

Bab

3

BAB 3 – PHP

Pokok Bahasan

- Pengantar
- Sintaks
- Case Sensitivity
- Comment
- Variable
- Echo and print

Tujuan Pembelajaran

- Mampu membuat halaman web dinamis menggunakan PHP

Waktu

- 2(2 x 50) menit

❖ PHP - Pengantar

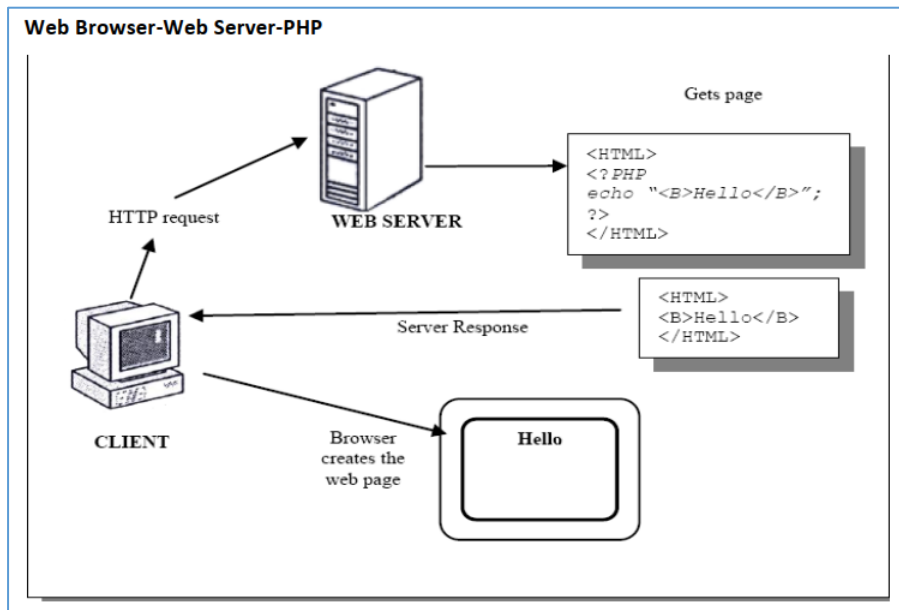
PHP adalah bahasa scripting yang menyatu dengan HTML dan dijalankan pada server side. Artinya semua sintaks yang kita berikan akan sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke browser hanya hasilnya saja. PHP/FI merupakan nama awal dari PHP. Dibuat pertama kali oleh Rasmus Lerdoff. Awalnya merupakan program CGI yang dikhususkan untuk menerima input melalui form yang ditampilkan dalam web browser.

PHP dikatakan sebagai sebuah server-side embedded script language artinya sintaks-sintaks dan perintah yang kita berikan akan sepenuhnya dijalankan oleh server tetapi disertakan pada halaman HTML biasa. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada web browser, tetapi prosesnya secara keseluruhan dijalankan di server. Apakah HTML saja tidak cukup ? Ya, tentu saja HTML tidak cukup, salah satunya adalah karena sifat HTML yang statis. HTML hanya berguna untuk desain, saat kita browsing, maka yang kita lihat itu sebenarnya adalah interpretasi dari HTML.

Sebagai contoh saat kita menggunakan Gmail untuk mengirim email, maka tampilan yang dapat kita lihat tersebut merupakan interpretasi dari HTML, sedangkan untuk dapat mengirim email, digunakan bahasa pemrograman web seperti PHP. Dengan catatan disini kita anggap HTML statis, sedang PHP dinamis. Sebelumnya kita harus menyamakan persepsi mengenai dinamis disini, yang dimaksud dinamis disini bukannya web yang dapat gerakgerak, tetapi dinamis berarti web yang dapat diubah-ubah tanpa mengubah langsung kedalam file webnya. PHP secara resmi merupakan singkatan dari PHP-Hypertext Preprocessor.

Jika bermaksud membuat Web menggunakan PHP, maka terlebih dahulu harus mengaktifkan Web Server. Untuk mencoba skrip PHP, tidak perlu mempunyai server yang terhubung ke internet. Kita dapat mengujinya dengan menggunakan sebuah komputer yang bertindak sebagai server dan sekaligus sebagai klien. Web Server yang dapat digunakan : Apache, IIS, Xitami.

