**LAPORAN I**

**DASAR-DASAR PHP**

Disusun untuk Memenuhi Matakuliah Praktikum Pemrograman Berbasis Web Yang Dibimbing oleh Heru Wahyu Herwanto, S.T., M.KOM

****

Oleh:

**Umi Kholifah**

**110533430510**

**S1 PTI 2011 offering B**

**UNIVERSITAS NEGERI MALANG**

**FAKULTAS TEKNIK**

**JURUSAN TEKNIK ELEKTRO**

**S1 PENDIDIKAN TEKNIK INFORMATIKA**

**OKTOBER, 2013**

**LAPORAN KE-1**

**DASAR-DASAR PHP**

1. **Tujuan**
2. Memahami struktur dasar dokumen PHP
3. Mampu membuat dokumen PHP yang baik dan benar.
4. Mampu memanfaatkan elemen-elemen dasar untuk menampilkan informasi
5. **Dasar Teori**
6. **PHP**

PHP sendiri sebenarnya merupakan singkatan dari “Hypertext Preprocessor”, yang merupakan sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML. Sebagian besar sintaks dalam PHP mirip dengan bahasa C, Java dan Perl, namun pada PHP ada beberapa fungsi yang lebih spesifik. Sedangkan tujuan utama dari penggunaan bahasa ini adalah untuk memungkinkan perancang web yang dinamis dan dapat bekerja secara otomatis.

1. **Tipe Data**
2. **Integer**

Tipe data integer adalah tipe data yang berguna untuk menyimpan bilangan bulat, bukan desimal. Sebagai contoh (1),(2),(3),(4),(5),(-1),(-2),(-3),(-4) dan lain-lain. Tipe data ini memiliki range antara -2,147,483,648 sampai dengan +2,147,483,648 platform 32bit.

Namun bila terdapat data diluar jangkauan tersebut, maka secara otomatis PHP akan mengkonversikannya ke dalam tipe data Floating Point.

Integer merupakan Tipe Data yang berisikan bilangan bulat negatif dan positif atautanpa ada nilai pecahan.

1. **Double Floating**

Tipe data floating point numbers biasa juga disebut dengan “double”, ” float” atau “real” adalah tipe data yang berguna untuk menyimpan bilangan desimal. Sebagai contoh (0.1),(1.3),(1.7),(1.8),(9.7),(2.4) dan lain-lain.

1. **Boolean**

Tipe Data ini adalah tipe data yang paling sederhana. Hanya berupa true atau false. Cara memasukan ke dalam variabel adalah dengan memberikan nilai true atau falsepada variabel tersebut. Penulisannya tidak dipengaruhi penggunaan huruf besar atau kecil.

1. **String**

String adalah sebuah tipe data yang terdiri dari kata, bisa berupa kata tunggal maupun kalimat. Penulisan string harus diapit oleh tanda petik, baik petik tunggal(‘ ‘) maupun petik ganda (” “).

1. **Objek**

Tipe data object dibuat dengan tujuan agar para programmer terbiasa dengan OOP. Tipe data object bisa berupa bilangan, variabel atau fungsi. Data ini dapat dimasukkan ke dalam kode program sehingga meringkas beberapa fungsi dan memperkecil ukuran file.

1. **Array**

Array atau Larik merupakan Tipe Compound Primitif, terdapat pada bahasa-bahasa pemrograman lain. Tipe data array digunakan untuk menyimpan banyak data dalam satu variabel. Jenis array dalam php ada 3 (tiga) macam yakni : numeric array, associative array, multidimensional array.

1. **Null**

NULL adalah Tipe Data yang tidak memuat apapun. Setiap Variabel yang diset menjadi Tipe Data NULL ini akan menjadikan Variabel tersebut kosong, seperti dalam penggunaan fungsi unset(). Cara untuk menset Variabel menjadi NULL adalah dengan memasukan nilai NULL pada variabel yang dimaksud.

1. **Resources**

Tipe Data Spesial yang satu ini di khususkan untuk menyimpan resource, sumber atau alamat. Variabel tersebut hanya dapat diciptakan oleh suatu fungsi khusus yang mengembalikan nilai berupa resource seperti penggunaan fungsi fopen, opendir, mysql\_connect, mysql\_query dan semacamnya.

1. **Komentar**

Komentar adalah bagian dari skrip PHP yang tidak diterjemahkan oleh browser. Fungsi komentar hanyalah sebagai dokumentasi dari skrip PHP yang kita buat atau merupakan penjelesan-penjelasan dari skrip tersebut. Terdapat 3 tipe komentar yang umum digunakan dalam skrip PHP :

1. Komentar 1: /\* tulis komentar disini \*/
2. Komentar 2: // tulis komentar disini
3. Komentar 3 : # tulis komentar disini

Pada tipe komentar yang pertama, semua yang berada diantara tanda /\* dan \*/ akan dianggap sebagai komentar oleh browser walaupun berada pada baris yang berbeda. Sedangkan pada tipe komentar ke-2 dan ke-3, browser hanya akan menganggap semua tulisan setelah tanda // atau # sebagai komentar jika berada pada baris yang sama dengan tanda tersebut.

1. **Tag PHP**

Banyak cara untuk menyisipkan PHP dalam script HTML, ada berbagai macam bentuk tag yang dapat digunakan, antara lain :

1. Cara I

<?php //menandai awal tag

...

?> //menandai akhir tag

1. Cara II

<? //menandai awal tag

...

?> //menandai akhir tag

1. Cara III

<% //menandai awal tag

...

%> //menandai akhir tag

Cara ini sama dengan tag pada ASP. Opsi ini bisa dilakukan jika nilai asp\_tags pada php.ini bernilai on.

1. Cara IV

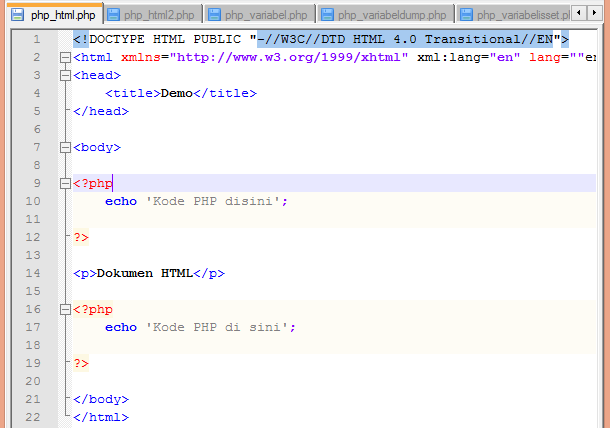
<script language=”php”> //menandai awal script

...

