

2018

[PENERAPAN STRUKTUR ALGORITMA DI PHP]

PEMROGRAMAN WEB LANJUT

S1 TEKNIK INFORMATIKA

AINUL YAQIN, M.KOM

TOPIK :

1. PHP Sintak Dasar
2. Aturan variabel, Konstanta, Operator
3. Struktur Kendali
4. Function
5. Array

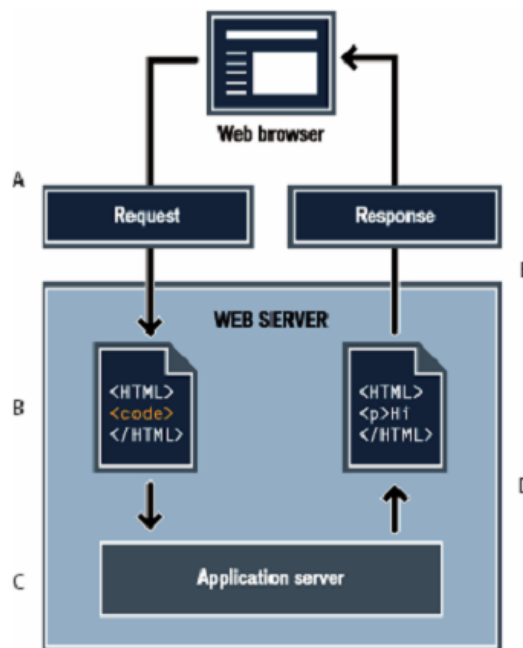
PHP

(PHP : Hypertext Preprocessor) adalah sebuah bahasa scripting yang paling populer saat ini. Berfungsi untuk membuat website dengan konten dinamis maupun aplikasi web lainnya. Berbeda dengan HTML yang hanya menampilkan konten statis, PHP dapat berinteraksi dengan database, file, folder bahkan dengan lingkungan sistem operasi. PHP termasuk kedalam golongan perangkat lunak interpreter. Yakni menerjemahkan baris demi baris program (tanpa mengcompilanya) kemudian mengeksekusinya.

PHP awal mula dikembangkan oleh Rasmus Lerdorf sebagai pengembangan dari Perl untuk menangani pengolahan form pada website pribadinya. Oleh karenanya, saat itu PHP disebut juga sebagai akronim dari Personal Home Page. Saat ini lebih dari 20 juta website menggunakan PHP backend aplikasi web.

1. Skema Aplikasi Web

Aplikasi web dinamis berbasiskan pada konsep arsitektur client - server, dimana server menangani apa yang dimininta oleh client, kemudian mengirimkan kembali hasilnya dalam bentuk HTML.



Skema aplikasi web pada gambar diatas dapat dijelaskan sebagai berikut:

- a. Client (web browser) meminta sebuah halaman pada sebuah website.
- b. Web server mencari halaman yang diminta, dan setelah ditemukan, memberikan halaman tersebut ke application server (PHP Interpreter).
- c. Application server mem-parse (membaca) dan memproses halaman tersebut.
- d. Application server mengirimkan halaman yang telah diproses kembali ke web server.
- e. Web server mengirimkan kembali halaman yang telah diproses kepada web browser dalam bentuk HTML.

2. Dasar PHP

2.1 Aturan Penulisan

Sebagaimana umumnya bahasa pemrograman, PHP mempunyai aturan-aturan tertentu dalam penulisan skripnya. Diantaranya sebagai berikut:

- a) Program disimpan dengan ekstensi .php, .php3, .php4, atau .php5.
- b) Semua skrip harus diapit oleh tanda:
 - `<?php` dan `?>`
 - `<script language="php">` dan `</script>`
 - `<? dan ?>`
 - `<% dan %>`
- c) Setiap perintah harus diakhiri dengan tanda titik koma (;).

Berbeda dengan bahasa banyak bahasa pemrograman web pada umumnya, skrip program PHP dapat langsung ditanamkan / disisipkan didalam HTML. Jika skrip PHP ditemukan dalam sebuah dokumen web, PHP Interpreter akan memproses skrip PHP dan hanya akan mengirimkan hasil berupa HTML ke browser client.