Skema hubungan



Pada prinsipnya server akan bekerja apabila ada permintaan dari client. Dalam hal ini client menggunakan kode-kode PHP untuk mengirimkan permintaan ke server (dapat dilihat pada gambar dibawah). Ketika menggunakan PHP sebagai server-side embedded script language maka server akan melakukan hal-hal sebagai berikut:

1. Membaca permintaan dari client/browser
2. Mencari halaman/page di server
3. Melakukan instruksi yang diberikan oleh PHP untuk melakukan modifikasi pada halaman/page

❖ PHP - Sintaks

Buatlah satu file PHP baru dengan nama **index.php** di folder `htdocs/pemweb` kemudian tambahkan script berikut ini

```
<!DOCTYPE html>
<html lang="en">

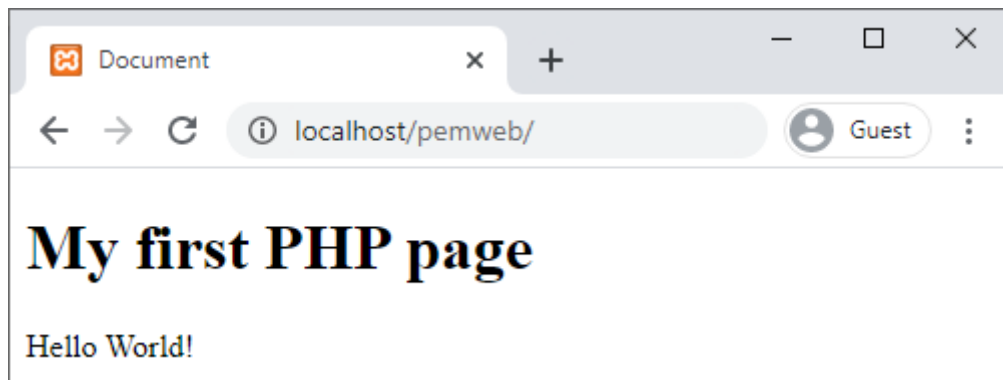
<head>
  <title>Document</title>
</head>

<body>
  <h1>My first PHP page</h1>

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

</body>
</html>
```

Buka browser dan ketikkan `localhost/pemweb` maka tampilannya akan menjadi seperti ini

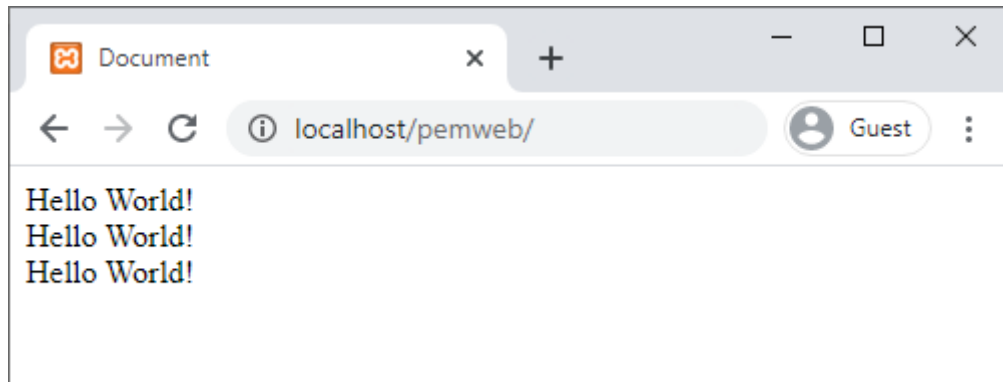


❖ PHP - Case Sensitivity

Untuk keyword atau fungsi di PHP user-defined tidak case sensitive. Ganti script di dalam tag `<body>` dengan script berikut

```
<?php
ECHO "Hello World!<br>";
echo "Hello World!<br>";
EcHo "Hello World!<br>";
?>
```

