Operator & Expression

Operator dalam pemrograman adalah simbol atau tanda khusus yang digunakan untuk melakukan operasi atau manipulasi pada data.

Expression (ekspresi) adalah kombinasi dari nilai-nilai, variabel, konstanta, dan operator yang dievaluasi untuk menghasilkan nilai tunggal.

Operand dalam pemrograman merujuk pada nilai atau variabel yang digunakan dalam operasi.



Macam-macam operator di PHP:

1) Operator Aritmatika

Operator	Nama	Contoh
+	Pemjumlahan	$\$x + \y
-	Pengurangan	$\$x - \y
*	Perkalian	$\$x * \y
/	Pembagian	$\$x / \y
%	Sisa bagi	$\$x \% \y
**	Eksponensial	$\$x ** \y

2) Operator Penugasan

Operator	Sama Dengan
$x = y$	$x = y$
$x += y$	$x = x + y$

$x -= y$	$x = x - y$
$x /= y$	$x = x / y$
$x \%= y$	$x = x \% y$

3) Operator Perbandingan

Operator	Nama	Contoh
<code>==</code>	Sama dengan	<code>\$x == \$y</code>
<code>===</code>	Identik	<code>\$x === \$y</code>
<code>!=</code>	Tidak sama dengan	<code>\$x != \$y</code>
<code>!==</code>	Tidak sama dengan identik	<code>\$x !== \$y</code>
<code>></code>	Lebih dari	<code>\$x > \$y</code>
<code><</code>	Kurang dari	<code>\$x < \$y</code>
<code>>=</code>	Lebih dari sama dengan	<code>\$x >= \$y</code>
<code><=</code>	Kurang dari sama dengan	<code>\$x <= \$y</code>

4) Operator Increment / Decrement

Operator	Deskripsi
$x++$	Mengembalikan x , kemudian menaikkan x sebanyak satu.
$++x$	Menaikkan x sebanyak satu, kemudian mengembalikan x .
$x--$	Mengembalikan x , kemudian menurunkan x sebanyak satu.
$--x$	Menurunkan x sebanyak satu, kemudian mengembalikan x .

5) Operator Logika

Operator	Hasil
and	True, jika x dan y keduanya true
or	True, jika salah satu x atau y true
&&	True, jika x dan y keduanya true
	True, jika salah satu x atau y true

!	True, jika \$x tidak true atau false
---	--------------------------------------

6) Operator String

Operator	Contoh	Deskripsi
.	\$txt1 . \$txt2	Menggabungkan \$txt1 dan \$txt2
.=	\$txt1 .= txt2	Menambahkan \$txt2 ke \$txt1

D. Struktur Kendali

1) Pengkondisian

Pengkondisian adalah bagian dari pemrograman yang memungkinkan kita untuk membuat keputusan dalam program berdasarkan kondisi tertentu.

- if

```
$nilai = 80;  
if ($nilai >= 70) {  
    // Jika nilai lebih dari / sama dengan 70  
    echo "Nilai Anda memenuhi syarat lulus.";  
}
```

- if - elseif - else

```
$nilai = 60;  
if ($nilai >= 80) {  
    // Jika nilai lebih dari / sama dengan 80  
    echo "Nilai Anda sangat baik.";  
} elseif ($nilai >= 70) {  
    // Jika nilai lebih dari / sama dengan 70  
    echo "Nilai Anda cukup baik.";  
} else {  
    // Jika nilai kurang dari 60  
    echo "Anda perlu belajar lebih rajin.";  
}
```

- switch case

```
$nilai = "A";  
switch ($nilai) {  
    case "A":  
        // Jika nilai sama dengan "A"  
        echo "Nilai Anda sangat baik.";  
        break;  
    case "B":  
        // Jika nilai sama dengan "B"  
        echo "Nilai Anda baik.";  
        break;  
    case "C":  
        // Jika nilai sama dengan "C"  
        echo "Nilai Anda cukup.";  
        break;  
    default:  
        // Jika tidak ada kasus yang cocok  
        echo "Anda perlu belajar lebih rajin";  
        break;  
}
```

2) Perulangan

Perulangan adalah bagian dari pemrograman yang memungkinkan kita untuk menjalankan serangkaian tugas berulang kali.

- For

```
for ($i = 1; $i <= 5; $i++) {  
    echo $i;  
} // Output: 1 2 3 4 5
```

- While

```
$counter = 1;  
while ($counter <= 5) {  
    echo $counter;  
    $counter++;  
} // Output: 1 2 3 4 5
```

- Do While

```
$counter = 1;  
do {  
    echo $counter;  
    $counter++;  
} while ($counter <= 5);  
// Output: 1 2 3 4 5
```

- Foreach

```
$colors = array("Red", "Green", "Blue");  
foreach ($colors as $color) {  
    echo $color . " ";  
} // Output: Red Green Blue
```

E. Function

Function adalah sebuah blok kode yang dapat digunakan untuk melakukan tugas tertentu secara terpisah dan dapat dipanggil (dijalankan) di berbagai bagian program. Fungsi dapat menerima input (argumen) dan mengembalikan output (nilai) setelah menjalankan serangkaian instruksi.

Terdapat 2 jenis function di PHP, Built-In Functions dan User-Defined Functions:

1) Built-In Functions

Fungsi bawaan (built-in functions) adalah fungsi-fungsi yang sudah disediakan oleh PHP dan dapat langsung digunakan tanpa perlu mendefinisikan ulang. PHP menyediakan banyak fungsi bawaan yang mencakup berbagai fitur dan fungsionalitas, seperti fungsi untuk manipulasi string, manipulasi array, operasi matematika, pengolahan file, pengaturan tanggal dan waktu, dan masih banyak lagi.