Latihan_01.php

```
<html>

  <head>

    <title>Program PHP Pertama Saya</title>

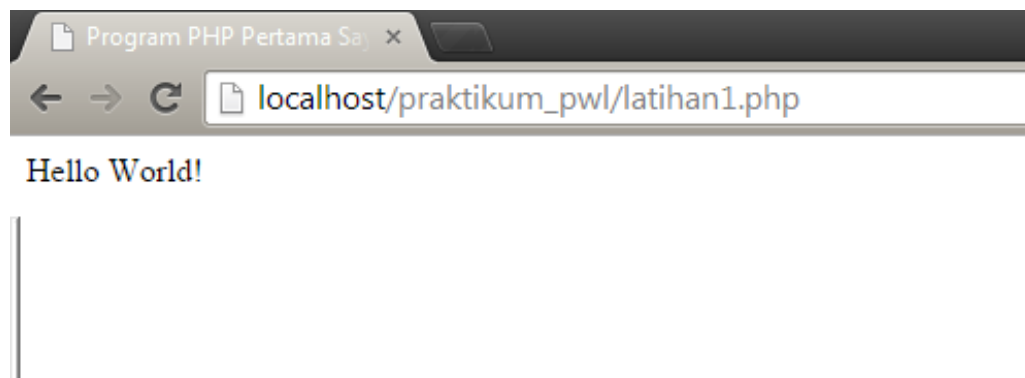
  </head>

  <body>

    <?php echo "Hello World!"; ?>

  </body>

</html>
```



Dalam sebuah program, terdapat dua hal penting: command (perintah) dan values (nilai-nilai). Command memiliki tugas spesifik yang telah ditentukan. Sedangkan values adalah informasi yang berkaitan dengan tugas yang dilakukan oleh command. Dari contoh-contoh sebelumnya, kita telah banyak menggunakan literal. Yaitu value yang telah diterapkan secara eksplisit didalam program pada sebuah command. Sebagai contoh baris program berikut:

```
echo "Hello World!";
```

Pernyataan diatas melakukan command `echo` dan menggunakan value literal yang diberikan setelahnya, yaitu `"Hello World!"`.

2.2 Variabel PHP

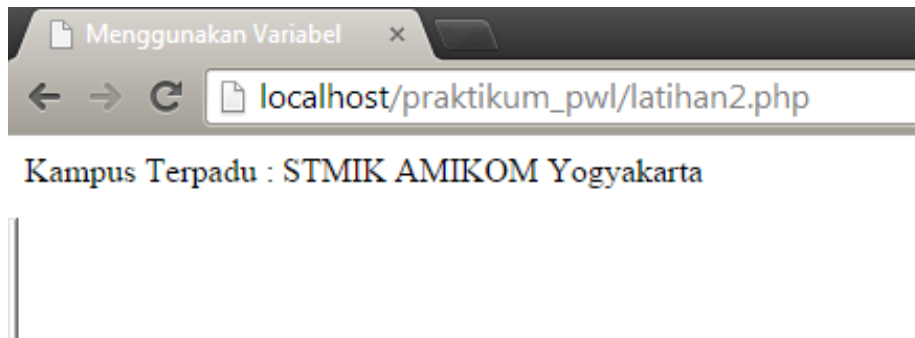
Variabel adalah sebuah tempat di memori untuk menyimpan data yang nilainya dapat berubah-ubah selama program dijalankan. Penulisan variabel didalam pemrograman PHP, ada aturan tertentu yang harus kita ketahui, diantaranya :

1. Variabel dimulai dengan tanda dollar (\$).
2. Harus dimulai dengan huruf atau underscore (_).
3. Tidak boleh mengandung spasi
4. Case sensitive atau huruf besar dan huruf kecil berbeda.
5. Karakter pertama setelah simbol dollar, tidak boleh menggunakan angka (harus huruf)
6. Setelah simbol dollar dan huruf, maka karakter selanjutnya boleh menggunakan angka
7. Contoh penulisan variabel :

Benar	Salah
\$variabel	\$var!abel
\$_nim	\$-nim
\$kelas01	\$01kelas
\$tepat_waktu	\$tepat-waktu

Contoh penggunaan variabel : Latihan_02.php

```
<html>
<head>
  <title>Menggunakan Variabel</title>
</head>
<body>
  <?php
    $kampus ="STMIK AMIKOM Yogyakarta";
    echo "Kampus Terpadu : $kampus";
  ?>
</body>
</html>
```