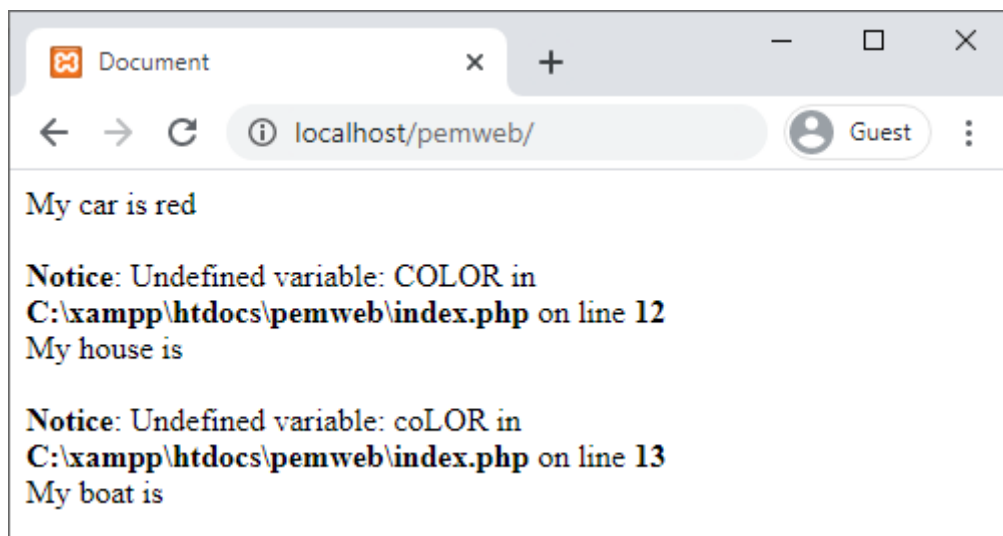
Refresh kembali browser



Sedangkan, apabila variabel yang dibuat oleh programmer bersifat case sensitive.

Ganti script dalam tag <body> dengan script berikut

```
<?php
$color = "red";
echo "My car is " . $color . "<br>";
echo "My house is " . $COLOR . "<br>";
echo "My boat is " . $coLOR . "<br>";
?>
```



❖ PHP - Comment

Terdapat tiga macam cara penulisan komentar di PHP:

1. <!-- komentar -->

Tulisan apapun yang berada di antara '<!--' dan '-->' akan dianggap sebagai komentar.

Cara seperti ini sangat berguna dan efisien untuk pemberian komentar dengan lebih dari satu baris

2. // komentar

Tulisan di baris yang sama setelah '//' akan dianggap sebagai komentar. Cara ini berguna untuk pemberian komentar singkat yang tak lebih dari 1 baris saja.

3. # komentar

Sama seperti '//', tulisan di baris yang sama setelah '#' akan dianggap sebagai komentar. Cara ini berguna untuk pemberian komentar singkat yang tak lebih dari 1 baris saja.

Contohnya adalah sebagai berikut

```
<!-- <?php  
$color = "red";  
echo "My car is " . $color . "<br>";  
echo "My house is " . $COLOR . "<br>";  
echo "My boat is " . $coLOR . "<br>";  
> -->
```

❖ PHP - Variable

Penulisan variabel diawali dengan tanda \$, dengan beberapa aturan penamaan, misalnya:

- Diawali dengan tanda \$ diikuti nama variabel
- Diawali dengan huruf atau garis bawah (underscore)
- Tidak bisa diawali dengan nomor
- Hanya dapat terdiri dari alfa-numerik karakter dan garis bawah
- Bersifat case-sensitif (\$cat dan \$CAT adalah dua variabel yang berbeda)

Beberapa tipe data yang bisa digunakan atau didukung oleh PHP adalah sebagai berikut:

- String
- Integer
- Float (floating point numbers - also called double)
- Boolean
- Array
- Object
- NULL

Contoh deklarasi variabel dalam PHP

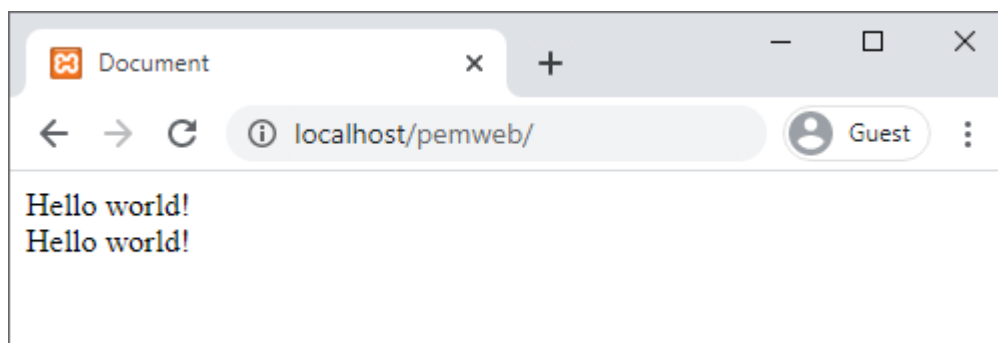
```
<?php
$txt = "Hello world!";
$x = 5;
$y = 10.5;
?>
```

