**Dasar Dasar PHP**

**1. Apa Itu PHP?**

PHP (Hypertext Preprocessor) adalah bahasa pemrograman sisi server yang digunakan untuk mengembangkan aplikasi web dinamis. PHP dapat berinteraksi dengan database, menangani formulir, mengelola sesi, dan banyak lagi.

**2. Sintaks Dasar PHP**

Kode PHP ditulis di dalam tag berikut:

<?php

// Kode PHP di sini

?>

Contoh:

<?php

echo "Hello, World!";

?>

**Output:**

Hello, World!

**3. Variabel dan Tipe Data**

PHP menggunakan simbol $ untuk mendefinisikan variabel.

**Contoh variabel:**

<?php

$nama = "Rei";

$umur = 19;

$tinggi = 1.75;

$status = true;

?>

**Tipe data utama di PHP:**

* String → "Hello"
* Integer → 19
* Float → 1.75
* Boolean → true / false
* Array → ["apel", "mangga", "pisang"]
* Object → new Class()
* NULL → null

**4. Operator dalam PHP**

**Operator Aritmatika:**

$a = 10;

$b = 5;

$hasil = $a + $b; // Penjumlahan (15)

**Operator Perbandingan:**

$a == $b // Sama dengan

$a != $b // Tidak sama dengan

$a > $b // Lebih besar

$a < $b // Lebih kecil

**Operator Logika:**

($a > 5 && $b < 10) // AND

($a > 5 || $b < 10) // OR

!($a == $b) // NOT

**5. Struktur Kontrol**

**Percabangan (if-else):**

<?php

$nilai = 85;

if ($nilai >= 80) {

echo "Lulus dengan nilai A";

} elseif ($nilai >= 70) {

echo "Lulus dengan nilai B";

} else {

echo "Tidak lulus";

}

?>

**Switch-case:**

<?php

$hari = "Senin";

switch ($hari) {

case "Senin":

echo "Hari ini Senin";

break;

case "Selasa":

echo "Hari ini Selasa";

break;

default:

echo "Hari tidak ditemukan";

}

?>

**Perulangan (Looping):**

<?php

for ($i = 1; $i <= 5; $i++) {

echo "Angka: $i <br>";

}

?>

<?php

$i = 1;

while ($i <= 5) {

echo "Angka: $i <br>";

$i++;

}

?>

**6. Array dalam PHP**

PHP mendukung array untuk menyimpan banyak data dalam satu variabel.

**Array numerik:**

$buah = ["Apel", "Mangga", "Pisang"];

echo $buah[0]; // Output: Apel

**Array asosiatif:**

$mahasiswa = ["nama" => "Rei", "umur" => 19, "jurusan" => "Sistem Informasi"];

echo $mahasiswa["nama"]; // Output: Rei

**Array multidimensi:**

$kelas = [

["Reifan", 19, "Sistem Informasi"],

["Andi", 20, "Teknik Informatika"]

];

echo $kelas[0][0]; // Output: Reifan

**7. Fungsi dalam PHP**

**Membuat fungsi:**

<?php

function sapa($nama) {

return "Halo, $nama!";

}

echo sapa("Rei"); // Output: Halo, Rei!

?>

**8. Form Handling (GET & POST)**

PHP dapat menangani data formulir yang dikirim oleh pengguna.

**Form HTML:**

<form method="POST" action="proses.php">

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

<input type="submit">

</form>

**proses.php:**

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$nama = $\_POST["nama"];

echo "Nama Anda: $nama";

}

?>

**9. Koneksi ke Database MySQL**

PHP dapat digunakan untuk berinteraksi dengan database menggunakan MySQL.

**Koneksi ke MySQL:**

<?php

$host = "localhost";

$user = "root";

$pass = "";

$db = "nama\_database";

$conn = new mysqli($host, $user, $pass, $db);

if ($conn->connect\_error) {

die("Koneksi gagal: " . $conn->connect\_error);

}

echo "Koneksi berhasil!";

?>

**Menampilkan data dari database:**

<?php

$sql = "SELECT \* FROM users";

$result = $conn->query($sql);

while ($row = $result->fetch\_assoc()) {

echo $row["nama"] . "<br>";

}

?>

**10. Session dan Cookies**

**Session:** Digunakan untuk menyimpan data sementara selama pengguna masih aktif di website.

<?php

session\_start();

$\_SESSION["username"] = "Rei";

echo "Session disimpan!";

?>

**Cookies:** Digunakan untuk menyimpan data dalam jangka waktu tertentu di browser pengguna.

<?php

setcookie("user", "Rei", time() + 3600);

echo "Cookie disimpan!";

?>

**Kesimpulan**

PHP adalah bahasa pemrograman server-side yang sangat berguna untuk membangun website dinamis. Dengan PHP, kita dapat memproses data pengguna, berinteraksi dengan database, mengelola sesi, dan banyak lagi.

**Syntax Pada PHP**

## ****1. Dasar PHP****

### ****Tag Pembuka dan Penutup PHP****

<?php // Tag pembuka PHP

?> // Tag penutup PHP

Fungsi: Menandakan bahwa kode di dalamnya adalah kode PHP.

## ****2. Komentar****

// Komentar satu baris

# Komentar satu baris

/\*

Komentar lebih dari satu baris

\*/

Fungsi: Memberikan catatan dalam kode yang tidak akan dieksekusi.

## ****3. Variabel & Tipe Data****

$nama = "Rei"; // String

$umur = 20; // Integer

$tinggi = 1.75; // Float

$status = true; // Boolean

Fungsi: Menyimpan data dengan berbagai tipe.