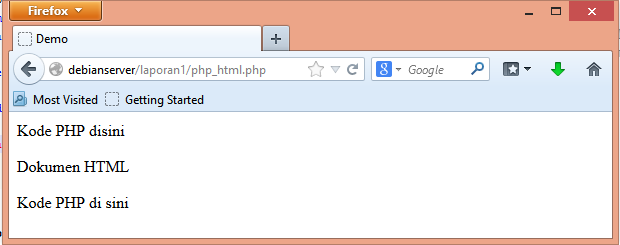
</script> //menandai akhir script

1. **Latihan**
2. **Latihan 1 Program PHP**
3. Kode HTML di PHP

* Source Code



* Screen shot

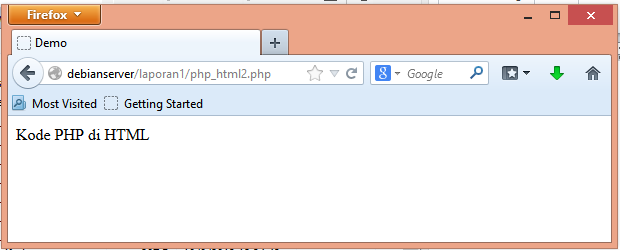


1. Kode-kode Kecil di dalam HTML

* Source Code

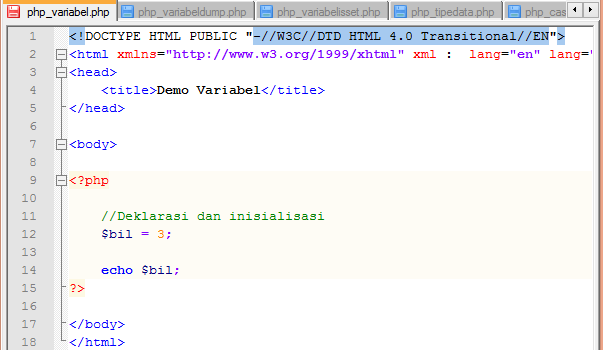


* Screen shot

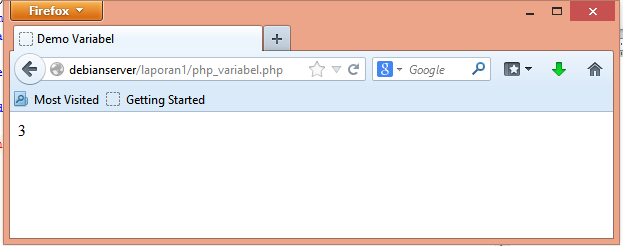


1. **Latihan 2 Variabel**
2. Variabel

* Source Code

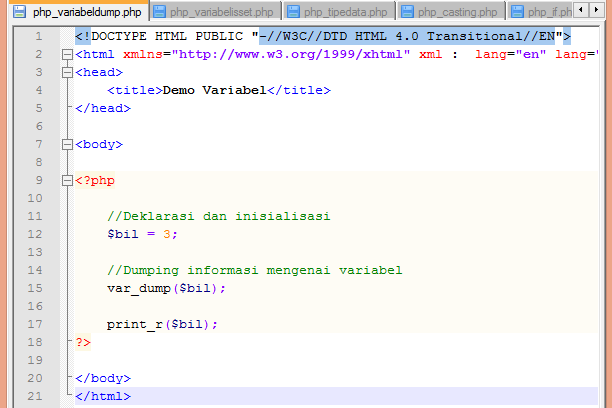


* Screen shot

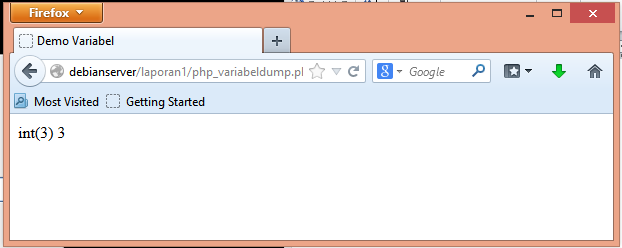


1. Var\_dump

* Source Code



* Screen shot



1. Perbedaan

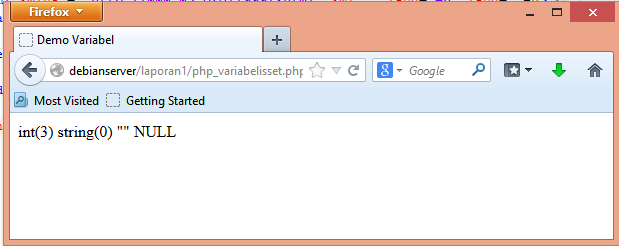
Vardump() menampilkan tipe data.

