**LAPORAN MODUL II**

**PRAKTIKUM PEMROGRAMAN BERBASIS WEB**

**DASAR-DASAR PHP**

Disusun untuk memenuhi tugas matakuliah Praktikum Pemrograman Berbasis Web yang dibimbing oleh Bapak Jauharul Fuady



**Disusun Oleh :**

**Achmad Khumaini**

**110533430635**

**UNIVERSITAS NEGERI MALANG**

**FAKULTAS TEKNIK**

**JURUSAN TEKNIK ELEKTRO**

**PRODI S1 PENDIDIKAN TEKNIK INFORMATIKA**

**OKTOPBER 2013**

**MODUL 2**

**DASAR-DASAR PHP**

1. **TUJUAN**

* Memahami struktur dasar dokumen PHP.
* Mampu membuat dokumen PHP yang baik dan benar
* Mampu memanfaatkan elemen-elemen dasar untuk mengolah dan menampilkan informasi.

1. **DASAR TEORI**

[PHP](http://www.php.net/) adalah singkatan dari "PHP: Hypertext Prepocessor". [**PHP**](http://maniacms.blogspot.com/2012/01/pengertian-php.html) adalah bahasa pemrograman server side yang sudah banyak digunakan pada saat ini, terutama untuk pembuatan website dinamis. Untuk hal-hal tertentu dalam pembuatan web, bahasa pemrograman PHP memang diperlukan, misalnya saja untuk memproses data yang dikirimkan oleh pengunjung web.

**Web Server** yang mendukung php dapat ditemukan dimana - mana dari mulai IIS sampai dengan apache, dengan configurasi yang relatif mudah. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan. Dalam sisi pemahamanan, php adalah bahasa scripting yang paling mudah karena referensi yang banyak.

Dalam PHP tipe data tidak dideklarasikan oleh programmer akan tetapi ditentukan secara otomatis oleh Intepreter PHP. Jadi anda tidak perlu deklarasikan tipe data tersebut jika ingin memberikan nilai pada suatu variabel. Anda dapat menuliskan **fungsi gettype()**untuk mengetahui type data dari nilai yang anda berikan.

           <?php  
                $a = 0.3;  
                $b = -3;  
                $c = "Hallo Berooo";  
                echo "$a adalah type data ". gettype($a)."<br />";  
                echo "$b adalah type data ". gettype($b)."<br />";  
                echo "$c adalah type data ". gettype($c)."<br />";  
           ?>

Ada 8 macam tipe data pada PHP, diantaranya :

**1.** **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](http://viruspintar.blogspot.com/) akan mengkonversikannya ke dalam tipe data Floating Point.

Integer merupakan Tipe Data yang berisikan **bilangan bulat** negatif dan positif atau**tanpa ada nilai pecahan**.

Tipe ini juga dapat dinyatakan dalam bentuk oktal (berbasis 8), desimal (berbasis 10), heksadesimal (berbasis 16). Seperti dalam contoh berikut :

            $oktal = 031;

            $desimal = 25;

            $heksadesimal = 0×10;

**2.** **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.

Contoh penulisannya :

<?php

          $a = 1.234;

         $b = 1.2e3;

        $c = 2E-6;

        $d = 1.2E8;

?>

**3.**   **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 **false**pada variabel tersebut. Penulisannya tidak dipengaruhi penggunaan huruf besar atau kecil.

Contoh penulisannya :

     <?php

              $a = TRUE; // bernilai 1

             $b = true; // bernilai 1

             $c = FALSE; // bernilai 0

            $d = false; // bernilai 0

      ?>

Setiap Tipe Data dalam PHP mempunyai Boolean untuk melakukan **Seleksi Kondisi**. dan Tipe Data tersebut akan memberikan nilai **true** atau **false** pada seleksi kondisi tersebut.

·Pada Variabel yang bertipe Data integer=0 (nol) mempunyai boolean false, selain itu true.

·Pada Variabel yang bertipe Data float=0.0 (nol) mempunyai boolean false, selain itu true.

·Pada Tipe Data string yang kosong, dan string="0" mempunyai boolean false, selain itu true.

·Array yang tidak mempunyai element mempunyai boolean false, selain itu true.

·Objek yang tidak mempunyai member mempunyai boolean false, selain itu true.

·Tipe NULL dan Variabel yang belum terisi mempunyai boolean false.

