

## MODUL IV

### PHP (Hypertext Preprocessor)

#### A. TUJUAN

- Memahami struktur dasar dokumen PHP
- Mampu membuat dokumen PHP yang baik dan benar
- Mampu memanfaatkan elemen – elemen dasar untuk mengolah dan menampilkan informasi

#### B. PETUNJUK

- Awali setiap aktivitas dengan do'a, semoga berkah dan medapat kemudahan
- Pahami tujuan, dasar teori, dan latihan – latihan praktikum dengan baik dan benar
- Kerjakan tugas – tugas praktikum dengan baik, sabar, dan jujur
- Tanyakan kepada asisten/ dosen apabila ada hal – hal yang kurang jelas

#### C. DASAR TEORI

##### 1. PHP

PHP( *Hypertext Preprocessor*) merupakan salah satu bahasa *web scripting* yang sangat *powerful*. Ditemukan pertama kali oleh Rasmus Lerdorf tahun 1994. Bahasa ini dimaksudkan untuk menghasilkan halaman – halaman web yang dinamis.

Karakteristik penulisan :

- Ditulis diantara tag
  - `<? dan ?>`
  - `<?php dan ?>`
  - `<script language="php"> dan </script>`
  - `<% dan %>`
- Setiap statement diakhiri dengan titik – koma (;)
- CASE SENSITIVE** untuk nama identifier yang dibuat oleh user (variable, Konstanta, fungsi dll), namun **TIDAK CASE SENSITIVE** untuk *identifier built – in* dari PHP, misal :
  - `$nama ≠ $NAMA`
  - `hitungLuas() ≠ HitungLuas()`
  - `echo = ECHO`
  - `while = WHILE`

##### 2. Tipe Data

PHP mendukung delapan tipe primitif, yang terdiri dari empat tipe skalar (boolean, integer, float/double, dan string). Dua tipe gabungan (array dan object) dan sisanya adalah tipe khusus (NULL dan *resource*). Meskipun keberadaan tipe data dalam deklarasi variabel tidak penting, namun dalam tahap selanjutnya sangatlah penting.

### 3. Komentar

PHP mendukung tiga jenis sintaks untuk menuliskan baris komentar, meliputi style C, C++ dan Perl/Shell. Penggunaan sebagai berikut:

```
/* awal
Komentar style C
akhir */
//Komentar style C++
# Komentar style Perl/Shell
```

### 4. Tag PHP

Parser PHP bekerja dengan menggunakan instruksi – instruksi yang diletakkan di antara tag pembuka dan penutup. Ada beberapa tag yang selalu tersedia diantaranya , ada dua dari empat jenis tag yang selalu tersedia tanpa memerlukan konfigurasi adalah style XML dan SCRIPT.

```
<?php
    echo 'style XML';
?>

<script language="php">
    echo 'style SCRIPT';
</script>
```

Dua style tag lainnya memerlukan aktivasi dukungan di file konfigurasi php.

```
<?
    // Aktifkan dukungan style SGML/Short di php.ini
    // short_open_tag = On

    echo 'style SGML';
?>

<%
    // Aktifkan dukungan style ASP di php.ini
    // asp_tags = On

    echo 'style ASP';
%>
```

## D. LATIHAN

### 1. Program PHP

Dalam pembuatan aplikasi web, instruksi – instruksi PHP akan dikombinasikan dengan elemen – elemen HTML. Meskipun PHP dapat digunakan untuk membungkus keseluruhan dokumen HTML, namun sebaiknya hal ini tidak kita lakukan. Hal ini dapat menyulitkan pembacaan kode, langkah ini juga kurang efisien.

Pendekatan yang disarankan dalam kombinasi ini yaitu memisahkan kode program PHP dari dokumen HTML. Sebagai contoh, ketikkan kode berikut dan simpan dengan nama **php\_html.php** (ekstensi file yang digunakan adalah **php**)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Demo</title>
</head>

<body>

<?php
echo 'Kode PHP di sini';
// ...
?>

<p> Dokumen HTML </p>

<?php
echo 'Kode PHP di sini';
// ...
?>

</body>
</html>
```

Berkenaan dengan kombinasi ini, tag PHP menyediakan shortcut untuk meringkas penulisan kode. Shortcut ini digunakan ketika ingin menuliskan kode – kode kecil di dalam HTML.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Demo</title>
</head>

<body>

<p> Kode <?php echo "PHP";?> di HTML </p>

</body>
</html>
```

## 2. Variabel

Digunakan untuk menyimpan sebuah value, data atau informasi. Beberapa aturan penulisannya :