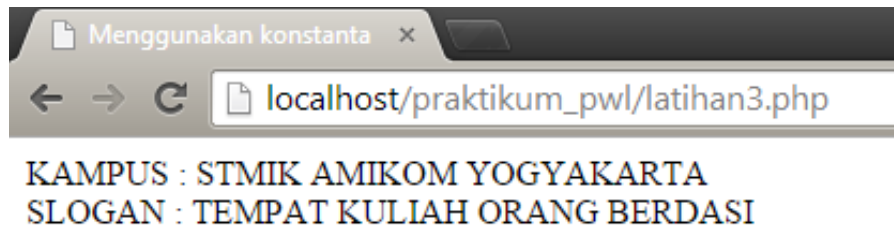
2.3 Konstanta

Konstanta adalah variabel yang nilainya tetap. Konstanta hanya diberi nilai pada awal program dan nilainya tidak pernah berubah selama program berjalan. PHP telah mendefinisikan beberapa konstanta, misalnya: `PHP_VERSION`, yaitu konstanta yang memberikan informasi tentang versi PHP yang digunakan. Selain konstanta yang telah disediakan oleh PHP kita juga dapat membuat konstanta sendiri. Aturan penulisan konstanta adalah sebagai berikut :

```
define ("nama_konstanta","nilai_konstanta")
```

Contoh penggunaan konstanta: Latihan_03.php

```
<html>
<head>
  <title>Menggunakan konstanta</title>
</head>
<body>
  <?php
    define("KAMPUS", "STMIK AMIKOM YOGYAKARTA");
    define("SLOGAN", "TEMPAT KULIAH ORANG BERDASI");
    print "KAMPUS : ".KAMPUS;
    echo "<br>SLOGAN : ".SLOGAN;
  ?>
</body>
</html>
```



2.4 Tipe Data

Pada PHP, tipe data variabel tidak didefinisikan oleh programmer, akan tetapi secara otomatis ditentukan oleh interpreter PHP. PHP memiliki 5 tipe data diantaranya :

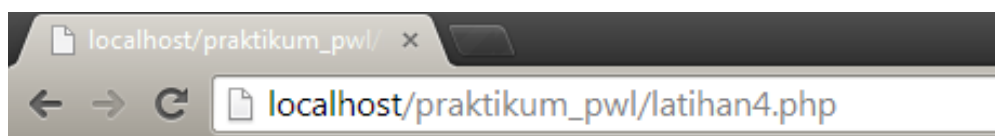
1. Integer (bilangan bulat)
2. Floating Point (bilangan pecahan atau desimal)
3. String (ditandai dengan adanya penggunaan tanda petik tunggal atau ganda)
4. Array (mengandung data dalam satu variabel index)
5. Object (ber-paradigma kepada object)

Contoh penulisan tipe data :

Tipe Data	Contoh	keterangan
Integer	<code>Sjumlah = 23;</code> <code>Snilai = -3;</code>	Bilangan bulat
Double	<code>Sipk = 3.85;</code> <code>Stotal = 13.00;</code>	Bilangan real
String	<code>Sjurusan = "Sistem Informasi";</code> <code>Sjenjang = "Strata Satu";</code>	Karakter, kalimat

Contoh penggunaan tipe data : File : latihan_04.php

```
<?php
$nim = "09.12.3508";
$nama = "M.Rezza Ardava";
$umur = 21;
$nilai = 90.25;
$status = TRUE;
echo "NIM : " . $nim. "<br>";
echo "Nama : $nama<br>";
print "Umur : " . $umur;
print "<br>";
printf ("Nilai : %.3f<br>", $nilai);
if ($status)
echo "Status : Aktif";
else
echo "Status : Tidak Aktif";
?>
```