**4.**   **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 (” “)

|  |  |
| --- | --- |
| Contoh penulisan tipe data string : | |
|  | <?php           $kata1 = 'string diapit petik tunggal';           $kata2 = "string diapit petik ganda";  ?> |

Perbedaan antara petik ganda dan petik tunggal adalah, ketika kita menggunakan petik tunggal maka variabel tidak akan terbaca nilainya. Beda halnya dengan penggunaan petik genap yang akan membaca nilai dari suatu variabel.

Lebih spesifiknya :

<?php

           $kata = "Haloo Berooo";

           echo 'nama saya adalah $kata';

                     //jika menggunakan tanda petik tunggal browser

                   //akan menampilkan $kata

          echo "nama saya adalah $kata";

                    //jika menggunakan tanda petik tunggal browser

                   //akan menampilkan Haloo Beroo

?>

Jika string yang tertutup dalam tanda kutip ganda (” “), PHP akan menafsirkan escape sequence lebih untuk karakter khusus. Maka penulisan untuk karakter khusus tersebut harus ditambahkan slash didepan karakter tersebut.

**5.**   **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.

Agar lebih mudah dapat menggunakan variabel dengan nilai string :

class Nama {

               var $jenis = “Beroo Haloo”;

                function ganti\_nama($jenis) {

                 $this->jenis = $jenis;

                }

}

$nama = new Nama;

echo $nama->jenis;

$nama->ganti\_nama(“Haloo Beroo”);

echo “<br>”.$nama->jenis;

Hasilnya :

Beroo Haloo

Haloo Beroo

**6.**   **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

<?php

          $hari[0]="Senin";

          $hari[1]="Selasa";

          $hari[2]="Rabu";

          $hari[3]="Kamis";

          $hari[4]="Jum'at";

          $hari[5]="Sabtu";

          $hari[6]="Minggu";

                     //menampilkan tanpa perulangan.

           echo "sekarang adalah hari $hari[3] <br />" ;

                   //menampilkan menggunakan perulangan

                 //perulangan akan dijelaskan pada artikel berikutnya

           for($i=0;$i<=6;$i++){

                   echo $hari[$i]."<br/>";

           }

?>

**Pada contoh numeric array diatas dapat disimpulkan bahwa pemanggilan data array tersebut menggunakan angka, angka dimulai dari 0 sampai jumlah data yang dimasukkan. $hari[0] adalah pemanggilan untuk data pertama, $hari[1] adalah pemanggilan untuk data kedua dan seterusnya.**

·            Associative Array

<?php

$hari = array("Senin"=>"Ayam","Selasa"=>"Lele","Rabu"=>"Tempe","Kamis"=>"Tahu","Jum'at"=>"Teri","Sabtu"=>"Krupuk","Minggu"=>"Puasa");

    //menampilkan tanpa perulangan.

echo "Menu hari senin adalah $hari[Senin]" ;

?>

**Pada contoh associative array diatas dapat disimpulkan bahwa pemanggilan data array tersebut menggunakan**[string](file:///F:\\web\\Tipe%20Data%20Dalam%20PHP%20_%20Kode%20PHP.htm" \l "con_string)**yang ditentukan oleh kita, dengan kata lain pemanggilan menggunakan angka kita ganti menjadi string yang kita inginkan.**

* Multidimensional Array

**Pengertian multidimension array**adalah array didalam array. Pemanggilan array ini bisa menggunakan numeric array maupun associative array. $makan[0]["pagi"] dapat diartikan sebagai menampilkan nilai array pertama dan array pagi dari variabel makan, $makan["senin"]["pagi"] dapat diartikan sebagai menampilkan nilai array senin dan array pagi dari variabel makan.

<?php

       $makan = array(

              array("pagi"=>"bubur","siang"=>"ayam","sore"=>"bebek"),

             array("pagi"=>"bubur","siang"=>"tahu","sore"=>"tempe")

        );

             //menampilkan tanpa perulangan.

         echo "Menu makan pada hari senin pagi adalah " . $makan[0]["pagi"];

        echo "<br />";

        echo "Menu makan pada hari senin pagi adalah " . $makan[1]["sore"];

?>

**7.**   **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 seperti pada

Contoh berikut:

<?php

    # Nilai awal $foo adalan 10

    $foo = 10;

     # $foo akan menjadi NULL

    $foo = NULL;

    # Sama dengan memanggil unset()

    unset($foo);

