

## MODUL 7

### PHP (Hypertext Preprocessor)

#### A. Pengenalan PHP



PHP atau PHP Hypertext Preprocessor adalah sebuah bahasa script berbasis server (server-side) yang mampu mem-parsing kode php dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client (browser). Dengan menambahkan skrip PHP, anda bisa menjadikan halaman HTML menjadi lebih powerful, dinamis dan bisa dipakai sebagai aplikasi lengkap, misalnya web portal, e-learning, e-library, dll.

PHP pertama kali dikembangkan oleh seorang programmer bernama Rasmus Lerdorf pada tahun 1995. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakan PHP/FI sehingga banyak pemrograman yang tertarik untuk ikut mengembangkan PHP. Dan selanjutnya pada tahun 1997 perusahaan bernama Zend, mengembangkan interpreter PHP tersebut menjadi lebih baik. Kode PHP diproses melalui pemrosesan dari sisi server, makanya PHP disebut skrip server-side. Sehingga kode PHP tidak bisa diberikan langsung ketika ada permintaan dari client(browser).

Kode PHP dimasukkan ke dalam kode HTML dengan cara menyelipkannya di dalam kode HTML. Untuk membedakan kode PHP dengan kode HTML, di depan kode PHP tersebut diberi tag pembuka dan diakhir kode PHP diberi tag penutup.

Dengan adanya kode PHP, sebuah halaman web bisa melakukan banyak hal yang dinamis, seperti mengakses database, membuat gambar,

membaca dan menulis file, dan sebagainya. Hasil pengolahan kode PHP akan dikembalikan lagi dalam bentuk kode HTML untuk ditampilkan di browser.

## B. PHP Dasar

### 1. Penulisan Kode PHP

#### Sintaks PHP Standar

Pengelolaan kode PHP dapat dilakukan dengan beberapa cara untuk masuk ke dalam mode PHP seperti berikut:

Jenis Tag	Tag Pembuka	Tag Penutup
Tag Standar	<?php	?>
Tag Pendek	<?	?>
Tag ASP	<%	%>
Tag Script	<script language = "php">	</script>

Buatlah program pertama php dengan text editor, tuliskan program berikut:

```
<?php
echo "Ini Adalah Script PHP Pertama Saya<br>";
echo "Saya sedang belajar PHP di Polibang";
?>
```

Simpan dengan nama **latihan1.php**

Untuk melihat hasilnya buka browser masuk ke dalam localhost atau htdocs dan folder penyimpanan.

#### Latihan!!!

Buatlah program php dengan menggunakan seluruh cara pengelolaan kode PHP.

#### Menyematkan PHP dalam HTML

PHP dapat disematkan dalam halaman web HTML dengan menulis pernyataan PHP dalam dokumen HTML, seperti yang ditunjukkan pada contoh berikut:

```
<!DOCTYPE HTML>
<html>

<head>
  <title>Applikasi PHP</title>
</head>
```

```
<body>
    <?php
        echo "Selamat Datang di Web PHP Polibang";
    ?>
</body>

</html>
```

### Sensitivitas Huruf Besar dalam PHP

Nama variabel dalam PHP bersifat case-sensitive. Akibatnya variabel \$warna, \$WARNA dan \$WARna diperlakukan sebagai tiga variabel yang berbeda seperti berikut:

```
<!DOCTYPE html>
<html>

<body>

    <?php
        $warna = "red";
        echo "Rumah saya " . $warna . "<br>";
        echo "Mobil saya " . $WARNA . "<br>";
        echo "Sepeda saya " . $WARna . "<br>";
    ?>

</body>

</html>
```

### Latihan!!!

Buatlah program php yang menunjukkan bahwa program PHP sensitive terhadap besar kecil huruf.

#### 2. Fungsi PHP dalam Web

Sebuah program perulangan ditulis dengan menggunakan kode HTML seperti berikut:

```
<html>

<head>
    <title>Contoh Penggunaan List dalam HTML</title>
</head>

<body>
    <h2>Daftar Nama Barang</h2>
    <ol>
```

```
<li>Barang ke-1</li>
<li>Barang ke-2</li>
<li>Barang ke-3</li>
<li>Barang ke-4</li>
<li>Barang ke-5</li>
<li>Barang ke-6</li>
<li>Barang ke-7</li>
<li>Barang ke-8</li>
<li>Barang ke-9</li>
<li>Barang ke-10</li>
<li>Dan seterusnya sampei ke-50</li>
</ol>
</body>
</html>
```