NIM : 09.12.3508
Nama : M.Rezza Ardava
Umur : 21
Nilai : 90.250
Status : Aktif

2.5 Operator

2.5.1 Operator Aritmatika

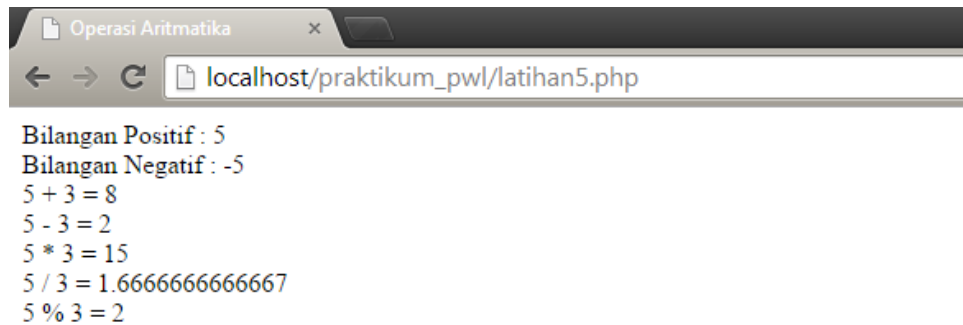
Operator aritmatika digunakan untuk melakukan perhitungan matematika.

Operator Nama Operasi Contoh

Operator	Nama Operasi	Contoh
+ (Unary)	Positif	+5
- (Unary)	Negatif	-5
+	Penjumlahan	5+5
-	Pengurangan	5-5
*	Perkalian	5*5
/	Pembagian	5/5
%	Hasil Bagi	5%5

Contoh Latihan5.php

```
<html>
  <head>
    <title>Operasi Aritmatika</title>
  </head>
  <body>
    Bilangan Positif : <?php echo +5 ?>
    <br />
    Bilangan Negatif : <?php echo -5 ?>
    <br />
    5 + 3 = <?php echo 5 + 3 ?>
    <br />
    5 - 3 = <?php echo 5 - 3 ?>
    <br />
    5 * 3 = <?php echo 5 * 3 ?>
    <br />
    5 / 3 = <?php echo 5 / 3 ?>
    <br />
    5 % 3 = <?php echo 5 % 3 ?>
  </body>
</html>
```



The screenshot shows a web browser window with the title 'Operasi Aritmatika'. The address bar displays 'localhost/praktikum_pwl/latihan5.php'. The page content lists several arithmetic operations and their results:

```

Bilangan Positif : 5
Bilangan Negatif : -5
5 + 3 = 8
5 - 3 = 2
5 * 3 = 15
5 / 3 = 1.66666666666667
5 % 3 = 2
    
```

2.5.2 Operator Perbandingan/Relasional

Operator	Keterangan
==	Memeriksa apakah operand kiri bernilai sama dengan operand kanan
>	Memeriksa apakah operand kiri lebih besar dari operand kanan
<	Memeriksa apakah operand kiri lebih kecil dari operand kanan
>=	Memeriksa apakah operand kiri lebih besar atau sama dengan operand kiri
<=	Memeriksa apakah operand kiri lebih kecil atau sama dengan operand kiri
!=	Memeriksa apakah operand kiri tidak bernilai sama dengan operand kanan

2.5.3 Operator Logika

Logika	and	\$a and \$b	TRUE jika \$a dan \$b TRUE
	&&	\$a && \$b	TRUE jika \$a dan \$b TRUE
	or	\$a or \$b	TRUE jika \$a atau \$b TRUE
		\$a \$b	TRUE jika \$a dan/atau \$b TRUE
	xor	\$a xor \$b	TRUE jika \$a atau \$b TRUE, tapi tidak keduanya
	!	!\$a	TRUE jika \$a FALSE

2.5.4 Operator Penugasan

Operator penugasan digunakan untuk memberikan nilai pada sebuah variabel.

Contoh:

```
$nama = "ainul";
```

Pada contoh diatas, operator penugasan “=” digunakan untuk memasukkan string "ainul" kedalam variabel \$nama. Selain operator penugasan “=”, juga dikenal beberapa operator penugasan lain. Diataranya adalah sebagai berikut:

- Operator +=

Digunakan untuk menambah variabel dengan bilangan tertentu. Contoh:

```
$a += 5; akan sama artinya dengan $a = $a + 5;
```

- Operator -=

Digunakan untuk mengurangi variabel dengan bilangan tertentu . Contoh:

```
$a -= 5; akan sama artinya dengan $a = $a - 5;
```

- Operator *=

Digunakan untuk mengalikan variabel dengan bilangan tertentu. Contoh:

```
$a *= 5; akan sama artinya dengan $a = $a * 5;
```

- Operator /=

Digunakan untuk membagi variabel dengan dengan bilangan. Contoh:

```
$a /= 5; akan sama artinya dengan $a = $a / 5;
```

- Operator %=

Digunakan untuk mencari sisa hasil bagi dari sebuah variabel dengan bilangan tertentu. Contoh:

$\$a \% = 5$; akan sama artinya dengan $\$a = \$a \% 5$;

- Operator $\&=$

Digunakan untuk melakukan operasi logika AND pada sebuah variabel.

