



# Struktur Dasar PHP

# Sejarah PHP

- Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.
- Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.
- Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.
- Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.
- Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

# On this slide...



- Mengenal PHP
- Hello World PHP
- Variabel
- Tipe Data
- Konstanta
- Operator dalam PHP
- Komentar Program

# Sebelum itu...



- Harus sudah mengenal komputer.
  - File? Direktori? Membuat file? Dsb
- Harus dapat menggunakan editor (Notepad, Crimson Ed, Dreamweaver)
  - Menyimpan dalam \*.PHP ?
- XAMPP sudah terinstal dengan baik.
- Harus dapat menggunakan browser (IE, Mozilla, dll)

# Sebelum itu...



- Cukup mengerti dasar-dasar HTML (PW1)
- Penglihatan masih normal
  - Nama = nama = NAMA ?
  - , = ; = : ?
  - ‘ = “ ?
  - nama\_mhs = namamhs = nama mhs ?
- Dapat menggunakan Google dgn baik

# PHP itu ...



- Merupakan singkatan *recursive* dari **PHP : Hypertext Preprocessor**
- Pertama kali dibuat oleh [Rasmus Lerdorf](#) pada tahun 1994.
- Setiap satu statement (perintah) biasanya 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

# Penulisan Script PHP



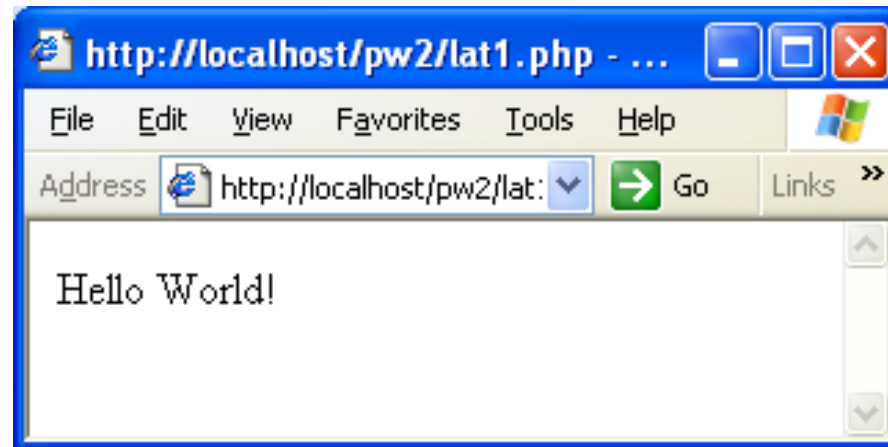
Harus diapit oleh ...

- `<? dan ?>` atau
- `<?php dan ?>` atau
- `<script language="php">` dan `</script>` atau
- `<% dan %>`

# Hello World!



```
1 <?php
2 echo "Hello world!";
3 ?>
```





# Variabel



- Digunakan untuk menyimpan sebuah value, data atau informasi
- Nama variabel diawali dengan tanda \$
- Panjang tidak terbatas
- Setelah tanda \$ diawali oleh huruf atau *under-score* (\_). Karakter berikutnya bisa 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 Program

```
<?php
// Pendeklarasian variabel
$nama_lengkap = "El Haqqi Pratama";
$_gaji = 5000000;

// Tampilkan variabel
echo "Nama Lengkap : $nama_lengkap <br>";
echo "Gaji : Rp. $_gaji <br>";

$_gaji = "Lima Juta Rupiah";
echo "Terbilang : $_gaji";
?>
```

# Tipe Data



- boolean
- integer
- float
- string
- array
- object
- resource
- NULL

Programmer pada dasarnya tidak perlu mendefinisikan tipe data

# Konstanta



- Konstanta merupakan variabel konstan yang nilainya tidak berubah-ubah.
- Menggunakan fungsi `define()`

```
<?php
$panjang = 10;
$lebar = 10;
define("luas_persegi",$panjang*$lebar);
echo "Luas persegi adalah = ".luas_persegi;
?>
```

```
<?php
define("panjang",10);
define("lebar",10);
$luas = panjang * lebar;
echo "Luas persegi adalah = $luas";
?>
```

# Operator



- Aritmatika

– +

– -

– \*

– /

– %

# Operator



- Aritmatika
  - Penugasan
- =

# Operator



- Aritmatika
- Penugasan
- Bitwise
  - &
  - |
  - ^
  - ~
  - <<
  - >>



# Operator



- Aritmatika
- Penugasan
- Bitwise
- Perbandingan

■ ==

■ ===

■ !=

■ <>

■ !==

■ <

■ >

■ <=

■ >=

# Operator



- Aritmatika
- Penugasan
- Bitwise
- Perbandingan
- Logika
  - and, &&, or, ||, xor, !

# Operator



- Aritmatika
- Penugasan
- Bitwise
- Perbandingan
- Logika
- String
  - Penggabungan → .

# Contoh program



```
<?
// Pendeklarasian
$angka1 = 20;
$angka2 = 5;
$jumlah = $angka1 + $angka2;
$kurang = $angka1 - $angka2;
$kali = $angka1 * $angka2;
$bagi = $angka1 / $angka2;
$sisas = $angka1 % $angka2;

// Tampilkan
echo "Nilai variabel \ $angka1 = $angka1 dan \ $angka2 = $angka2 <br>";
echo "Dijumlah : $jumlah <br>";
echo "Dikurang : $kurang <br>";
echo "Dikali : $kali <br>";
echo "Dibagi : $bagi <br>";
echo "Sisa Bagi : $sisas <br>";
?>
```

# Komentar (Remark)



- `/* ... */`
- `//`
- `#`

# Latihan

## 1. Apa outPut Program di Bawah???

```
<?php
$a = 5;// sesuai 2 digit npm terakhir contoh 20184350045
$b = 4;
echo "$a == $b : ". ($a == $b);
echo "<br>$a != $b : ". ($a != $b);
echo "<br>$a > $b : ". ($a > $b);
echo "<br>$a < $b : ". ($a < $b);
echo "<br>($a == $b) && ($a > $b) : ".(($a != $b) && ($a > $b));
echo "<br>($a == $b) || ($a > $b) : ".(($a != $b) || ($a > $b));
?>
```

Terima Kasih