1. Isset

* Source Code

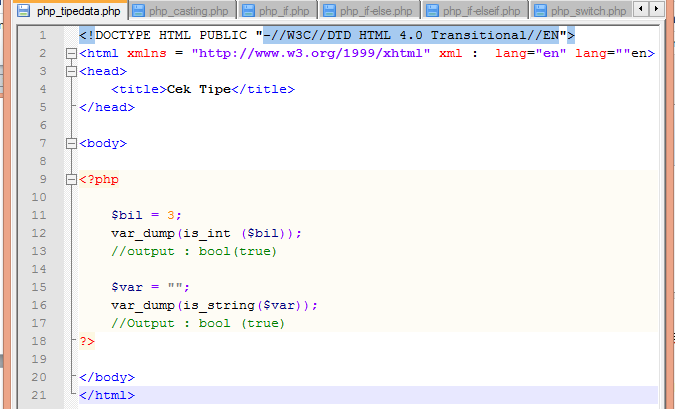


* Screen shot

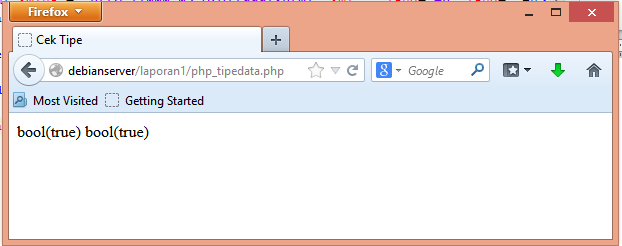


1. **Latihan 3 Tipe Data dan Casting**
2. Uji tipe data variabel

* Source Code

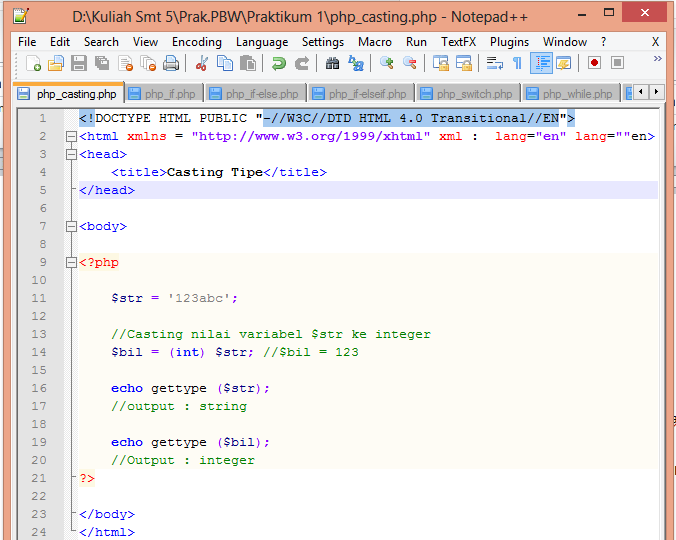


* Screen shot

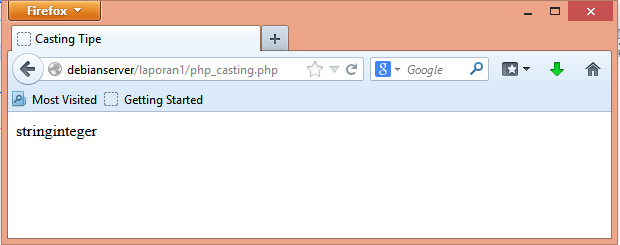


1. Casting

* Source Code

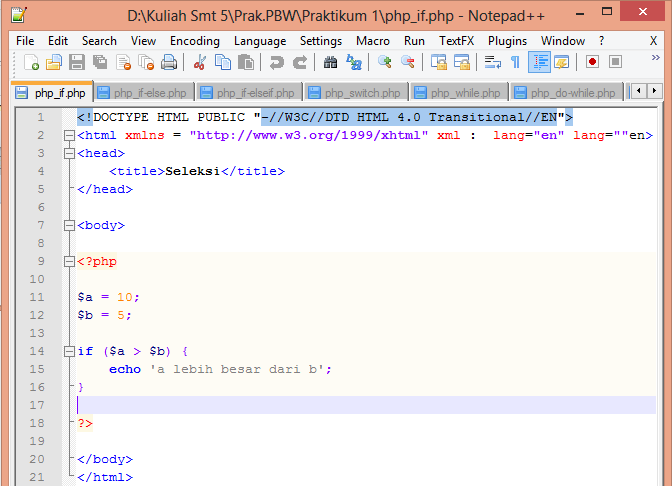


* Screen shot

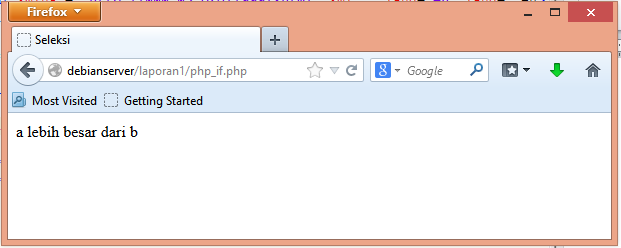


1. **Latihan 4 Pertanyaan Seleksi**
2. Pernyataan if

* Source Code

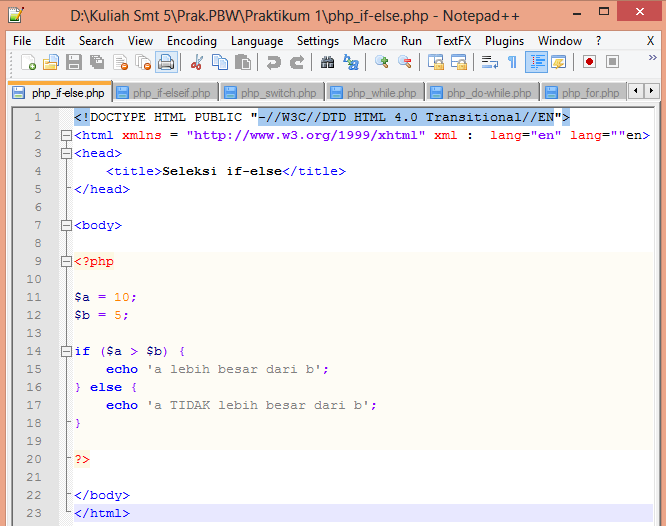


* Screen shot

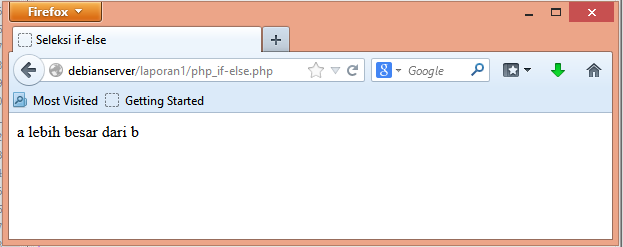


1. Pertanyaan if-else

* Source Code

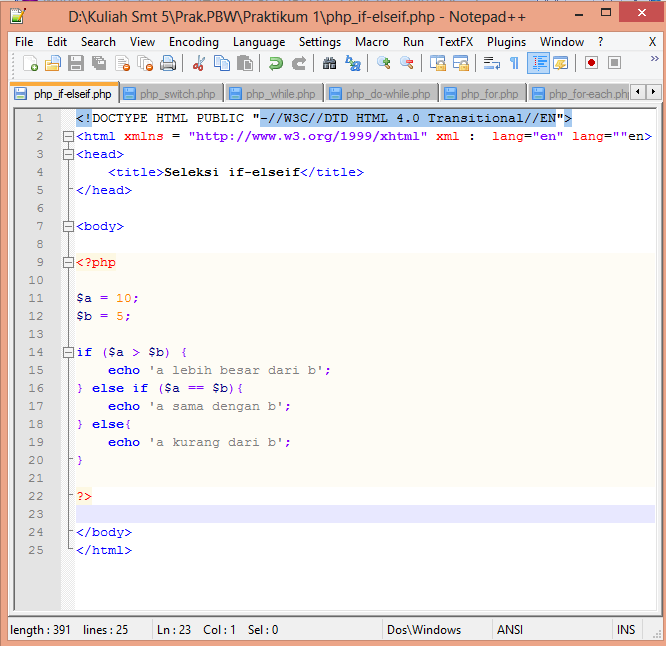


* Screen shot

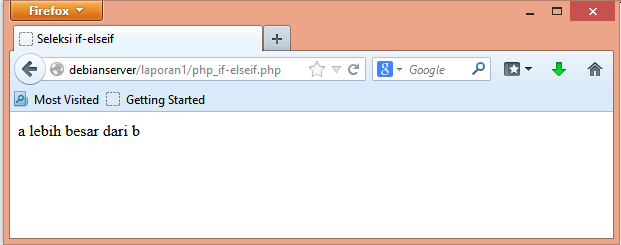


1. Pernyataan if-elseif

* Souce Code

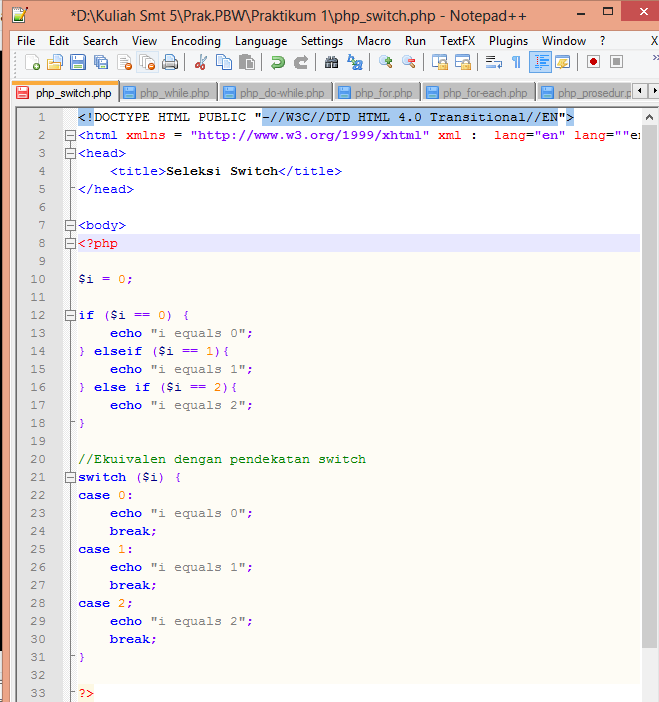


* Screen shot



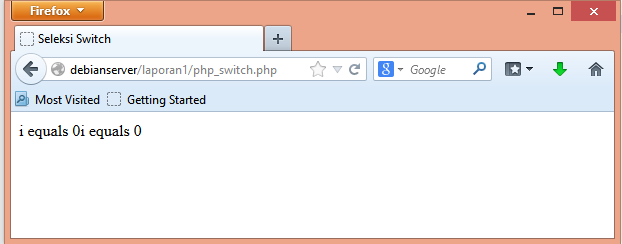
1. Pertanyaan switch

* Source Code



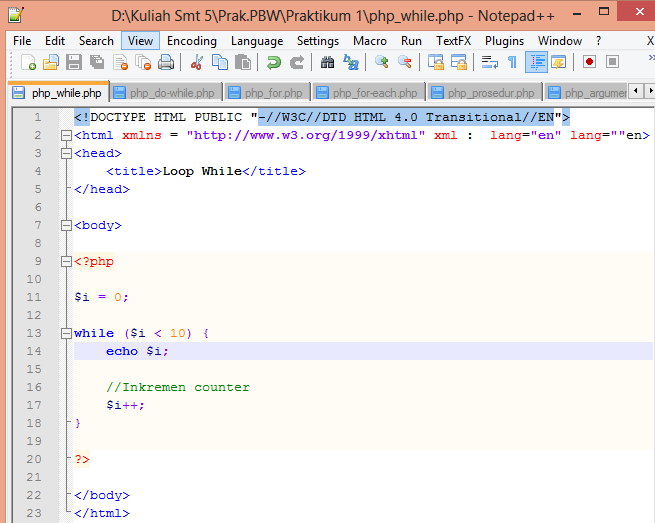


* Screen shot

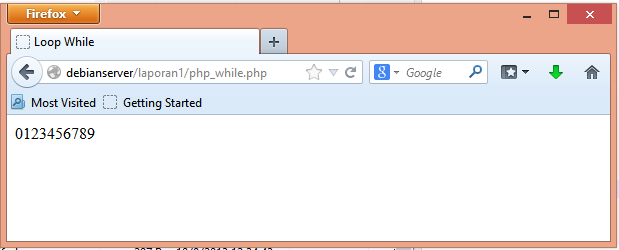