Contoh:

$\$a \&= 5$; akan sama artinya dengan $\$a = \$a \& \text{true}$;

- Operator $|=$

Digunakan untuk melakukan operasi logika OR pada sebuah variabel. Contoh:

$\$a |= 5$; akan sama artinya dengan $\$a = \$a | \text{false}$;

- Operator $\wedge=$

Digunakan untuk melakukan operasi logika XOR pada sebuah variabel.

Contoh:

$\$a \wedge= 5$; akan sama artinya dengan $\$a = \$a \wedge \text{false}$;

- Operator $.=$

Digunakan untuk menambahkan string pada sebuah variabel. Contoh:

$\$a .= \text{"Agung Yunmar"}$; akan sama artinya dengan $\$a = \$a . \text{"Agung Yunmar"}$;

- Operator $++$

Digunakan untuk menambahkan nilai satu (increment) pada sebuah variabel.

Contoh:

$\$a++$; akan sama artinya dengan $\$a = \$a + 1$;

- Operator $--$

Digunakan untuk mengurangi nilai satu (decrement) pada sebuah variabel.

Contoh:

$\$a--$; akan sama artinya dengan $\$a = \$a - 1$;

Contoh Latihan6.php

```
<html>
  <head>
    <title>Operator Penugasan</title>
  </head>
  <body>
    <?php
      $a = 10; // mula mula nilai variabel $a = 10

      echo "1. Penugasan +=";
      echo "<br />";

      $a += 8;
      echo "Variabel \$a saat ini bernilai : ".$a;
      ?>
      <br />
      <br />
      <?php
        $a = 10; // nilai variabel $a dikembalikan
        menjadi 10

        echo "2. Penugasan ++";
        echo "<br />";

        $a++;
        echo "Variabel \$a saat ini bernilai : ".$a;
        ?>
      </body>
</html>
```

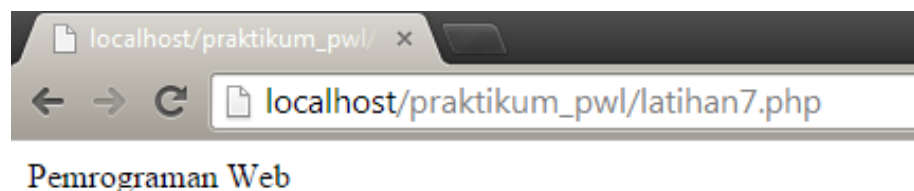
2.6 Operator Lain

2.6.1 Operator penggabungan string

Pada bahasa pemrograman PHP, string digabungkan dengan karakter . (titik).

Contoh Latihan7.php

```
<?php
$string_1 = "Pemrograman";
$string_2 = "Web";
echo $string_1." ".$string_2;
?>
```



2.6.2 Operator percabangan

Pada umumnya, operator percabangan dilakukan dengan struktur if-else, seperti pada contoh berikut:

```
If ($user == "ainul"){  
    echo "Selamat datang Ainul.";  
} else {  
    echo "Username salah.";  
}
```

Hal yang sama dapat dilakukan dengan cara:

```
echo ($user == "ainul" ? "Selamat datang Ainul." : "Username  
salah.");
```

Struktur diatas dapat dijelaskan sebagai berikut:

1. Pertama kali, PHP akan memeriksa apakah kondisi yang menjadi prasyarat bernilai benar. Yaitu apakah variabel \$user bernilai "Ainul".
2. Jika benar, pernyataan 1 yang dipilih, dan pernyataan 2 akan dipilih jika kondisi prasyarat bernilai salah.
3. Kemudian program akan menampilkan (echo) pernyataan yang telah dipilih tadi. Kondisi Pernyataan 1 Pernyataan 2