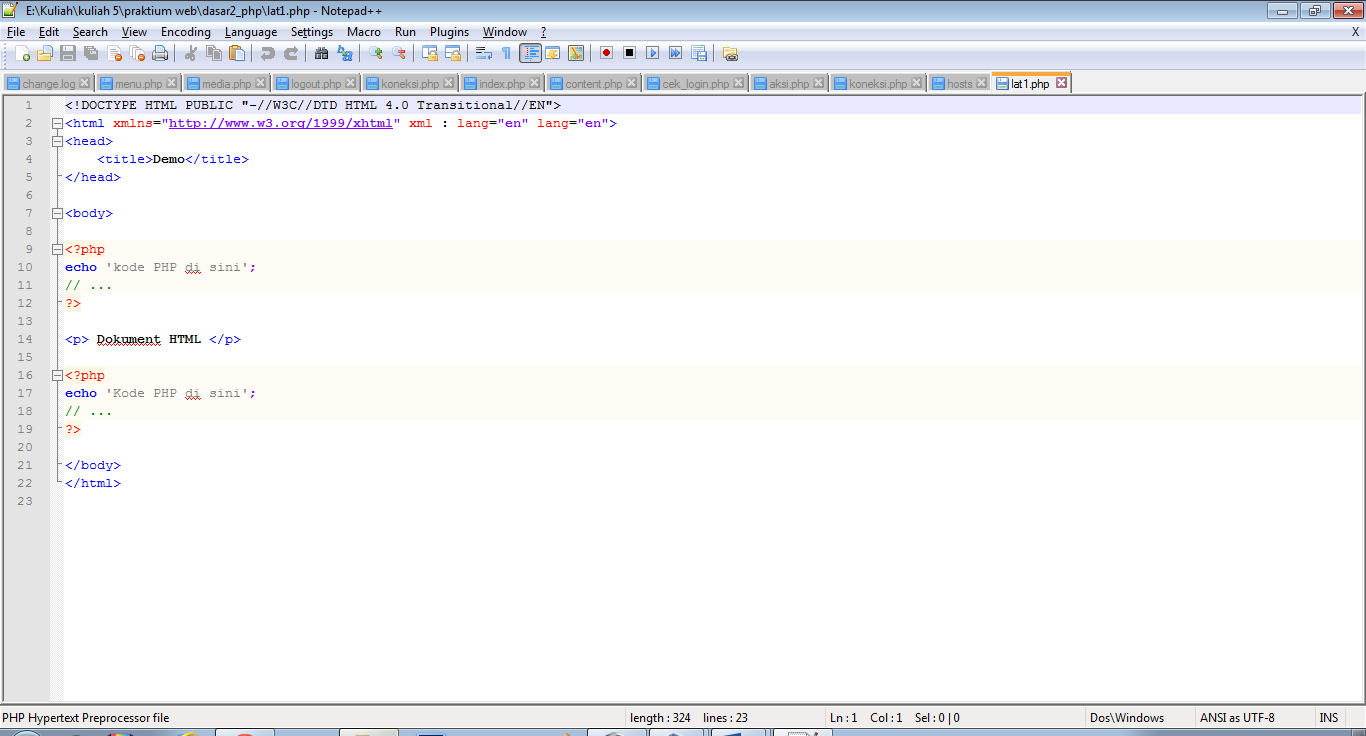
?>

**8.**   **Resources**

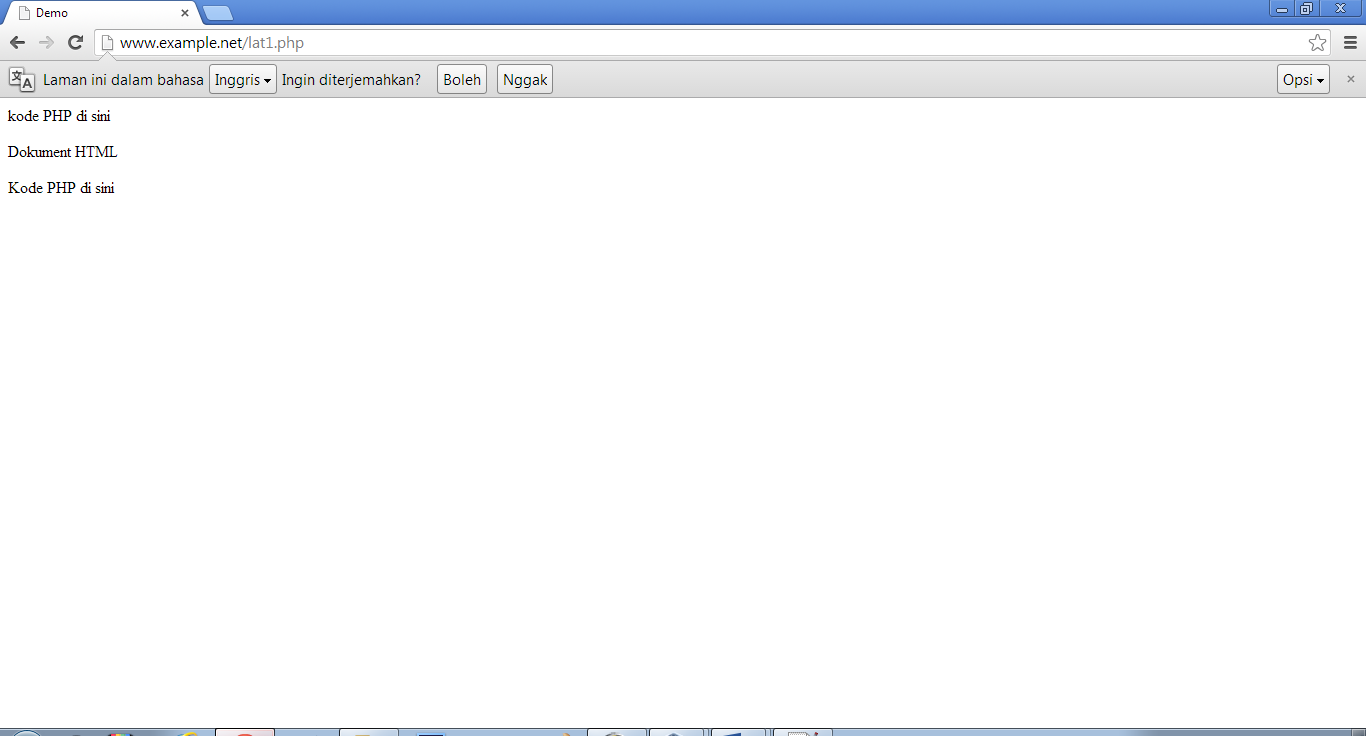
Tipe Data Spesial yang satu ini di khususkan untuk **menyimpan resource**, **sumber** atau **alamat**. Variabel tersebut hanya dapat diciptakan oleh suatu fungsi khusus yang mengembalikan nilai berupa resource seperti penggunaan fungsi **fopen**, **opendir**, **mysql\_connect**, **mysql\_query** dan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