1. **Latihan 5 Pengulangan**
2. Pengulangan while

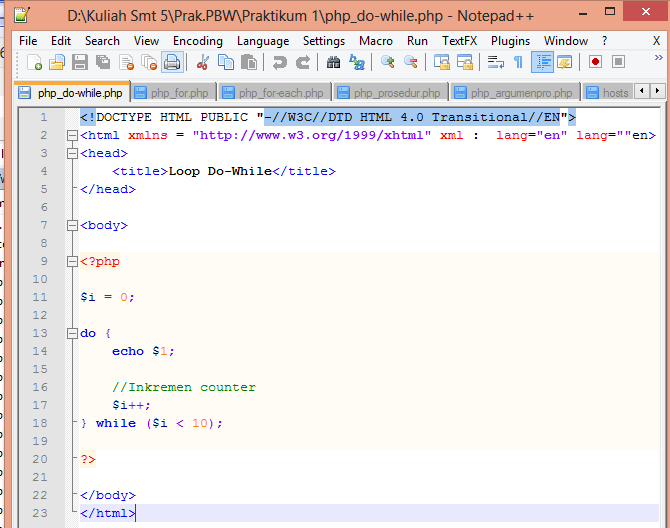
* Source Code

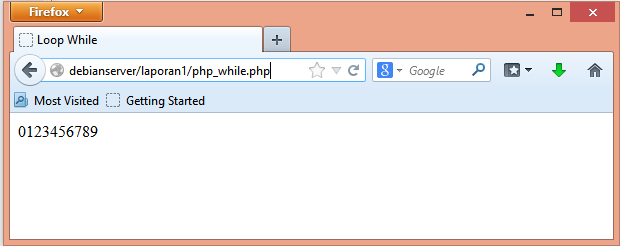


* Screen shot



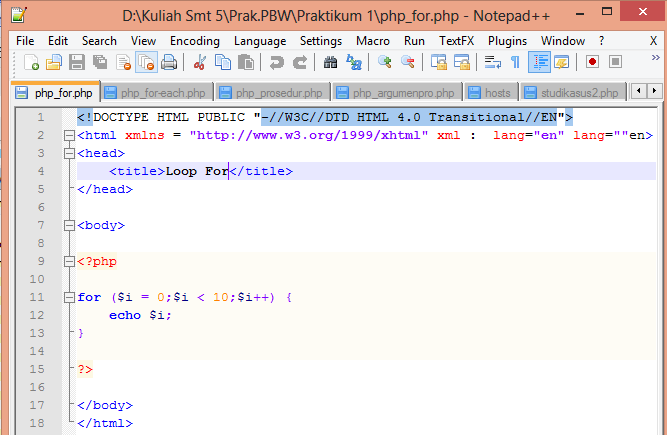
1. Pengulangan do-while

* Source Code
* Screen shot

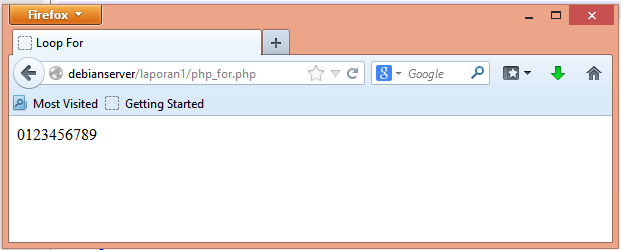


1. Pengulangan for

* Source Code

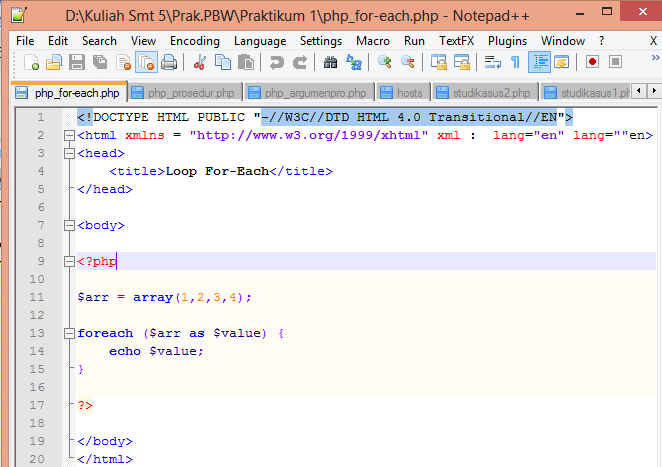


* Screen shot

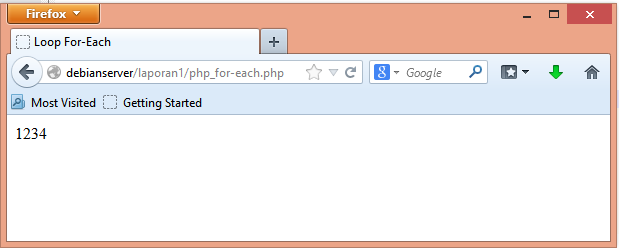


1. Pengulangan foreach

* Source Code

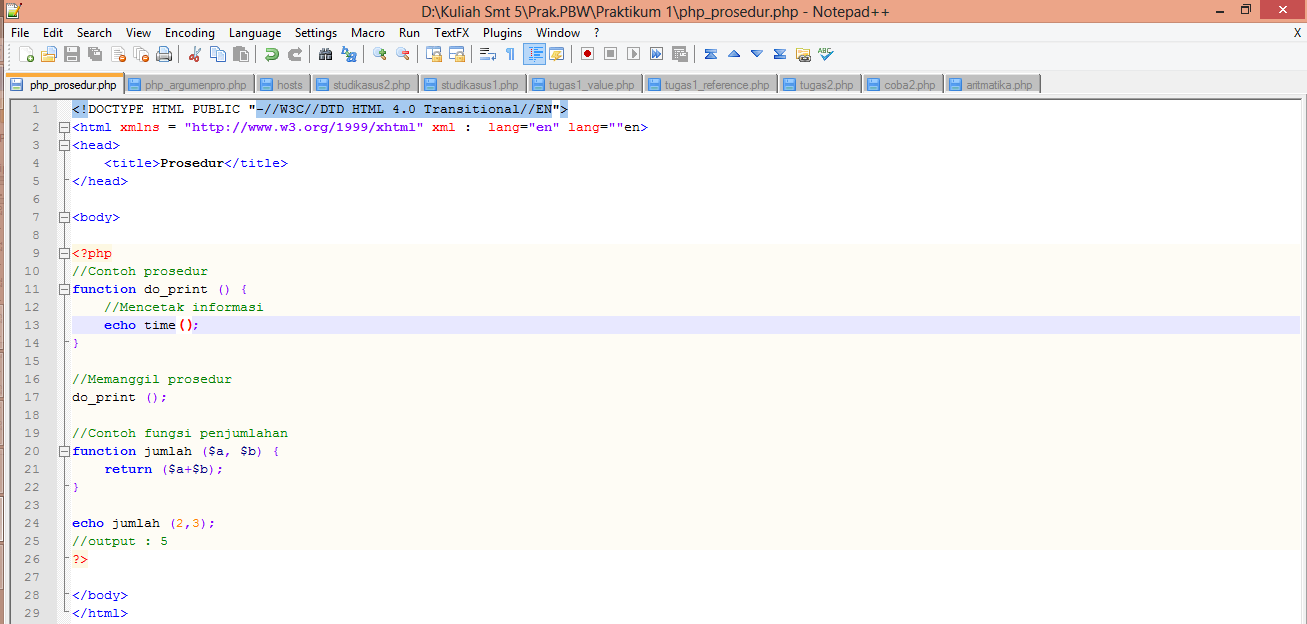


* Screen shot

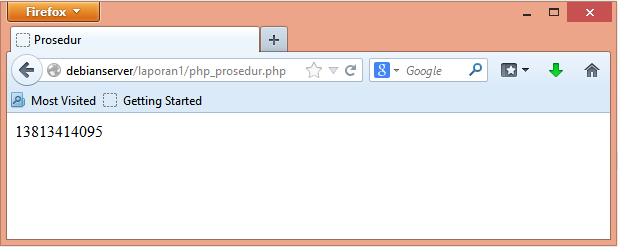


1. **Latihan 6 Fungsi dan Prosedur**
2. Definisi Fungsi/Prosedur

* Source Code

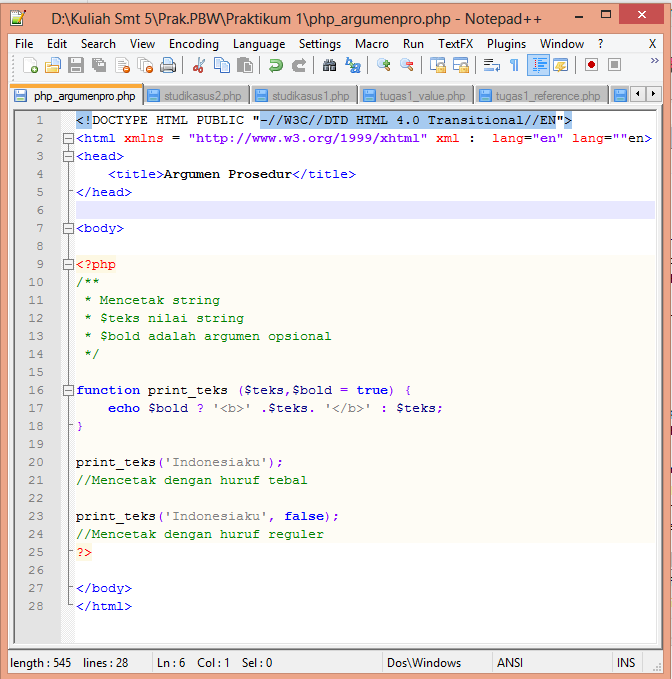


* Screen shot

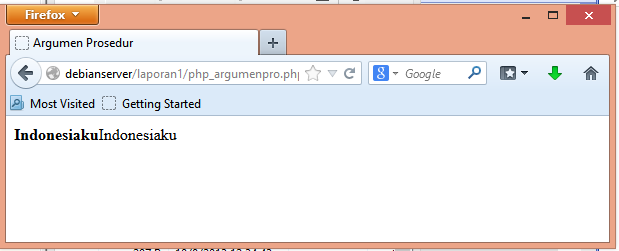


1. Argumen Fungsi/Prosedur

* Source Code



* Screen shot



1. **Studi Kasus**
2. **Studi Kasus 1**

* **Source Code**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Greeting</title>

</head>

<center>

<?php