Jika menggunakan PHP maka script akan menjadi lebih sederhana dan dinamis, cukup membuat perulangan for sebanyak 50 kali dengan instruksi seperti berikut ini:

```
<html>
<head>
  <title>Contoh Penggunaan List dalam HTML</title>
</head>
<body>
  <h2>Daftar Nama Barang</h2>
  <ol>
    <?php
      for ($i = 1; $i <= 50; $i++) {
        echo "<li>Barang ke-$i</li>";
      }
    <?>
  </ol>
</body>
</html>
```

### Latihan!!!

Buatlah program php yang menunjukkan bahwa program PHP akan membuat script menjadi lebih sederhana dan dinamis.

### 3. Variabel PHP

Variabel digunakan untuk menyimpan data, seperti string teks, angka, dll. Nilai variabel dapat berubah nilainya pada saat kode program sedang berjalan. Berikut beberapa hal penting yang perlu diketahui tentang variabel:

- Dalam PHP, variabel tidak perlu dideklarasikan sebelum menambahkan nilai ke dalamnya. PHP secara otomatis mengubah variabel ke tipe data yang benar, tergantung pada nilainya.
- Setelah mendeklarasikan variabel, itu dapat digunakan kembali di seluruh kode.
- Operator penugasan (=) digunakan untuk memberikan nilai ke variabel.

Dalam PHP terdapat aturan dalam penamaan variabel seperti:

- Semua variabel dalam PHP dimulai dengan tanda \$, diikuti dengan nama variabel.
- Nama variabel harus dimulai dengan huruf atau karakter garis bawah \_.
- Nama variabel tidak boleh diawali dengan angka.
- Nama variabel dalam PHP hanya boleh berisi karakter alfanumerik dan garis bawah (A-z, 0-9, dan \_).
- Nama variabel tidak boleh berisi spasi.

Dalam variabel PHP dapat dideklarasikan sebagai: \$var\_name = value. Dalam contoh berikut dibuat dua variabel di mana yang pertama telah ditetapkan dengan nilai string dan yang kedua telah ditetapkan dengan angka. Kemudian menampilkan nilai variabel di browser menggunakan pernyataan echo. Pernyataan echo PHP sering digunakan untuk menampilkan data ke browser.

```
<?php
// Deklarasi variabel
$txt = "Hello POLIBANG!";
$number = 10;

// Menampilkan nilai variabel
echo $txt; // Output: Hello World!
echo $number; // Output: 10
?>
```

### Latihan!!!

Buatlah program php dengan menggunakan minimal 5 variabel dan kombinasikan variabel tersebut saat ditampilkan.

## Lingkup Variabel PHP

Dalam PHP, variabel dapat dideklarasikan di mana saja dalam skrip. Ruang lingkup variabel adalah bagian dari skrip di mana variabel dapat direferensikan/digunakan.

PHP memiliki tiga cakupan variabel yang berbeda: local, global dan statis. Variabel yang dideklarasikan di luar fungsi memiliki LINGKUP GLOBAL dan hanya dapat diakses di luar fungsi seperti berikut:

```
<?php
    $x = 5; // global scope

    function myTest()
    {
        // penggunaan x didalam function akan menghasilkan error
        echo "<p>Variable x didalam function is: $x</p>";
    }
    myTest();

    echo "<p>Variable x diluar function is: $x</p>";
?>
```

Variabel yang dideklarasikan dalam suatu fungsi memiliki LINGKUP LOKAL dan hanya dapat diakses di dalam fungsi itu:

```
<?php
    function myTest()
    {
        $x = 5; // local scope
        echo "<p>Variable x inside function is: $x</p>";
    }
    myTest();

    // penggunaan x diluar function akan menghasilkan error
    echo "<p>Variable x outside function is: $x</p>";
?>
```

Kata kunci global digunakan untuk mengakses variabel global dari dalam suatu fungsi. Untuk melakukan ini, gunakan kata kunci global sebelum variabel (di dalam fungsi) seperti berikut:

```
<?php
    $x = 5;
    $y = 10;

    function myTest()
```

```
{
    global $x, $y;
    $y = $x + $y;
}

myTest();
echo $y; // outputs 15
?>
```

### Latihan!!!

Buatlah program php dengan menggunakan lingkup variabel local dan global.