## ****4. Operator****

**Operator Aritmatika**

$a + $b; // Penjumlahan

$a - $b; // Pengurangan

$a \* $b; // Perkalian

$a / $b; // Pembagian

$a % $b; // Modulus (sisa bagi)

**Operator Perbandingan**

$a == $b; // Sama dengan

$a != $b; // Tidak sama dengan

$a > $b; // Lebih besar

$a < $b; // Lebih kecil

$a >= $b; // Lebih besar sama dengan

$a <= $b; // Lebih kecil sama dengan

**Operator Logika**

($a > 5 && $b < 10); // AND

($a > 5 || $b < 10); // OR

!($a == $b); // NOT

## ****5. Struktur Kontrol****

**If-Else**

if ($nilai > 80) {

echo "A";

} elseif ($nilai > 60) {

echo "B";

} else {

echo "C";

}

Fungsi: Mengeksekusi kode berdasarkan kondisi.

**Switch-Case**

switch ($warna) {

case "merah":

echo "Warna merah";

break;

case "biru":

echo "Warna biru";

break;

default:

echo "Warna tidak diketahui";

}

Fungsi: Alternatif if-else dengan banyak kondisi.

**Looping (Perulangan)**

for ($i = 1; $i <= 5; $i++) {

echo $i;

}

Fungsi: Mengulang kode sebanyak yang ditentukan.

while ($i <= 5) {

echo $i;

$i++;

}

Fungsi: Mengulang kode selama kondisi benar.

do {

echo $i;

$i++;

} while ($i <= 5);

Fungsi: Menjalankan kode setidaknya sekali sebelum mengecek kondisi.

foreach ($array as $value) {

echo $value;

}

Fungsi: Iterasi pada array.

## ****6. Array****

**Array Numerik**

$buah = ["Apel", "Mangga", "Pisang"];

echo $buah[0]; // Output: Apel

Fungsi: Menyimpan banyak nilai dalam satu variabel.

**Array Asosiatif**

$person = ["nama" => "Rei", "umur" => 20];

echo $person["nama"];

Fungsi: Menggunakan kunci sebagai indeks array.

**Array Multidimensi**

$kelas = [

["Rei", 19],

["Andi", 20]

];

echo $kelas[0][0]; // Output: Rei

Fungsi: Array dalam array.

## ****7. Fungsi (Function)****

function sapa($nama) {

return "Halo, $nama!";

}

echo sapa("Rei");

**Fungsi**: Membuat blok kode yang bisa digunakan kembali.

**Fungsi dengan Parameter Default**

function salam($nama = "Guest") {

return "Halo, $nama!";

}

echo salam(); // Output: Halo, Guest!

**Fungsi**: Jika tidak ada parameter yang diberikan, maka akan menggunakan nilai default.

## ****8. Superglobal Variables****

PHP memiliki beberapa variabel superglobal yang bisa digunakan di mana saja.

| **Superglobal** | **Fungsi** |
| --- | --- |
| $\_GET | Mengambil data dari URL (Query String). |
| $\_POST | Mengambil data dari form yang dikirim dengan metode POST. |
| $\_REQUEST | Kombinasi dari $\_GET dan $\_POST. |
| $\_SESSION | Menyimpan data pengguna selama sesi aktif. |
| $\_COOKIE | Menyimpan data di browser pengguna. |
| $\_SERVER | Menyimpan informasi server dan request pengguna. |
| $\_FILES | Mengelola upload file. |

**Contoh $\_GET**

echo "Nama: " . $\_GET['nama'];

URL: index.php?nama=Rei → Output: Nama: Rei

**Contoh $\_POST**

echo "Nama: " . $\_POST['nama'];

Digunakan dalam form dengan metode POST.

## ****9. Session dan Cookie****

**Session** (Data sementara yang disimpan selama sesi browser)

session\_start();

$\_SESSION["user"] = "Rei";

echo $\_SESSION["user"];

**Cookie** (Data disimpan di browser pengguna)

setcookie("username", "Rei", time() + 3600, "/");

echo $\_COOKIE["username"];

## ****10. File Handling (Baca & Tulis File)****

**Membuka dan Membaca File**

$file = fopen("data.txt", "r");

echo fread($file, filesize("data.txt"));

fclose($file);

**Menulis ke File**

$file = fopen("data.txt", "w");

fwrite($file, "Halo, ini file baru!");

fclose($file);

## ****11. Koneksi ke Database (MySQLi & PDO)****

**Koneksi MySQLi (Procedural)**

$koneksi = mysqli\_connect("localhost", "root", "", "db\_saya");

if (!$koneksi) {

die("Koneksi gagal: " . mysqli\_connect\_error());

}

**Mengambil Data dari Database**

$result = mysqli\_query($koneksi, "SELECT \* FROM users");

while ($row = mysqli\_fetch\_assoc($result)) {

echo $row["nama"];

}

**PDO (Object-Oriented)**

try {

$db = new PDO("mysql:host=localhost;dbname=db\_saya", "root", "");

} catch (PDOException $e) {

echo "Koneksi gagal: " . $e->getMessage();

}

## ****12. Exception Handling****

try {

$angka = 0;

if ($angka == 0) {

throw new Exception("Angka tidak boleh nol!");

}

} catch (Exception $e) {

echo "Error: " . $e->getMessage();

}

**Fungsi**: Menangani error tanpa menghentikan eksekusi program.

## ****13. Object-Oriented Programming (OOP) dalam PHP****

**Membuat Class dan Object**

class Mobil {

public $merk;

function setMerk($merk) {

$this->merk = $merk;

}

function getMerk() {

return $this->merk;

}

}

$mobil1 = new Mobil();

$mobil1->setMerk("Toyota");

echo $mobil1->getMerk();

**Fungsi**: Menerapkan konsep OOP dalam PHP.