$Bulan = array("01" => "Januari","02" => "Februari","03" => "Maret","04" => "April", "05" => "Mei","06" => "Juni","07" => "Juli", "08" => "Agustus","09" => "September","10" => "Oktober","11" => "November","12" => "Desember");

$Hari = array("Sunday" => "Minggu","Monday" => "Senin","Tuesday" => "Selasa","Wednesday" => "Rabu","Thursday" => "Kamis","Friday" => "Jum'at","Saturday" => "Sabtu");

$hari = date("l");

$tanggal = (int)date("d");

$bulan = date("m");

$tahun = date("Y");

$waktu = date("H : m : s"); // G - hour, 24-hour format without leading zeros; i.e. "0" to "23" ; i - minutes; i.e. "00" to "59" ; s - seconds; i.e. "00" to "59"

if(date("H") >= 4 AND date("H") < 12)

{

$greeting = "Selamat Pagi";

}

elseif(date("H") >= 12 AND date("H") < 17)

{

$greeting = " Selamat Siang";

}

else

{

$greeting = "Selamat Malam";

}

echo "<h1>==== <b><i>$greeting</i></b> ====</h1>";

echo "<h2>Sekarang hari $Hari[$hari], tanggal $tanggal $Bulan[$bulan] $tahun</h2><br>";

echo "<h2>Waktu menunjukkan pukul $waktu WIB</h2><br>";