#### 4. PHP Echo dan Print Statements

Echo dan print kurang lebih memiliki fungsi yang sama. Keduanya digunakan untuk menampilkan data ke layar. Terdapat perbedaan kecil:

Echo tidak memiliki nilai kembalian sedangkan print memiliki nilai kembalian 1 sehingga dapat digunakan dalam ekspresi. Echo dapat mengambil beberapa parameter (walaupun penggunaan seperti itu jarang terjadi) sementara print dapat mengambil satu argumen. Echo sedikit lebih cepat daripada print.

#### PHP echo Statement

Pernyataan echo dapat menampilkan satu atau lebih string. Secara umum, pernyataan echo dapat menampilkan apa saja yang dapat ditampilkan ke browser, seperti string, angka, nilai variabel, hasil ekspresi, dll. Pernyataan echo dapat digunakan dengan atau tanpa tanda kurung: echo atau echo().

Berikut akan menunjukkan cara menampilkan kode HTML menggunakan pernyataan echo:

```
<?php
// Menampilkan dengan HTML code
echo "<h4>Ini Adalah heading.</h4>";
echo "<h4 style='color: red;'>Heading Menggunakan
style.</h4>";
echo "<h2>PHP POLIBANG Menyenangkan!</h2>";
echo "Hello world!<br>";
echo "I'm about to learn PHP!<br>";
echo "This ", "string ", "was ", "made ", "with multiple
parameters.";
?>
```

Berikut akan menunjukkan cara menampilkan variabel menggunakan pernyataan echo:

```
<?php
// Pembuatan variables
$txt = "Selamat Datang POLIBANG";
$num = 123456789;
$colors = array("Red", "Green", "Blue");

// Menampilkan variables
echo $txt;
echo "<br>";
echo $num;
echo "<br>";
echo $colors[2];
?>
```

### PHP print Statement

Penggunaan pernyataan print (alternatif untuk echo) untuk menampilkan output ke browser. Seperti echo, print juga merupakan konstruksi bahasa bukan fungsi nyata. Pernyataan print dapat digunakan dengan atau tanpa tanda kurung: print atau print().

Berikut akan menunjukkan cara menampilkan kode HTML menggunakan pernyataan print:

```
<?php
// Displaying HTML code
print "<h2>PHP di POLIBANG menyenangkan!</h2>";
print "<h4>Ini adalah contoh heading.</h4>";
print "<h4 style='color: blue;'>Ini Heading dengan
style.</h4>";
print "Hello world!<br>";
print "I'm about to learn PHP!";
?>
```

Berikut akan menunjukkan cara menampilkan variabel menggunakan pernyataan print:

```
<?php
// Pembuatan variables
$txt = "Selamat Datang POLIBANG";
$num = 123456789;
$colors = array("Red", "Green", "Blue");
$x = 5;
$y = 4;
```



```
// Menampilkan variables  
echo $txt;  
echo "<br>";  
echo $num;  
echo "<br>";  
echo $colors[2];  
print $x + $y;  
?>
```

### Latihan!!!

Buatlah program php dengan menggunakan echo statement untuk menampilkan data tentang Politeknik Balekambang.

Buatlah program php dengan menggunakan print statement untuk menampilkan data diri dengan warna setiap baris tulisan berbeda.

### 5. PHP Tipe Data

Nilai yang ditetapkan ke variabel PHP dari tipe data seperti string sederhana dan tipe numerik ke tipe data yang lebih kompleks seperti array dan objek.

PHP mendukung total delapan tipe data berikut:

- String

String adalah urutan karakter, di mana setiap karakter sama dengan satu byte. String dapat menampung huruf, angka, dan karakter khusus dan dapat berukuran hingga 2GB (maksimum 2147483647 byte). Cara paling sederhana untuk menentukan string adalah dengan mengapitnya dalam tanda kutip tunggal ('Halo dunia!'), namun Anda juga dapat menggunakan tanda kutip ganda ("Halo dunia!").

```
<?php  
$a = 'Hello world!';  
echo $a;  
echo "<br>";  
  
$b = "Hello world!";  
echo $b;  
echo "<br>";  
  
$c = 'Stay here, I\'ll be back.';  
echo $c;  
?>
```

- Integer

Integer adalah bilangan bulat, tanpa titik desimal (... , -2, -1, 0, 1, 2, ...). Bilangan bulat dapat ditentukan dalam desimal (basis 10), heksadesimal (basis 16 - diawali dengan 0x) atau oktal (basis 8 - diawali dengan 0) notasi, opsional didahului dengan tanda (- atau +).

```
<?php
$a = 123; // decimal number
var_dump($a);
echo "<br>";

$b = -123; // a negative number
var_dump($b);
echo "<br>";

$c = 0x1A; // hexadecimal number
var_dump($c);
echo "<br>";

$d = 0123; // octal number
var_dump($d);
?>
```