String

Penulisan string dapat ditulis di dalam tanda kutip satu atau petik dua "Hello world".

```
<?php
$x = "Hello world!";
$y = 'Hello world!';

echo $x;
echo "<br>";
echo $y;
?>
```

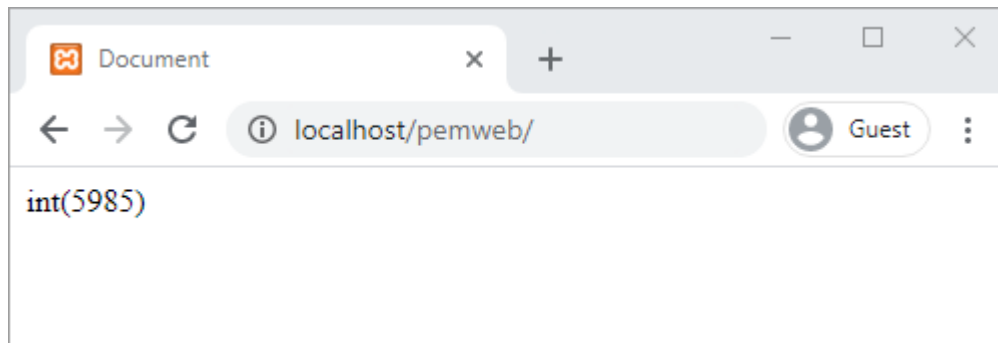


Integer

Terdapat beberapa aturan terkait dengan penggunaan tipe data integer yaitu:

- Harus mempunyai minimal satu digit
- Tidak diperbolehkan mempunyai koma decimal
- Dapat bernilai positif atau negatif
- Fungsi var_dump() digunakan untuk mengetahui tipe data dan isi variabel

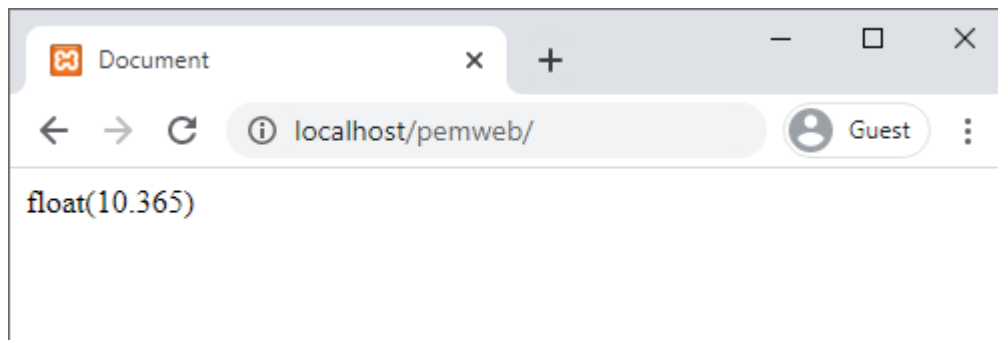
```
<?php
$x = 5985;
var_dump($x);
?>
```