1. **LATIHAN**

**Latihan 1**



Hasil

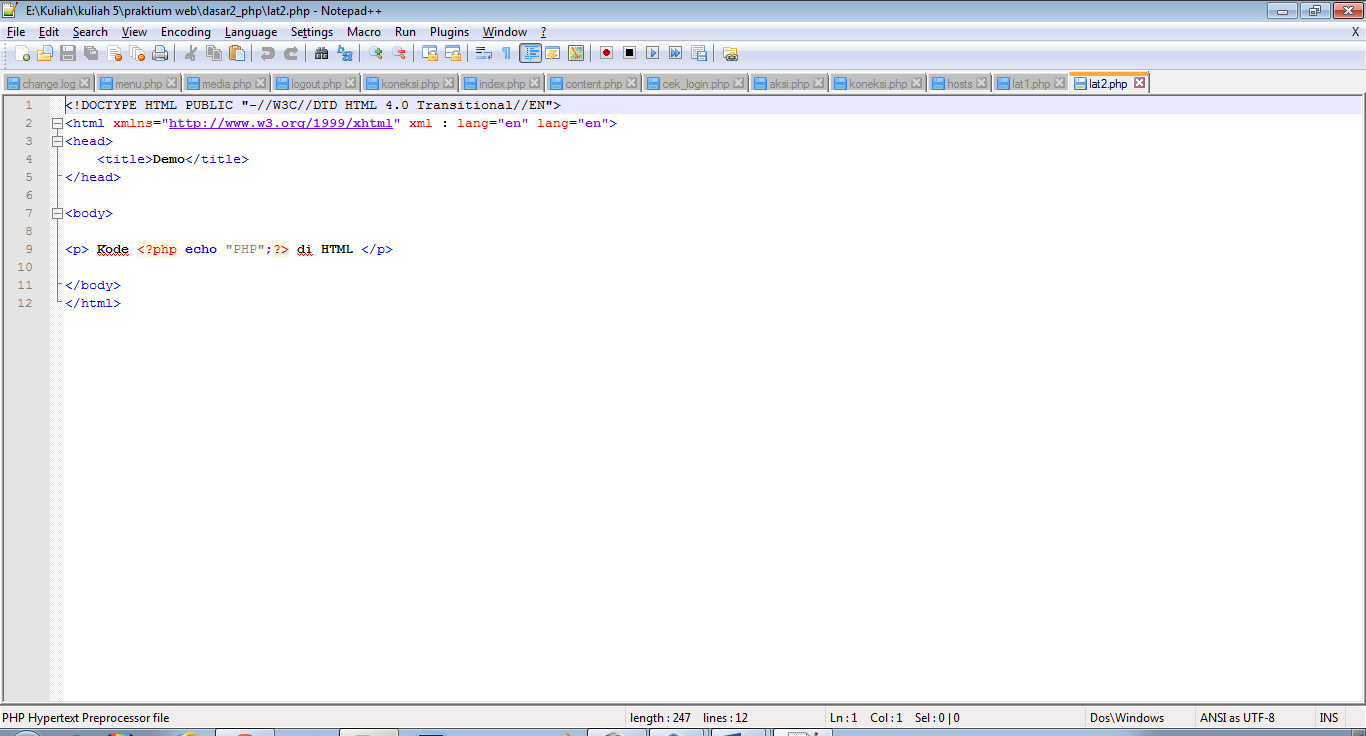


Penjelasan

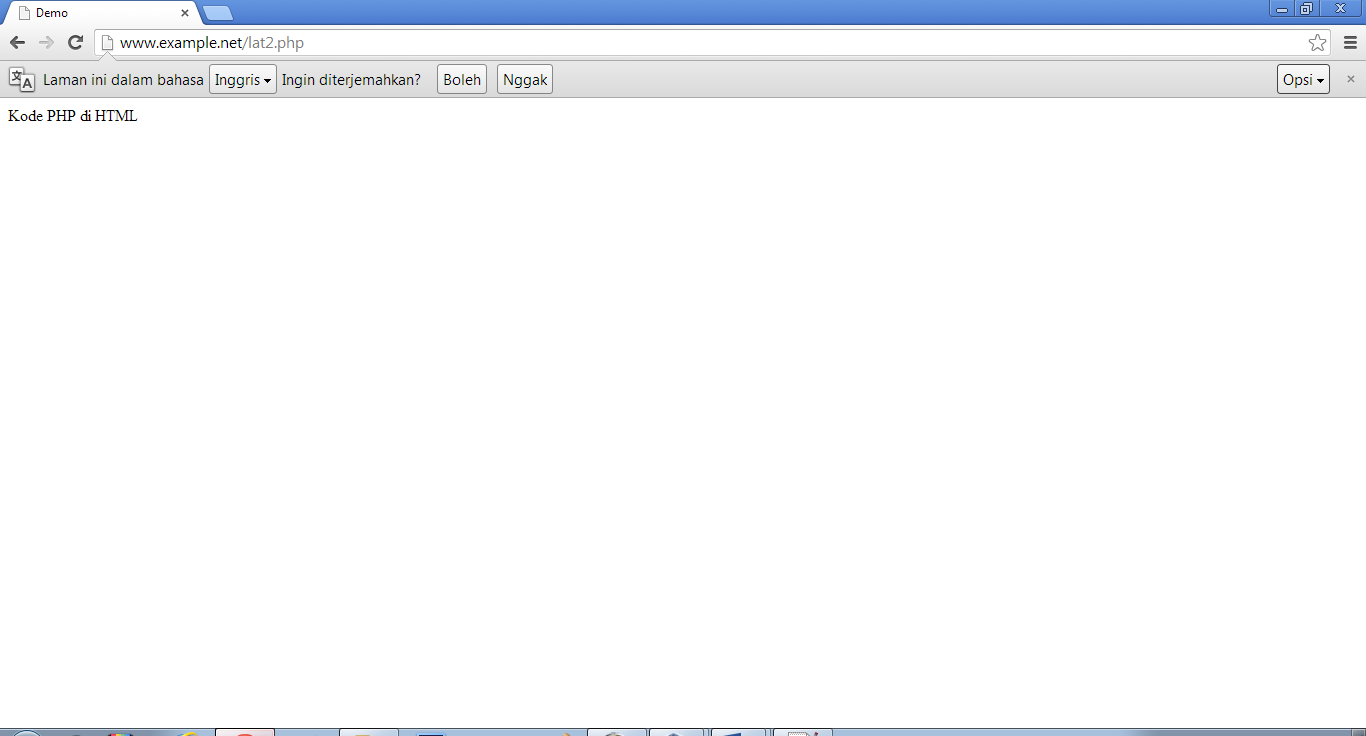
Keterangan

Script diatas digunakan untuk menampilkan tulisan pada halaman web dengan menggunakan script PHP dan HTML. Pada PHP menggunakan “echo”, sedangkan pada HTML menggunakan <p>.