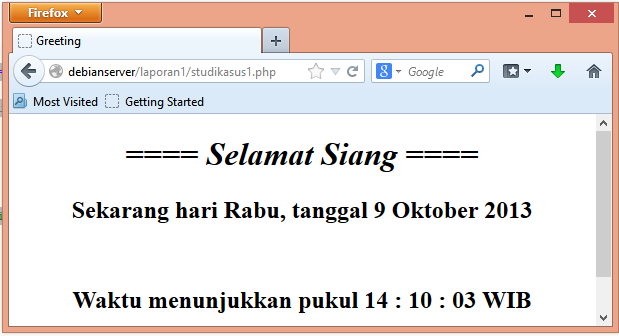
?>

</center>

</body>

</html>

* Screen shot



1. **Studi Kasus 2**

* **Source Code**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Matriks PHP</title>

</head>

<center>

<h3>\_-\_ Membuat MATRIKS dengan PHP \_-\_</h3>

<table>

<form method="get">

<tr>

<td> Masukkan baris <br><br></td>

<td>: <input type="text" size="15" name="baris"><br><br></td>

</tr>

<tr>

<td> Masukkan kolom<br><br> </td>

<td>: <input type="text" size="15" name="kolom"><br><br></td>

</tr>

<tr>

<td></td>

<td align="center"> <input type="submit" value="Generate"></td>

</tr>

</form>

</table>

<br>

<?php

$baris = $\_GET["baris"];

$kolom = $\_GET["kolom"];

function generate($baris, $kolom){

$cell = 0;

echo "<table border='2' cellpadding='10'>";

for($i = 0; $i < $baris; $i++){

echo "<tr>";

for($j = 0; $j < $kolom; $j++){

++$cell;

echo "<td>$cell</td>";

}

echo "</tr>";

}

echo "</table>";

}

if(isset($baris) AND isset($kolom)){

generate($baris, $kolom);