Float

Variabel dengan isi nilai decimal

```
<?php
$x = 10.365;
var_dump($x);
?>
```



Boolean

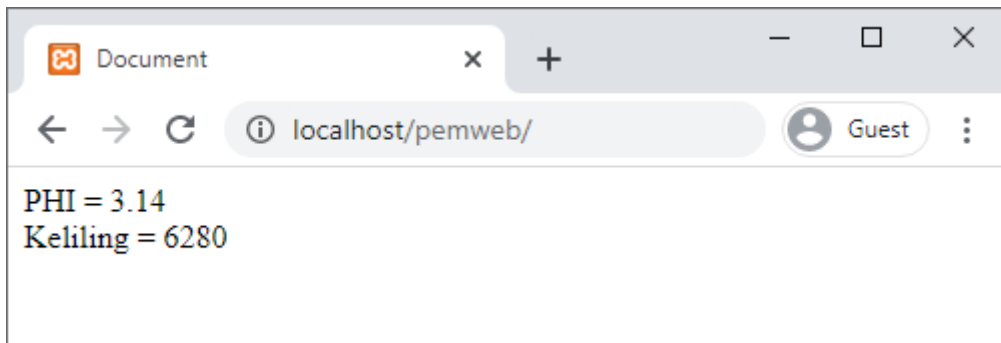
Tipe data ini bernilai TRUE atau FALSE

```
<?php
$x = true;
$y = false;
?>
```

Konstanta

Konstanta menyatakan nilai yang tetap di dalam program. Penulisannya selalu diapit tanda petik tunggal atau petik ganda.

```
<?php
define("PHI", 3.14);
$jari_jari = 1000;
$keliling = 2 * PHI * $jari_jari;
printf("PHI = %s<br>\n", PHI);
printf("Keliling = %s <br>\n", $keliling);
?>
```

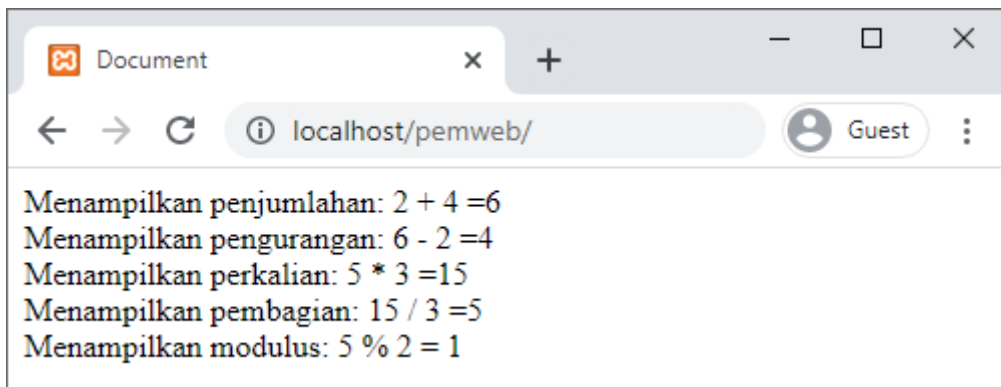



Operator aritmatik

Berikut ini adalah contoh penggunaan operator aritmatik

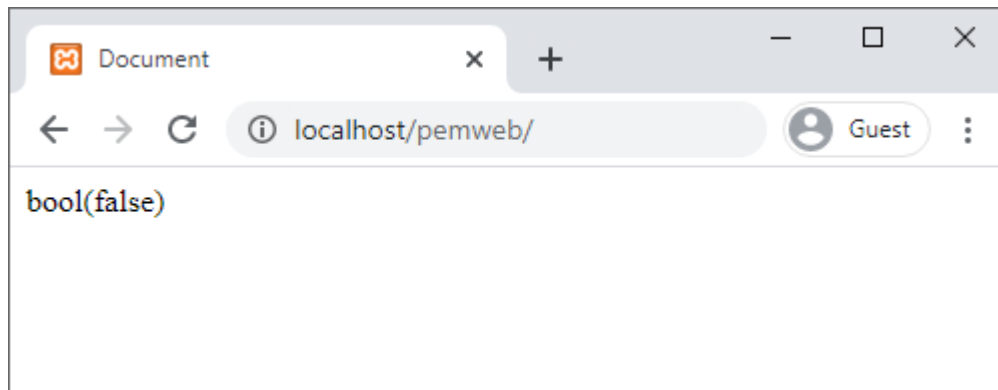
```
<?php
$penjumlahan = 2 + 4;
$pengurangan = 6 - 2;
$perkalian = 5 * 3;
$pembagian = 15 / 3;
$modulus = 5 % 2;

echo "Menampilkan penjumlahan: 2 + 4 =" . $penjumlahan . "<br>";
echo "Menampilkan pengurangan: 6 - 2 =" . $pengurangan . "<br>";
echo "Menampilkan perkalian: 5 * 3 =" . $perkalian . "<br>";
echo "Menampilkan pembagian: 15 / 3 =" . $pembagian . "<br>";
echo "Menampilkan modulus: 5 % 2 =" . $modulus . "<br>";
?>
```