- Nama variabel diawali dengan tanda \$
- Panjang tidak terbatas
- Setelah tanda \$ diawali oleh huruf atau *under – score* (\_). Karakter berikutnya terdiri dari huruf, angka dan karakter tertentu yang diperbolehkan (karakter ASCII dari 127 -255)
- Bersifat case sensitive
- Tidak perlu dideklarasikan
- Tidak boleh mengandung spasi
- Contoh:

Benar	Salah
- \$_name	- \$3name
- \$first_name	- \$name?
- \$name	- \$first+name
- \$name_3	- \$first.name
	- \$first name

Simpan file berikut dengan nama **demo\_var.php**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Demo Variabel</title>
</head>

<body>

<?php

// Deklarasi dan inisialisasi
$bil = 3;

echo $bil;

?>

</body>
</html>
```

Di tahap pengembangan, kita bisa memanfaatkan fungsi `var_dump()` atau `print_r()` untuk memudahkan pemeriksaan variabel. Simpan dengan nama file `demo_var2.php`

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Demo Variabel</title>
</head>

<body>

<?php

// Deklarasi dan inisialisasi
$bil = 3;

// Dumping informasi mengenai variabel
var_dump($bil);

print_r($bil);

?>

</body>
</html>
```

Amati dan pahami hasil keluaran dari fungsi `var_dump()`. Apa bedanya dengan konstruksi hasasa `echo()`? Jelaskan!

### 3. Tipe Data dan Casting

PHP menyediakan fungsi – fungsi berawalan `is_` yang dapat dimanfaatkan untuk menguji tipe data dari suatu variabel.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Cek Tipe</title>
</head>
<body>

<?php

$bil = 3;
var_dump(is_int($bil));
// Output: bool(true)

$var = "";
var_dump(is_string($var));
// Output: bool(true)

?>

</body>
</html>
```

Kecuali fungsi `is_bool()`, penamaan fungsi – fungsi lainnya persis seperti tipe data terkait, misalnya `is_integer()`, `is_float()`, `is_object()` dan `is_null()`.

### Casting

Untuk menguji aplikasi web, kita mempublikasikannya ke web server, baik secara lokal maupun internet. Lingkungan lokal tentu merupakan pilihan yang efisien, khususnya ketika aplikasi masih dalam tahap pengembangan.

Operator	Deskripsi
(int), (integer)	Casting ke integer
(double), (float), (real)	Casting ke floating point
(string)	Casting ke string
(array)	Casting ke array
(object)	Casting ke object
(bool), (boolean)	Casting ke boolean
(unset)	Casting ke null, seperti pemanggilan unset

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Casting Tipe</title>
</head>
<body>

<?php

$str = '123abc';

// Casting nilai variabel $str ke integer
$bil = (int) $str; // $bil = 123

echo gettype($str);
// Output: string

echo gettype($bil);
// Output: integer

?>

</body>
</html>
```

#### 4. Pernyataan Seleksi

Pernyataan seleksi untuk pengambilan keputusan merupakan bagian dari bahasa pemrograman yang sangat penting. Pernyataan – pernyataan pengambilan keputusan di PHP diklasifikan ke dalam empat bagian yaitu:

##### a. if

Pernyataan if terdiri dari suatu ekspresi dan sebuah statemen atau blok statemen yang dieksekusi apabila ekspresi bernilai true.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
```

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Seleksi</title>
</head>
<body>

<?php

$a = 10;
$b = 5;

if ($a > $b) {
    echo 'a lebih besar dari b';
}

?>

</body>
</html>
```

### b. if-else

Pernyataan ini sebenarnya merupakan ekspansi dari pernyataan if. Disini blok pertama akan dieksekusi manakala ekspresi bernilai true . Jika ekspresi bernilai false , blok kedua yang dieksekusi.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Seleksi if-else</title>
</head>
<body>

<?php

$a = 10;
$b = 5;

if ($a > $b) {
    echo 'a lebih besar dari b';
} else {
    echo 'a TIDAK lebih besar dari b';
}

?>

</body>
</html>
```

### c. if-elseif

Pernyataan ini sebenarnya merupakan ekspansi dari pernyataan if -else. Dimana di tambahkan lagi blok if – elseif . Bentuk pernyataan if-elseif memungkinkan kita untuk menciptakan seleksi yang lebih kompleks.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Seleksi if-elseif</title>
</head>
<body>

<?php

$a = 10;
$b = 5;

if ($a > $b) {
    echo 'a lebih besar dari b';
} elseif ($a == $b) {
    echo 'a sama dengan b';
} else {
    echo 'a kurang dari b';
}

?>

</body>
</html>
```