}

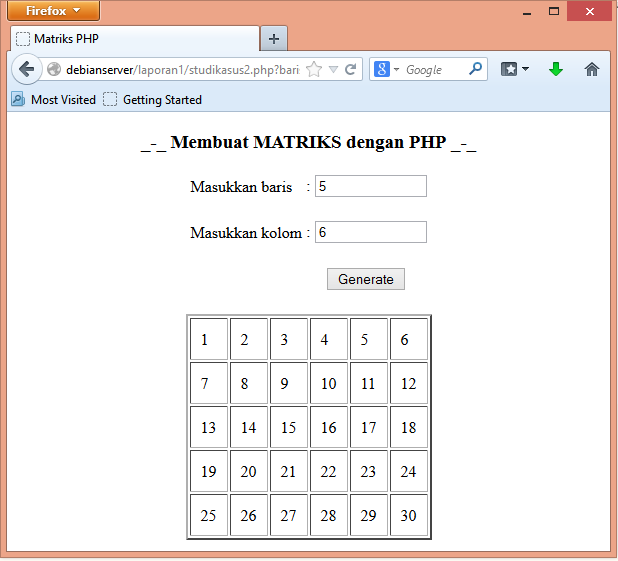
?>

</center>

</body>

</html>

* Screenshot

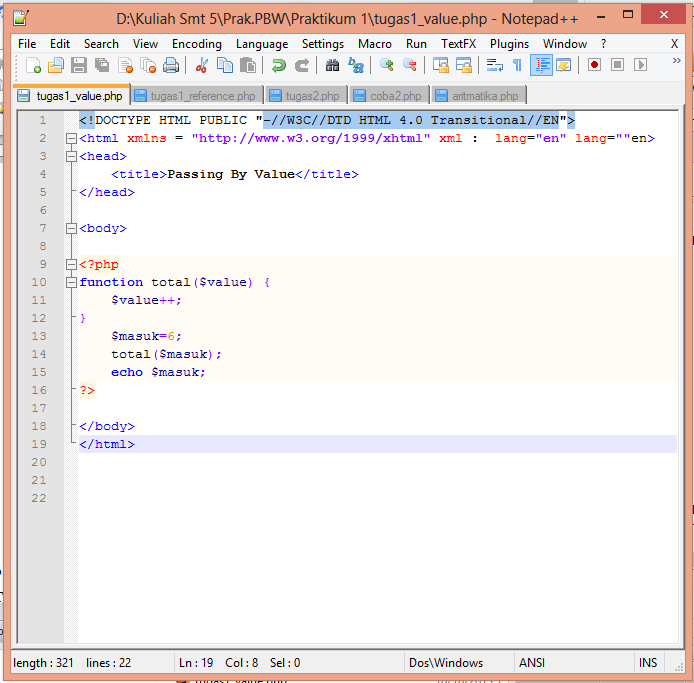


1. **Tugas Praktikum**
2. **Tugas Praktikum 1**
3. **Passing by Value**

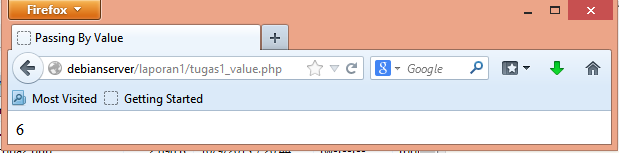
Variabel yang diberikan oleh parameter hanya akan diolah didalam fungsi, variabel yang berada diluar fungsi akan tetap. Maka variabel yang dipakai oleh suatu fungsi tidak akan mempengaruhi variabel yang ada diluar. Semua nilai yang di-pass masuk atau keluar dari fungsi, artinya PHP membuat  kopian dari nilai original dan nilai kopian itulah yang kita akses  dan kita manipulasi, bukan nilai originalnya. Nilai original tidak akan berubah, walaupun telah dilakukan manipulasi pada nilai. Ketika *pass-by-value* terjadi, method membuat sebuah salinan dari nilai variabel yang dikirimkan ke method. Walaupun demikian, method tidak dapat secara langsung memodifikasi nilai variabel pengirimnya meskipun parameter salinannya sudah dimodifikasi nilainya di dalam method.

Contoh :

* Source Code



* Screen shot



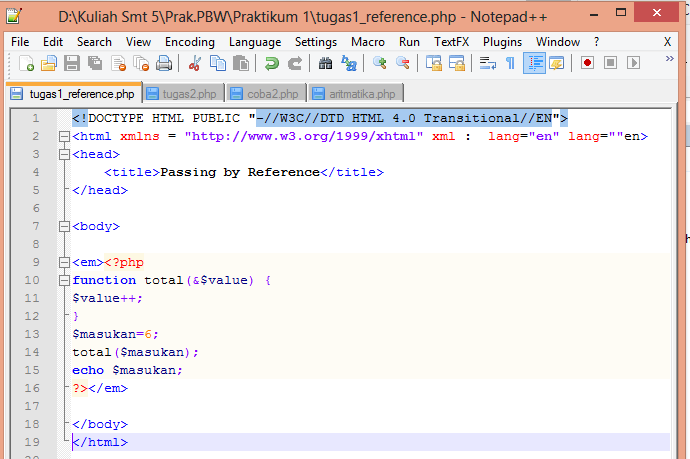
1. **Passing by Reference**

Passing by reference kebalikan dari Passing By Value, hasil olahan yang telah diproses di dalam fungsi dapat digunakan di luar fungsi, dengan kata lain value yang di luar fungsi akan turut berubah seiring perubahan yang terjadi pada variabel didalam fungsi. Passing By Reference memberikan nilai aslinya untuk diakses dan dimanipulasi. Untuk mengubah jadi passing by value menjadi reference, kita hanya cukup menambahkan operator '&' pada argumennya.

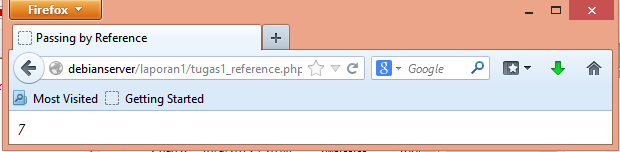
Ketika sebuah pass-by-reference terjadi, alamat memori dari nilai pada sebuah variabel dilewatkan pada saat pemanggilan method. Hal ini berarti bahwa method menyalin alamat memori dari variabel yang dilewatkan pada method. Ini tidak seperti pada pass-by-value, method dapat memodifikasi variabel asli dengan menggunakan alamat memori tersebut, meskipun berbeda nama variabel yang digunakan dalam method dengan variabel aslinya, kedua variabel ini menunjukkan lokasi dari data yang sama.