Operator perbandingan

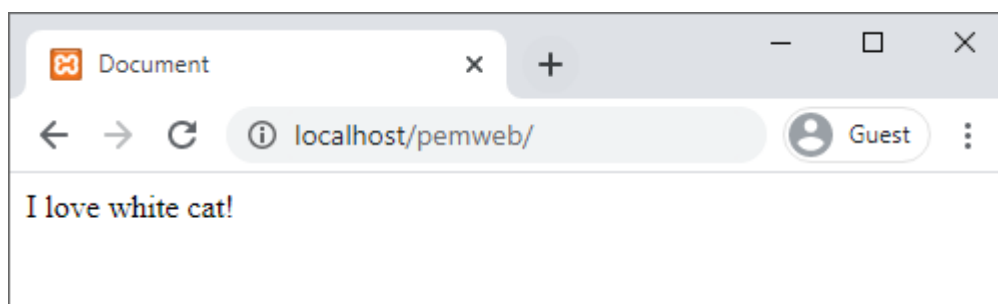
```
<?php
$x = 4;
$y = 5;
var_dump($x == $y)
?>
```



❖ PHP – Echo and print

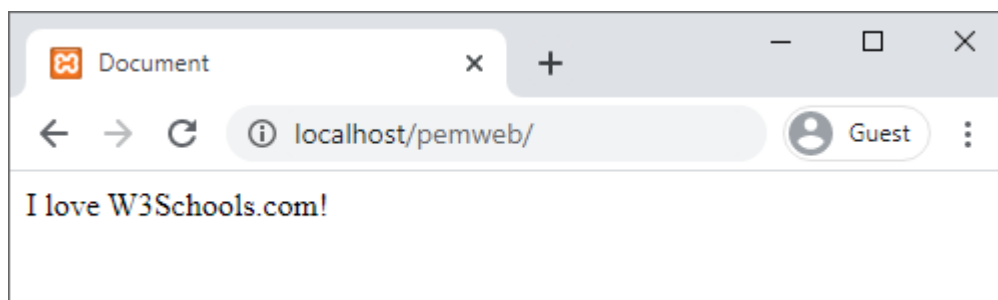
Fungsi ini digunakan untuk menampilkan pesan ke layar.

```
<?php
$txt = "white cat";
echo "I love $txt!";
?>
```



Contoh penulisan lain:

```
<?php
$txt = "W3Schools.com";
echo "I love " . $txt . "!";
?>
```



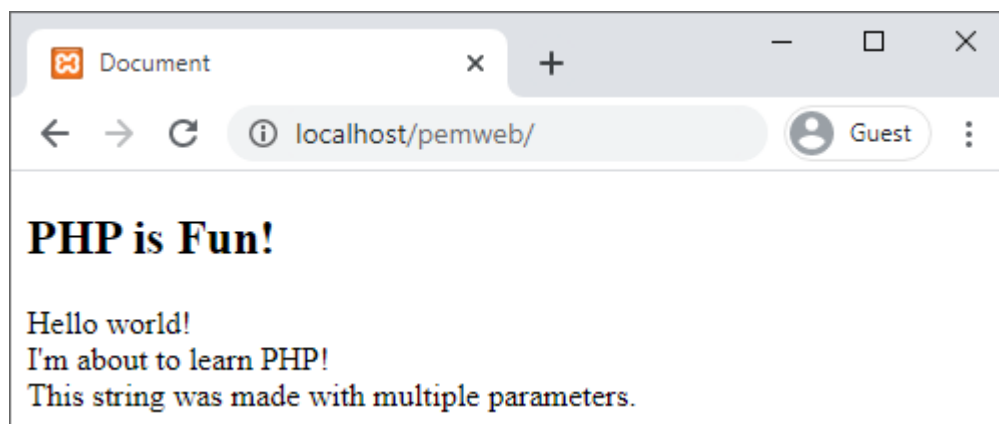
The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters (although such usage is rare) while print can take one argument. echo is marginally faster than print. (w3schools.com).