### d. Pernyataan switch

Sebuah pernyataan control flow yang dimulai dengan suatu ekspresi dan mentransfer kontrol ke satu kasus berdasarkan nilai ekspresi.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Seleksi switch</title>
</head>
<body>

<?php

$i = 0;

if ($i == 0) {
    echo "i equals 0";
} elseif ($i == 1) {
    echo "i equals 1";
} elseif ($i == 2) {
    echo "i equals 2";
}

// Ekuivalen, dengan pendekatan switch
switch ($i) {
case 0:
    echo "i equals 0";
    break;
case 1:
    echo "i equals 1";
    break;
case 2:
    echo "i equals 2";
    break;
}

?>

</body>
</html>
```

## 5. Pengulangan

Bagian ini akan menguraikan jenis – jenis dan implementasi struktur kontrol perulangan

- Pengulangan while

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Loop while</title>
</head>
<body>

<?php
$i = 0;

while ($i < 10) {
    echo $i;

    // Inkremen counter
    $i++;
}

?>

</body>
</html>
```

- Pengulangan do – while

Bentuk pengulangan ini mirip dengan while, kecuali bahwa ekspresi pengontrolan pengulangan dilakukan di akhir blok. Ini juga berarti bahwa blok pengulangan akan dieksekusi setidaknya satu kali, meskipun bernilai false.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Loop do-while</title>
</head>
<body>

<?php

$i = 0;

do {
  echo $i;

  // Inkremen counter
  $i++;
} while ($i < 10);

?>

</body>
</html>
```

- Pengulangan for  
Struktur :
  1. Ekspresi inisialisasi,
  2. Kondisi perulangan, dan
  3. Ekspresi inkremen

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Loop for</title>
</head>
<body>

<?php

for ($i = 0; $i < 10; $i++) {
  echo $i;
}

?>

</body>
</html>
```

- Pengulangan foreach  
PHP juga menyediakan konstruksi foreach yang dapat digunakan untuk melakukan iterasi di array atau koleksi.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Loop foreach</title>
</head>
<body>

<?php

$arr = array(1, 2, 3, 4);

foreach ($arr as $value) {
  echo $value;
}

?>

</body>
</html>
```



## 6. Fungsi dan Prosedur

- Definisi Fungsi/Prosedur

```
<?php
// Contoh prosedur
function do_print() {
    // Mencetak informasi timestamp
    echo time();
}

// Memanggil prosedur
do_print();

echo '<br />';

// Contoh fungsi penjumlahan
function jumlah($a, $b) {
    return ($a + $b);
}

echo jumlah(2, 3);
// Output: 5
?>
```

- Argumen Fungsi/Prosedur

Suatu fungsi dapat memiliki nol atau lebih argumen. Adapun jika diperlukan, juga bisa dideklarasikan argumen yang sifatnya opsional. Deklarasi ini sekaligus menginisialisasi nilai default pada argumen. Selain itu, argumen opsional harus diletakkan di urutan paling belakang.

```
<?php
/**
 * Mencetak string
 * $teks nilai string
 * $bold adalah argumen opsional
 */
function print_teks($teks, $bold = true) {
    echo $bold ? '<b>' . $teks . '</b>' : $teks;
}

print_teks('Indonesiaku');
// Mencetak dengan huruf tebal

print_teks('Indonesiaku', false);
// Mencetak dengan huruf reguler
?>
```

## E. TUGAS PRAKTIKUM

- Buat fungsi greeting yang menerima argumen integer jam (format 24 jam) dan mengembalikan nilai string (Selamat Pagi, Selamat Siang atau Selamat malam). Jika tidak memenuhi maka akan mengembalikan string kosong.
- Buat fungsi sederhana untuk generate matriks sel pada tabel. Misalkan diberikan argumen 3 (baris) dan 4 (kolom). Maka program akan menampilkan tabel berisi 3 baris dan 4 kolom sel.

## F. STUDI KASUS

- Uraikan secara ringkas menangani passing argumen di PHP dan berikan contoh passing by value dan by reference.
- Buat program sederhana untuk generate sel tabel secara fleksibel. Tekniknya, buat sebuah fungsi yang menerima argumen berapa jumlah sel dan jumlah kolom. Jadi, pembentukan sel tabel didasarkan pada nilai jumlah sel dan jumlah kolom yang diberikan. Sebagai contoh, hasil program dengan argumen 12 (sel) dan 5 (kolom) diperlihatkan seperti Gambar 1.

1	2	3	4	5
6	7	8	9	10
11	12			