- Float (floating point numbers - also called double)

Floating point numbers (juga dikenal sebagai "Float", "ganda", atau "bilangan real") adalah angka desimal atau pecahan, seperti yang ditunjukkan pada contoh di bawah ini.

```
<?php
$a = 1.234;
var_dump($a);
echo "<br>";

$b = 10.2e3;
var_dump($b);
echo "<br>";

$c = 4E-10;
var_dump($c);
?>
```

- Boolean

Boolean seperti saklar yang hanya memiliki dua kemungkinan nilai yaitu 1 (benar) atau 0 (salah).

```
<?php
// Assign the value TRUE to a variable
$show_error = true;
var_dump($show_error);
?>
```

- Array

Array adalah variabel yang dapat menampung lebih dari satu nilai dalam satu waktu. Berguna untuk menggabungkan serangkaian item terkait, misalnya sekumpulan nama negara atau kota. Array secara formal didefinisikan sebagai kumpulan nilai data yang diindeks. Setiap indeks (juga dikenal sebagai kunci) dari sebuah array adalah unik dan mengacu pada nilai yang sesuai.

```
<?php
$colors = array("Red", "Green", "Blue");
var_dump($colors);
echo "<br>";

$color_codes = array(
    "Red" => "#ff0000",
    "Green" => "#00ff00",
    "Blue" => "#0000ff"
);
var_dump($color_codes);
?>
```

- Object

Objek adalah tipe data yang tidak hanya memungkinkan penyimpanan data tetapi juga informasi tentang cara memproses data tersebut. Objek adalah instance spesifik dari kelas yang berfungsi sebagai templat untuk objek. Objek dibuat berdasarkan template ini melalui kata kunci baru. Setiap objek memiliki properti dan metode yang sesuai dengan kelas induknya. Setiap instance objek sepenuhnya independen, dengan properti dan metodenya sendiri, dan dengan demikian dapat dimanipulasi secara independen dari objek lain dari kelas yang sama. Berikut adalah contoh sederhana dari definisi kelas yang diikuti dengan pembuatan objek.

```
<?php
// Class definition
class greeting{
    // properties
    public $str = "Hello World!";

    // methods
    function show_greeting(){
        return $this->str;
    }
}

// Create object from class
$message = new greeting;
var_dump($message);
?>
```

- NULL

Nilai NULL khusus digunakan untuk mewakili variabel kosong di PHP. Variabel bertipe NULL adalah variabel tanpa data. NULL adalah satu-satunya nilai yang mungkin dari tipe null.

```
<?php
$a = NULL;
var_dump($a);
echo "<br>";

$b = "Hello World!";
$b = NULL;
var_dump($b);
?>
```

- Resource

Sebuah sumber daya adalah variabel khusus, memegang referensi ke sumber daya eksternal. Variabel sumber daya biasanya menahan penanganan khusus untuk membuka file dan koneksi database.

```
<?php
// Open a file for reading
$handle = fopen("note.txt", "r");
var_dump($handle);
echo "<br>";

// Connect to MySQL database server with default setting
$link = mysqli_connect("localhost", "root", "");
```

```
var_dump($link);
```

### Latihan!!!

Buatlah program php dengan memanfaatkan 4 konsep tipe data dalam satu program.

#### 6. PHP Operators

Operator adalah simbol yang memberi tahu prosesor PHP untuk melakukan tindakan tertentu. Misalnya, simbol penjumlahan (+) adalah operator yang memerintahkan PHP untuk menambahkan dua variabel atau nilai, sedangkan simbol lebih besar dari (>) adalah operator yang memberi tahu PHP untuk membandingkan dua nilai. Daftar berikut menjelaskan berbagai operator yang digunakan dalam PHP.