Perbedaan antara echo dan print adalah

Echo	Print
No return value	Return value
Multiple parameters	One argument
faster	slower

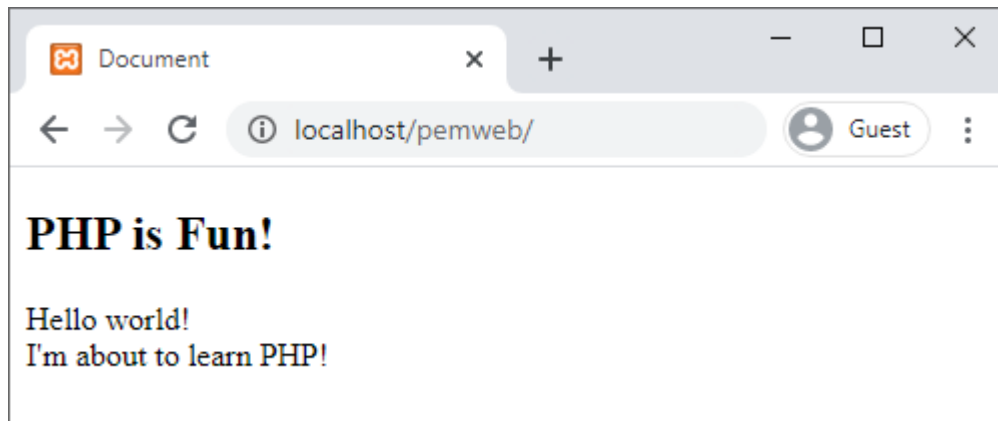
```
<?php
echo "<h2>PHP is Fun!</h2>";
echo "Hello world!<br>";
echo "I'm about to learn PHP!<br>";
echo "This ", "string ", "was ", "made ", "with multiple
parameters.";
?>
```

Echo dapat menampilkan tag HTML, pada contoh kali ini kata “PHP is fun” ditulis menggunakan heading 2.



Print dapat menampilkan tag HTML, pada contoh kali ini kata “PHP is fun” ditulis menggunakan heading 2.

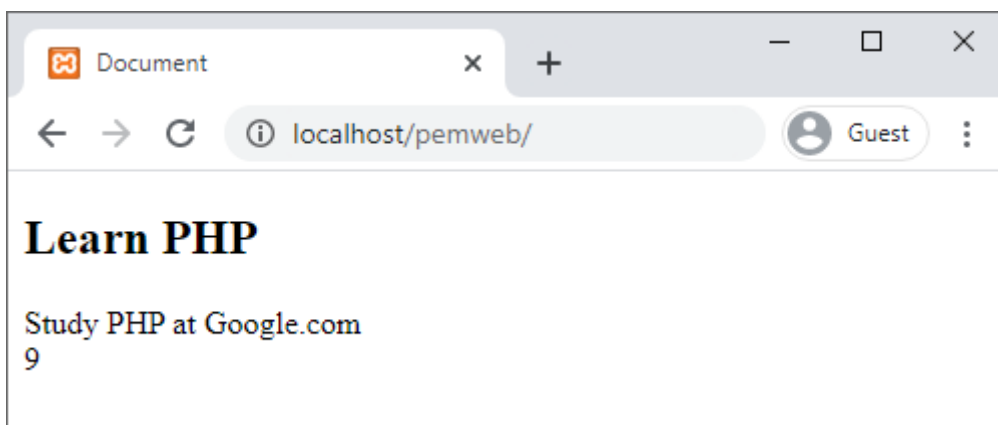
```
<?php
print "<h2>PHP is Fun!</h2>";
print "Hello world!<br>";
print "I'm about to learn PHP!";
?>
```



Echo untuk menampilkan variable

```
<?php
$txt1 = "Learn PHP";
$txt2 = "Google.com";
$x = 5;
$y = 4;

echo "<h2>" . $txt1 . "</h2>";
echo "Study PHP at " . $txt2 . "<br>";
echo $x + $y;
?>
```



Print untuk menampilkan variable

```
<?php
$txt1 = "Learn PHP";
$txt2 = "Google.com";
$x = 5;
$y = 4;

print "<h2>" . $txt1 . "</h2>";
print "Study PHP at " . $txt2 . "<br>";
print $x + $y;
?>
```