Contoh ;

* Source code



* Screen shot



1. **Tugas Praktikum 2**

* **Source Code tabelcell.html**

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>Generate Table Cell</title>

</head>

<script language="JavaScript" type="text/javascript">

function getmax() {

var R = parseInt(document.getElementById('JumlahRow').value);

var C = parseInt(document.getElementById('JumlahColum').value);

var X = parseInt(document.getElementById('JumlahCell').value);

var cellmax = document.getElementById('maxcells');

var total = 'N/A';

total = R \* C;

cellmax.value = new String(total);

if (X > total){

alert('Cell Total Yang Anda Masukkan Terlalu Besar, Nilai Maksimum Cells = ' + total);

document.getElementById('CellsTotal').value = new String();

}

}

</script>

<body>

<form method="post" action="tabelcell.php">

<h3 align="center" class="style1">Tabel Fleksibel Dengan Generate Table</h3>

<div align="center">

<table width="327" border="0" bgcolor="#FFFFFF">

<tr bgcolor="#FF0000">

<td width="121" style="text-align:center">Rows</td>

<td width="196"><strong>= </strong>

<input name="JumlahRow" type="text" id="JumlahRow" onKeyUp="getmax();" onfocus="this.select();"></td>

</tr>

<tr bgcolor="#FF0000">

<td style="text-align:center"><label>Columns</label></td>

<td><strong>= </strong>

<input name="JumlahColum" type="text" id="JumlahColum" onKeyUp="getmax();" onfocus="this.select();"></td>

</tr>

<tr bgcolor="#FF0000">

<td style="text-align:center">Cell Total </td>

<td><strong>= </strong>

<input name="JumlahCell" type="text" id="JumlahCell" onKeyUp="getmax();" onFocus="this.select();"></td>

</tr>

<tr bgcolor="#FF0000">

<td style="text-align:center">Max Cells </td>

<td><strong>= </strong>

<input name="maxcells" type="text" id="maxcells" disabled="disabled" style="text-align:center"></td>

</tr>

<tr>

<td> <input type="submit" name="Generate" value="G e n e r a t e"></td>

<td><input type="reset" name="Reset" value="R e s e t"></td>

</tr>

</table>

</div>

</form>

</body>

</html>

* **Source Code tabelcell.php**

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>Hasil Generate Tabel Cell</title>

</head>

<body>

<div align="center">

<?php

$rows = 1;

$columns = 1;

$cells = 1;

?>

<?php $rows = (int) $\_POST["JumlahRow"]; ?>

<?php $columns = (int) $\_POST["JumlahColum"]; ?>

<?php $cells = (int) $\_POST["JumlahCell"]; ?>

<strong> Anda memilih </strong> <?php echo $rows; ?> <em>rows,</em><br />

<strong> Anda memilih</strong> <?php echo $columns; ?> <em>columns,</em><br />

<strong>Ternyata Anda Membutuhkan</strong> <?php echo $cells; ?> <em>cells,</em><br />

<br />

<br />

<?php

$width = $columns \* 75;

echo "<table width=".$width." border=1>";

$rw = 0;

$cel = 1;

while ($rw < $rows && $cel <= $cells) {

echo "<tr>";

$cl = 0;

while ($cl < $columns) {

if ($cel <= $cells) {

echo "<td><div align=center>".$cel."</div></td>";

$cel++;

}

$cl++;

}

echo "</tr>";

$rw++;

}

echo "</table>";

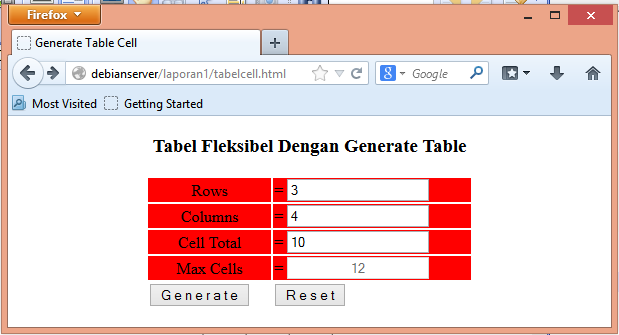
?>

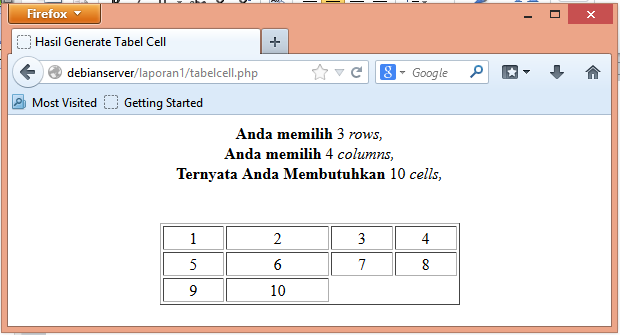
</div>

</body>

</html>

* **Screenshot**





1. **Kesimpulan**
2. Struktur dasar dokumen PHP antara lain tag PHP, komentar dan tipe data. Tag PHP terdiri dari 4 macam dengan penggunaan yang berbeda. Sedangkan komentar pada PHP ada 3. Dan tipe data pada php ada 8 mcam.
3. Untuk membuat dokumen PHP yang baik dan benar diperukan pemahaman yang cukup mengenai struktur PHP. Serta penggunaan fungsi-fungsi yang ada pada PHP.
4. Seperti halnya bahasa pemograman lain, PHP memiliki elemen-elemen dasar untuk menampilkan informasi seperti variabel, tipe data, casting serta berbagai operator yang terdapat didalamnya seperti operator logika, aritmatika, kondisi, looping.

**DAFTAR PUSTAKA**

Anonim.2008.*KomentarpadaPHP*.(http://learningisfun.wordpress.com/2008/04/23/komentar-pada-php/ diakses tanggal 9 Oktober 2013).

Anonim.2012.*Tipe Data dalam PHP*.( http://ajarwebs.blogspot.com/2012/11/tipe-data-dalam-php.html diakses tanggal 9 Oktober 2013).

Fujiwati.2012.*TagtagDasarPHP*.(http://fujiwatipemrograman.blogspot.com/2012/07/tag-tag-dasar-php.html diakses tanggal 9 Oktober 2013).

Hastomo.\_\_\_\_\_\_.P*engertiandanSejarah PHP*. (http://hastomo.net/php/pengertian-dan-sejarah-php/ diakses tanggal 9 Oktober 2013).

Prasetya,Didik Dwi.2010. *Dasar-Dasar PHP*. Malang: Pendidikan Teknik Informatika-Teknik Elektro-Universitas Negeri Malang.