

TUGAS MINGGU KE-1

VARIABEL TYPEDATA DAN OPERATOR PADA PHP



NAMA : ILHAM AHMAD DHANI

NIM : E32232276

GOLONGAN : D

PROGRAM STUDI TEKNIK KOMPUTER

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI JEMBER

2024

PENDAHULUAN

A. Latar Belakang

PHP di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group. Situs resmi PHP beralamat di <http://www.php.net>. PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dan lainnya.

PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: Hypertext Preprocessor. PHP dapat digunakan dengan gratis (free) dan bersifat Open Source. PHP dirilis dalam lisensi PHP License, sedikit berbeda dengan lisensi GNU General Public License (GPL) yang biasa digunakan untuk proyek Open Source. Kemudahan dan kepopuleran PHP sudah menjadi standar bagi programmer web di seluruh dunia. Menurut wikipedia pada februari 2014, sekitar 82% dari web server di dunia menggunakan PHP. PHP juga menjadi dasar dari aplikasi CMS (Content Management System) populer seperti Joomla, Drupal, dan WordPress.

B. Pembahasan Teori

Seperti bahasa pemrograman lainnya, PHP juga memiliki aturan penulisan seperti case sensitivity (perbedaan antara huruf besar dan kecil), cara mengakhiri sebuah baris perintah, serta pengaruh penggunaan spasi dalam membuat kode program PHP.

Berikut adalah aturan dasar penulisan kode PHP:

a. Case Sensitivity (perbedaan huruf besar dan kecil) dalam PHP

PHP membedakan huruf besar dan huruf kecil (case sensitive) untuk penamaan variabel, sehingga *\$nama*, *\$Nama* dan *\$NAMA* akan dianggap sebagai 3 variabel yang berbeda. Sering kali error terjadi karena salah menuliskan nama variabel, yang seharusnya menggunakan huruf kecil ditulis dengan huruf besar. Sering kali error terjadi karena salah menuliskan nama variabel, yang seharusnya menggunakan huruf kecil ditulis dengan huruf besar.

b. Penulisan Baris Perintah dalam PHP

Statement (baris perintah) di dalam PHP adalah kumpulan perintah PHP yang menginstruksikan PHP untuk melakukan sesuatu. Baris perintah ini bisa terdiri dari satu baris singkat (seperti perintah `echo` untuk menampilkan text di layar) atau bisa sesuatu yang lebih rumit dan terdiri dari beberapa baris, seperti kondisi `if`, atau kode perulangan (loop). PHP menggunakan tanda semicolon (titik koma) “;” sebagai tanda akhir baris perintah. Kumpulan baris perintah yang menggunakan tanda kurung kurawal seperti kondisi `IF` atau perulangan (loop) tidak butuh tanda titik koma setelah kurung penutup.

c. Karakter Spasi dan Tab dalam PHP

Secara umum, karakter spasi dan tab diabaikan ketika mengeksekusi kode program PHP. Anda boleh memecah sebuah statement menjadi beberapa baris atau menyatukan beberapa statement dalam sebuah baris yang panjang. Kemudian menambahkan beberapa spasi atau tab di awal untuk memudahkan membaca kode program (indenting)

1. Variabel

Variabel adalah kode program yang digunakan untuk menampung nilai tertentu. Nilai yang disimpan di dalam variabel selanjutnya dapat dipindahkan ke dalam database, atau ditampilkan kembali ke pengguna.

Aturan pada penulisan variabel dalam PHP adalah sebagai berikut:

- a. Variabel di dalam PHP harus diawali dengan dollar sign atau tanda dollar (\$).
- b. Setelah tanda (\$), sebuah variabel PHP harus diikuti dengan karakter pertama berupa huruf atau underscore (“_”), kemudian untuk karakter kedua dan seterusnya bisa menggunakan huruf, angka atau underscore (“_”).

```
<?php  
$nama_alamat = "budi. Jember";  
echo $nama_alamat;  
<?
```

2. Tipe Data

Beberapa tipe data terdengar familiar, seperti tipe data angka, desimal dan text. Namun juga akan menemukan tipe data lain seperti boolean dan array.

- a. Tipe data Integer

Tipe data yang berupa angka bulat seperti: 1, 22, dan -172. Jika data yang dimiliki kemungkinan akan mengandung pecahan, maka tipe data yang digunakan adalah float. Jika tidak diberi tanda, maka bernilai positif. Jangkauan angka integer bergantung kepada kemampuan komputasi komputer, namun biasanya dimulai dari -2,147,483,648 sampai +2,147,483,647, atau 32bit. Tipe data integer digunakan juga untuk menampung angka base 16 (hexadesimal), base 8 (octal), dan base 2 (binary). Untuk mengetahui nilai maksimal tipe data integer pada komputer, PHP menyediakan konstanta (“*print PHP_INT_MAX*”).

```
<?php
$umur=18;
echo $umur;
<?
```

3. Tipe Data Float

Float (disebut juga tipe data floating point, atau real number) adalah tipe data angka yang memiliki bagian desimal di akhir angka, atau memiliki floating point (floating point adalah istilah dalam bahasa Inggris untuk menyebut tanda “titik” yang menandakan bilangan desimal).