- PHP Arithmetic Operators

Operator aritmatika digunakan untuk melakukan operasi aritmatika umum, seperti penambahan, pengurangan, perkalian, dll. Berikut contoh lengkap penerapan operator aritmatika PHP:

```
<?php
$x = 10;
$y = 4;
echo($x + $y); // Outputs: 14
echo($x - $y); // Outputs: 6
echo($x * $y); // Outputs: 40
echo($x / $y); // Outputs: 2.5
echo($x % $y); // Outputs: 2
?>
```

- PHP Assignment Operators

Operator penugasan digunakan untuk menetapkan nilai ke variabel. Berikut daftar lengkap operator penugasan PHP:

```
<?php
$x = 10;
echo $x; // Outputs: 10

$x = 20;
$x += 30;
echo $x; // Outputs: 50

$x = 50;
$x -= 20;
```

```
echo $x; // Outputs: 30

$x = 5;
$x *= 25;
echo $x; // Outputs: 125

$x = 50;
$x /= 10;
echo $x; // Outputs: 5

$x = 100;
$x %= 15;
echo $x; // Outputs: 10
?>
```

- PHP Comparison Operators

Operator perbandingan digunakan untuk membandingkan dua nilai dengan cara Boolean.

```
<?php
$x = 25;
$y = 35;
$z = "25";
var_dump($x == $z); // Outputs: boolean true
var_dump($x === $z); // Outputs: boolean false
var_dump($x != $y); // Outputs: boolean true
var_dump($x !== $z); // Outputs: boolean true
var_dump($x < $y); // Outputs: boolean true
var_dump($x > $y); // Outputs: boolean false
var_dump($x <= $y); // Outputs: boolean true
var_dump($x >= $y); // Outputs: boolean false
?>
```

- PHP Incrementing and Decrementing Operators

Operator kenaikan/penurunan digunakan untuk menambah/mengurangi nilai variabel.

```
<?php
$x = 10;
echo ++$x; // Outputs: 11
echo $x; // Outputs: 11

$x = 10;
echo $x++; // Outputs: 10
echo $x; // Outputs: 11
```

```
$x = 10;
echo --$x; // Outputs: 9
echo $x;   // Outputs: 9

$x = 10;
echo $x--; // Outputs: 10
echo $x;   // Outputs: 9
?>
```

- PHP Logical Operators

Operator logika biasanya digunakan untuk menggabungkan pernyataan kondisional.

```
<?php
$year = 2014;
// Leap years are divisible by 400 or by 4 but not 100
if(($year % 400 == 0) || (($year % 100 != 0) && ($year % 4 == 0))){
    echo "$year is a leap year.";
} else{
    echo "$year is not a leap year.";
}
?>
```

- PHP String Operators

Ada dua operator yang dirancang khusus untuk string.

```
<?php
$x = "Hello";
$y = " World!";
echo $x . $y; // Outputs: Hello World!

$x .= $y;
echo $x; // Outputs: Hello World!
?>
```

- PHP Array Operators

Operator array digunakan untuk membandingkan array:

```
<?php
$x = array("a" => "Red", "b" => "Green", "c" => "Blue");
$y = array("u" => "Yellow", "v" => "Orange", "w" => "Pink");
$z = $x + $y; // Union of $x and $y
var_dump($z);
var_dump($x == $y); // Outputs: boolean false
var_dump($x === $y); // Outputs: boolean false
```

```
var_dump($x != $y); // Outputs: boolean true
var_dump($x <> $y); // Outputs: boolean true
var_dump($x !== $y); // Outputs: boolean true
?>
```

## Latihan!!!

Buatlah program php dengan menggunakan Operator untuk menampilkan seperti berikut:

## Belajar Operator PHP

### Operator Aritmatika

```
5 + 10 = 15
5 - 10 = -5
5 * 10 = 50
5 / 10 = 0.5
5 % 10 = 5
5 ** 10 = 9765625
-a = -5
```

### Operator Penugasan

```
int(15)
int(-5)
int(-500)
int(-50)
```

### Operator Perbandingan

```
90 > 80 = bool(true)
3 >= 3 = bool(true)
3 < 6 = bool(true)
5 <= 3 = bool(false)
'a' < 'b' = bool(true)
'abc' < 'b' = bool(true)
```

## C. Tugas Praktikum

### 1. Praktikum 1

Buatlah program web untuk mengolah nama-nama Dosen dengan kriteria sebagai berikut:

- Nama-nama disimpan didalam variabel array.
- Program menampilkan jumlah kata dan jumlah huruf pada nama yang tuliskan.
- Program menampilkan kebalikan dari nama yang diinputkan.

### 2. Praktikum 2

Buatlah program web untuk menginputkan data menggunakan form dan penanganan input data dengan kriteria sebagai berikut:



## PHP Form Validation Example

\* required field.

Name:  \*

E-mail:  \*

Website:

Comment:

Gender: ☐ Female ☐ Male \*

Simpan data ke dalam database

Tampilkan data yang sudah disimpan di database

3.