latihan 2



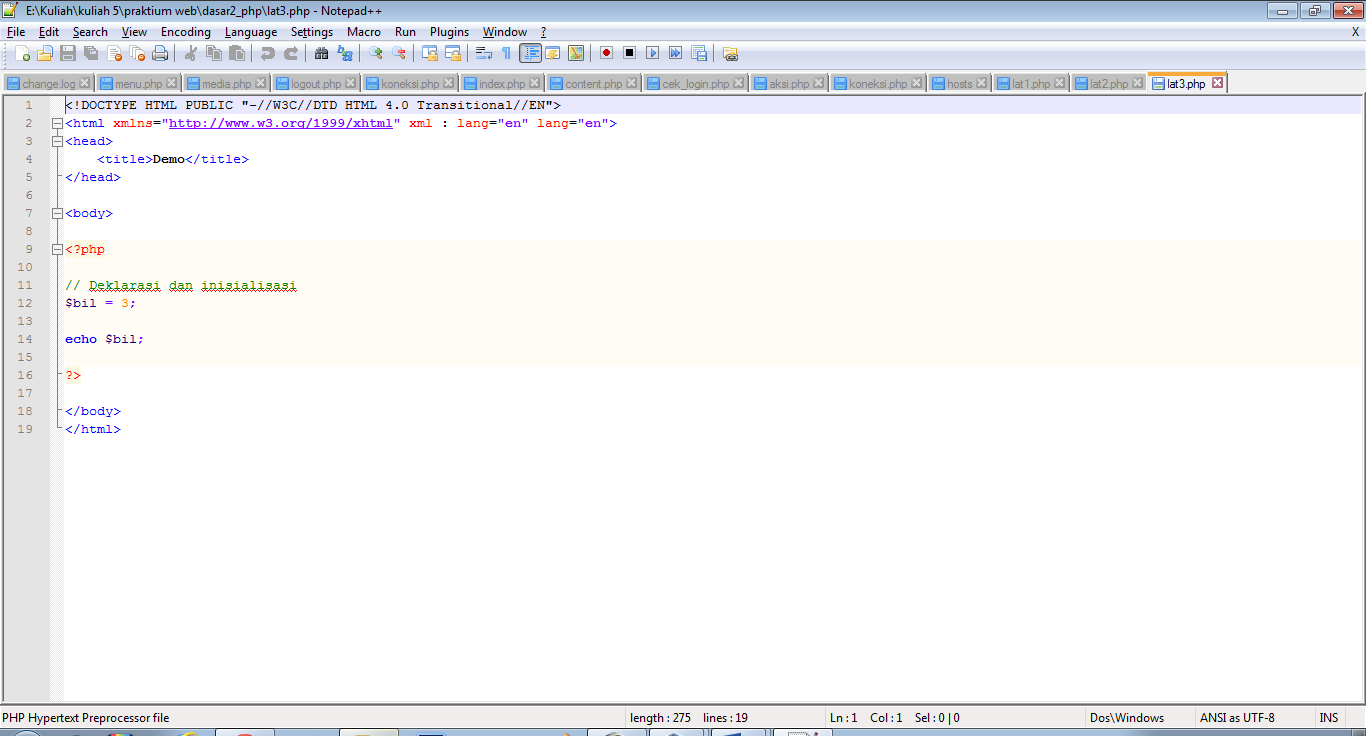
hasil



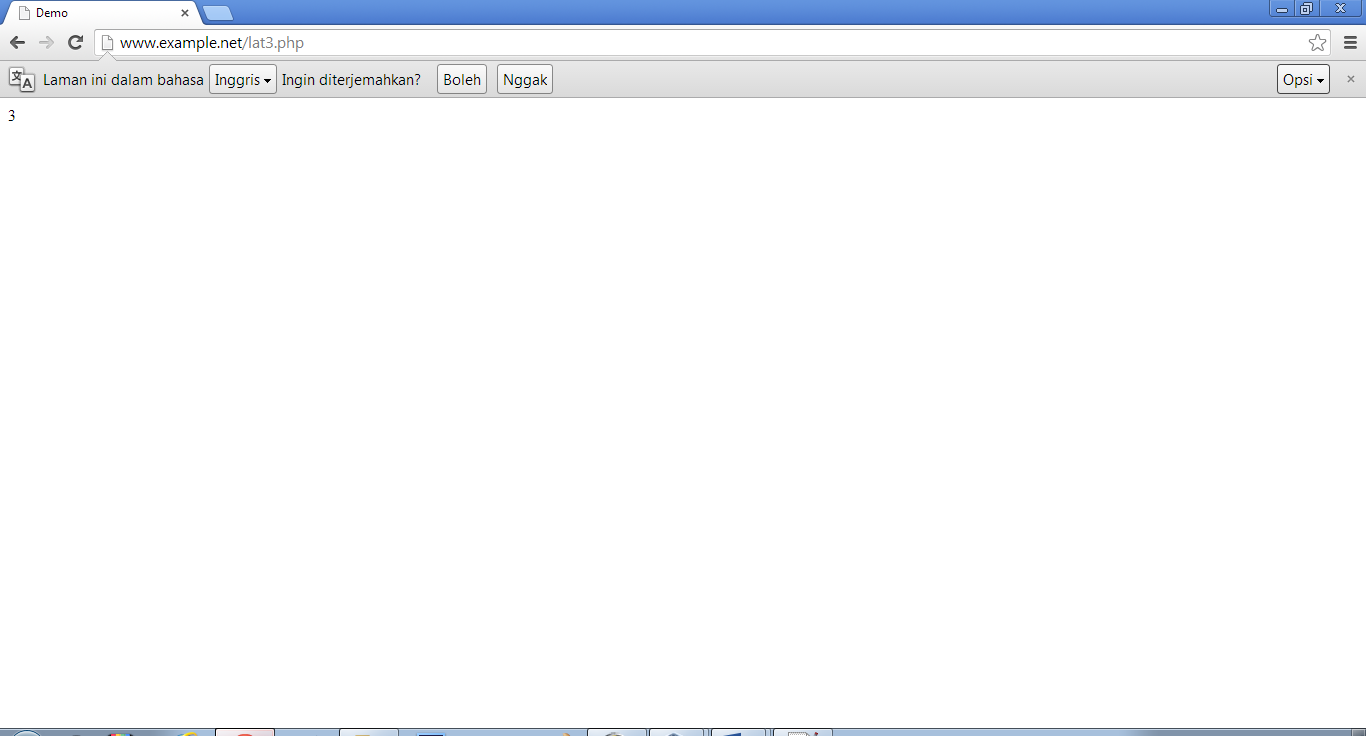
Keterangan

Script diatas hampir sama seperti pada latihan sebelumnya yaitu menampilkan tulisan pada halaman web, namun yang berbeda adalah penggunaan script PHP yang disingkat menjadi 1 baris.