PHP mendukung 2 cara penulisan tipe data float, yang pertama yaitu penulisan desimal, 0.17 atau 9.47 dan yang kedua berupa penulisan format scientific notation, seperti 0.314E1, atau 12.0E-3. Penggunaan scientific notation digunakan untuk menyederhanakan penulisan, 0.314E1 adalah sama dengan 0.314×10^1 atau 3.14, dan 12.0E-3 sama dengan 12.0×10^{-3} , atau 0.012. Sama seperti tipe data integer, variabel dengan tipe data float juga dapat melakukan operasi numerik seperti penambahan, pembagian, perkalian, dan lain-lain. Berikut contoh penulisan bilangan float dalam PHP:

```
<?php
$angka_float1= 0.78;
$angka_float2= 14.99;
$angka_scientific1=0.314E1;
$angka_scientific2=0.3365E-3;
echo $angka_float1; // 0.78
echo "<br />";
echo $angka_float2; //14.99
echo "<br />";
echo $angka_scientific1; //3.14
echo "<br />";
echo $angka_scientific2; //0.0003365
?>
```

4. Tipe Data String

Adalah tipe data yang berisi text, kalimat, atau kumpulan karakter. PHP menyediakan 4 cara penulisan tipe data string, yakni Single Quoted, Double Quoted, Heredoc, dan Nowdoc.

5. Operator

Operator adalah nilai asal yang digunakan didalam proses operasi. operator Aritmatika adalah operator matematis yang terdiri dari operator penambahan, pengurangan, perkalian, pembagian, modulus, dan eksponensial, sedangkan operator adalah instruksi yang diberikan untuk mendapatkan hasil dari proses tersebut. Operator dapat dibedakan menjadi 3, yaitu Operator Unary, Binary dan Ternary.

- a. Operator unary adalah operator yang hanya memiliki 1 operand, contohnya karakter – (tanda minus). Tanda minus digunakan membuat sebuah angka menjadi negatif, contohnya: -5, atau karakter + untuk menegaskan nilai positif, contohnya: +5.
- b. Operator binary adalah operator yang memiliki 2 operand. Operator jenis ini adalah yang paling banyak digunakan, misalkan 5×2 , atau $10/3$.
- c. Operator Ternary adalah operator yang memiliki 3 operand. Didalam PHP hanya dikenal 1 operator ternary, yaitu operator kondisi (? :).

Operator Aritmatika

Operator	Nama	Contoh	Keterangan
+	Addition	$\$x + \y	Tambah
-	Subtrction	$\$x - \y	Kurang
*	Multiplication	$\$x * \y	Kali
/	Division	$\$x / \y	Bagi
%	Modulus	$\$x \% \y	Sisa bagi
**	Exponentiation	$\$x ** \y	Pangkat

C. Rumusan Masalah

1. Tanda “ ” pada nilai variabel biasanya menandakan sebuah string, tetapi pada saat dilakukan operasi penjumlahan ternyata menghasilkan nilai jumlah seperti pada umumnya penjumlahan aritmatika, mengapa bisa terjadi?
2. Apa yang membedakan pembagian dengan operator / dan operator modulo % ?
3. Tambahkan kode program operator agar bisa menghasilkan nilai gabungan string tugas1 dan tugas2 menjadi 9080?

D. Capaian Pembelajaran

1. Mahasiswa mampu memahami dan mengimplementasikan PHP dalam pengembangan WEB
2. Mahasiswa mampu memahami dan mengimplementasikan Variabel, Type data dan Operator pada PHP

D. Tujuan

1. Ketepatan dalam memahami dan mengimplementasikan PHP dalam pengembangan WEB
2. Ketepatan dalam memahami dan mengimplementasikan Variabel, Type data dan Operator pada PHP

PROSEDUR KERJA

A. Alat dan Bahan

1. Client Server: xampp
2. Text Editor: Visual Code/Sublime Text 3
3. Management Storage: Git dan Github
4. Tools Dependency Manager Multiplatform: Composer

B. Langkah Kerja

1. Variable

```
<?php
$nim="E32232276";
$nama="Ilham";
$prodi="Teknik Komputer";
echo "Selamat Praktikum";
echo "<br/>";
echo $nim;
echo "<br/>";
echo $nama;
echo "<br/>";
echo $prodi;
?>
```

2. Tipe Data

```
<?php
$x = 5985;
var_dump($x);
echo "<br>";
$x = -345; // negative number
var_dump($x);
echo "<br>";
$x = 0x8C; // hexadecimal number
var_dump($x);
echo "<br>";
$x = 047; // octal number
var_dump($x);
echo "<br>";
$x = 10.365;
var_dump($x);
echo "<br>";
$x = 2.4e3;
var_dump($x);
```



```

echo "<br>";
$x = 8E-5;
var_dump($x);
echo "<br>";
echo strlen("Hello world!");
echo "<br>";
echo strpos("Hello world!", "world");
?>