Contoh : `date()`, `strlen()`, `substr()`

Liat Referensinya di sini :

https://www.w3schools.com/php/php_ref_overview.asp

2) User-Defined Functions

Fungsi yang dibuat oleh pengguna (user-defined functions) adalah fungsi-fungsi yang didefinisikan oleh pengguna

sesuai dengan kebutuhan mereka. Dalam PHP, kita dapat membuat fungsi sendiri dengan menggunakan statement `function`. Dengan User-Defined Functions, kita dapat mengorganisasi dan mengelompokkan bagian-bagian kode yang sering digunakan ke dalam blok kode terpisah yang dapat dipanggil berulang kali.

```
// Membuat fungsi dengan nama greetings
// $nama adalah parameter
function greetings($nama) {
    echo "Halo, $nama! Selamat datang!";
}
```

```
// Memanggil fungsi
// 'Jhon' adalah argument
greetings('John');
// Output: Halo, John! Selamat datang!
```

Contoh Lain :

```
function addNumbers($a, $b) {
    // mengembalikan hasil dari penjumlahan
    // parameter $a dengan $b
    return $a + $b;
}
echo addNumbers(5, 2);
// Output: 7
```


3. Pengelolaan Forms

A. Menerima input pengguna melalui form HTML

Variable superglobals seperti `$_POST` akan digunakan untuk mengumpulkan data hasil inputan dari form HTML. Superglobals di PHP adalah sebuah variable bawaan dari PHP

Buat file `form.php` di dalam folder belajar-php yang telah dibuat sebelumnya. Isi file nya dengan kode berikut:

```
<form method="post" action="form.php">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>

<?php
// Cek apakah ada permintaan POST yang dikirimkan
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // Ambil nilai yang dikirimkan melalui input dengan name
  $name = $_POST['fname'];

  // Periksa apakah nilai "fname" kosong atau tidak
  if (empty($name)) {
    // Jika kosong, tampilkan pesan "Name is empty"
    echo "Name is empty";
  } else {
    // Jika tidak kosong, tampilkan nilai "fname" yang dikirimkan
    echo $name;
  }
}
?>
```

Maleesss bikin kodenya manual, apalagi warnain satu persatu syntax-nya, ribet coy mending skrinsut aja

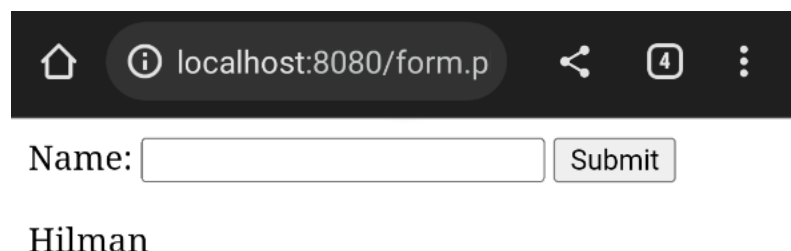
Penjelasan kode di atas:

- 1) `<form method="post" action="form.php">` : elemen `<form>` dalam HTML yang memiliki atribut `method="post"` dan `action="form.php"`. Atribut `method` menentukan metode HTTP yang digunakan untuk mengirimkan data formulir, dalam hal ini metode POST. Atribut `action` menentukan URL atau lokasi file yang akan menerima data formulir dan karena di sini `action`-nya merujuk ada file `form.php` itu sendiri, artinya prosesnya akan dilakukan di halaman yang sama bukan ke halaman yang lain
- 2) `if ($_SERVER["REQUEST_METHOD"] == "POST") {}` : kondisi dalam PHP yang memeriksa apakah permintaan yang diterima adalah metode POST. `$_SERVER["REQUEST_METHOD"]` mengambil nilai metode permintaan yang dikirimkan ke server. Jika metodenya adalah POST, maka kode di dalam blok `if` akan dieksekusi.
- 3) `$name = $_POST['fname']` : variabel `$name` yang akan menyimpan nilai yang dikirimkan melalui input dengan nama `"fname"` dalam formulir. `$_POST` adalah array superglobal di PHP yang berisi data yang dikirimkan melalui metode POST.

- 4) `if (empty($name)) {}` : kondisi dalam PHP yang memeriksa apakah nilai `$name` kosong atau tidak. `empty($name)` digunakan untuk memeriksa apakah variabel kosong atau tidak memiliki nilai.
- 5) `echo "Name is empty"` : perintah untuk mencetak teks "Name is empty" jika nilai `$name` kosong. Pesan ini akan ditampilkan jika input dengan name "`fname`" tidak diisi.
- 6) `echo $name` : perintah untuk mencetak nilai `$name` jika nilai tidak kosong. Nilai ini akan ditampilkan sebagai hasil jika input dengan name "`fname`" diisi dan dikirimkan melalui formulir.

Berikutnya Aktifkan web servernya lewat Termux jika belum, dengan menggunakan command `php -S localhost:8080` di dalam directory atau folder belajar-php.

Jika berhasil, akan seperti ini hasilnya



localhost:8080/form.p

Name:

Submit

Hilman

B. Validasi data masukan

C. Menyimpan dan mengolah data inputan

4. Penanganan File

5. Koneksi ke Database

6. Pengelolaan Session dan Cookies

7. Pengolahan Gambar

8. Framework PHP