```

3. Operator 1

```

<?php
$tugas1=90;
$tugas2=80;
$jumlah=$tugas1 + $tugas2;
$rerata=$jumlah/2 ;
echo "Nilai Tugas I :".$tugas1."<br>";
echo "Nilai Tugas II:".$tugas2."<br>";
echo "Jumlah Tugas :".$jumlah."<br>";
echo "Rata Rata Tugas :".$rerata."<br>";
$gabungan = $tugas1.$tugas2;
echo "Nilai Tugas1 dan Tugas2: " . $gabungan."<br>";
?>

```

4. Operator 2

```

<?php
$x = 10;
$y = 6;
echo ($x + $y);
echo "<br>";
echo ($x - $y);
echo "<br>";
echo ($x * $y);
echo "<br>";
echo ($x / $y);
echo "<br>";
echo ($x % $y);
echo "<br>";
$a = "Hello";
$b = $a . " world!";
echo $b;
echo "<br>";
echo $a . " -- " . $b . " ini string operator ";
?>

```

HASIL PRAKTIKUM

1. String dan Aritmatika

Tanda " " dalam pemrograman biasanya digunakan untuk merepresentasikan string, tetapi dalam kondisi di mana terlibat dalam operasi matematika dengan tipe data numerik, seperti integer dan string, tanda tersebut dapat mengalami konversi menjadi nilai numerik. Dalam situasi tersebut, nilai string akan dikonversi menjadi tipe data numerik yang sesuai.

```
$angka1 = "10";  
$angka = 5;  
$hasil = $angka1 + $angka;  
echo $hasil;
```

2. Modulus dan Division

Operator	Nama	Perbedaan
/	Division	1. Menghasilkan hasil dari pembagian 2. Hasil pembagian adalah float jika terdapat sisa
%	Modulus	1. Output sisa hasil bagi 2. Hasil dari operasi modulo adalah sisa pembagian dari bilangan pertama yang lebih besar oleh bilangan kedua

3. Operator

```
<?php  
$tugas1=90;  
$tugas2=80;  
$jumlah=$tugas1 + $tugas2;  
$rerata=$jumlah/2 ;  
echo "Nilai Tugas I :".$tugas1."<br>"; // Nilai Tugas I : 90  
echo "Nilai Tugas II:".$tugas2."<br>"; // Nilai Tugas II : 80  
echo "Jumlah Tugas :".$jumlah."<br>"; // Jumlah Tugas : 170  
echo "Rata Rata Tugas :".$rerata."<br>"; // Rata Rata Tugas 85  
$gabungan = $tugas1.$tugas2;  
echo "Nilai Tugas1 dan Tugas2: " . $gabungan."<br>"; // 9080  
?>
```

KESIMPULAN

Dalam pemrograman, tanda " " sering digunakan untuk merepresentasikan string, yang merupakan kumpulan karakter. Namun, dalam beberapa konteks, seperti saat terlibat dalam operasi matematika dengan tipe data numerik, seperti integer dan string, tanda tersebut dapat mengalami konversi menjadi nilai numerik. Dalam situasi ini, nilai string akan dikonversi menjadi tipe data numerik yang sesuai untuk operasi matematika. Sebagai contoh, dalam PHP, jika ada operasi penjumlahan antara bilangan bulat dan string yang mengandung nilai numerik, PHP akan mengonversi nilai string tersebut menjadi bilangan bulat dan kemudian melakukan operasi penjumlahan aritmatika.

Selain itu, dalam pemrograman, terdapat operator modulus (%) yang berguna untuk menghitung sisa pembagian dari dua bilangan. Operasi modulus membagi bilangan pertama (dividend) dengan bilangan kedua (divisor) dan menghasilkan sisa pembagiannya. Contoh penggunaan operator modulus adalah saat perlu menghitung sisa dari pembagian, misalnya untuk menentukan bilangan genap atau ganjil.

Pemahaman tentang perilaku tanda " ", operator matematika, dan operator modulus menjadi krusial dalam pemrograman, terutama saat melakukan operasi terkait numerik dan manipulasi string. Kemampuan untuk mengelola konversi tipe data dan memanfaatkan operator matematika dengan benar dapat mempermudah proses pengembangan dan memastikan keakuratan operasi dalam kode program.

DAFTAR PUSTAKA

i gede. (n.d.). Variabel, Type data dan Operator pada PHP. In *bkpm pemrograman web* (Vol. 1 dan 2, pp. 1–32). essay, politeknik negeri jember. Retrieved February 2, 2024, from https://elearning-jti.polije.ac.id/pluginfile.php/60465/mod_resource/content/1/Minggu%201%20-%20Acara%201%20dan%202.pdf.