Latihan 3



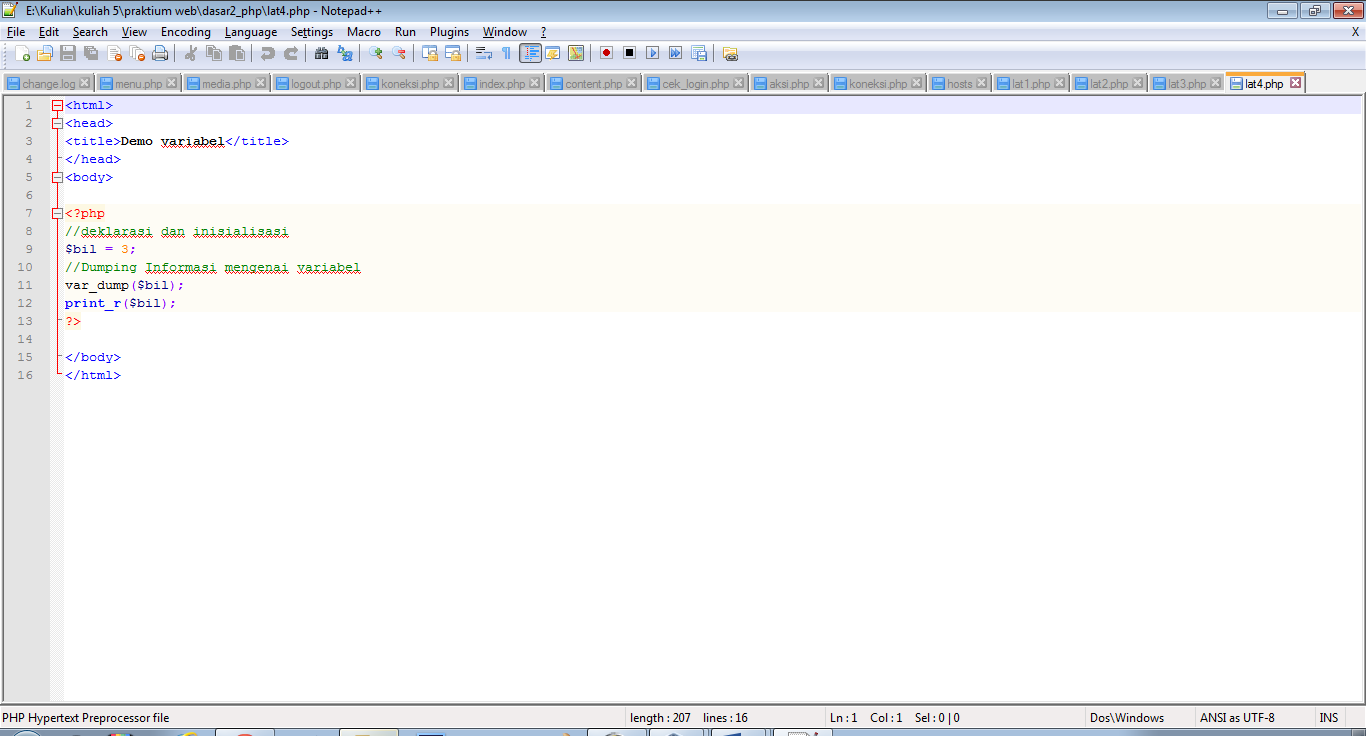
Hasil



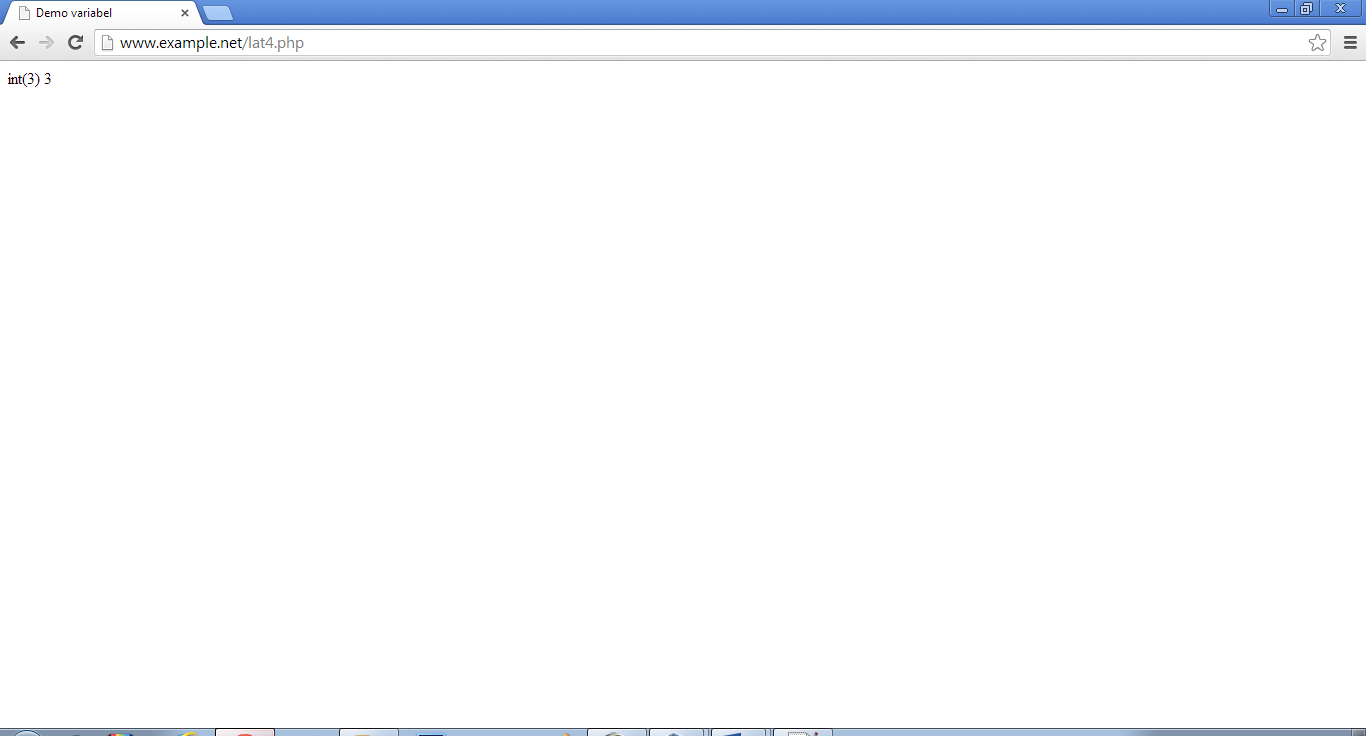
Keterangan

Script diatas menggunakan variabel “bil” yang telah diberikan nilai “3”. Kemudian variabel ini dipanggil dengan menggunakan “echo” senhingga menampilkan nilainya.

Latihan 4



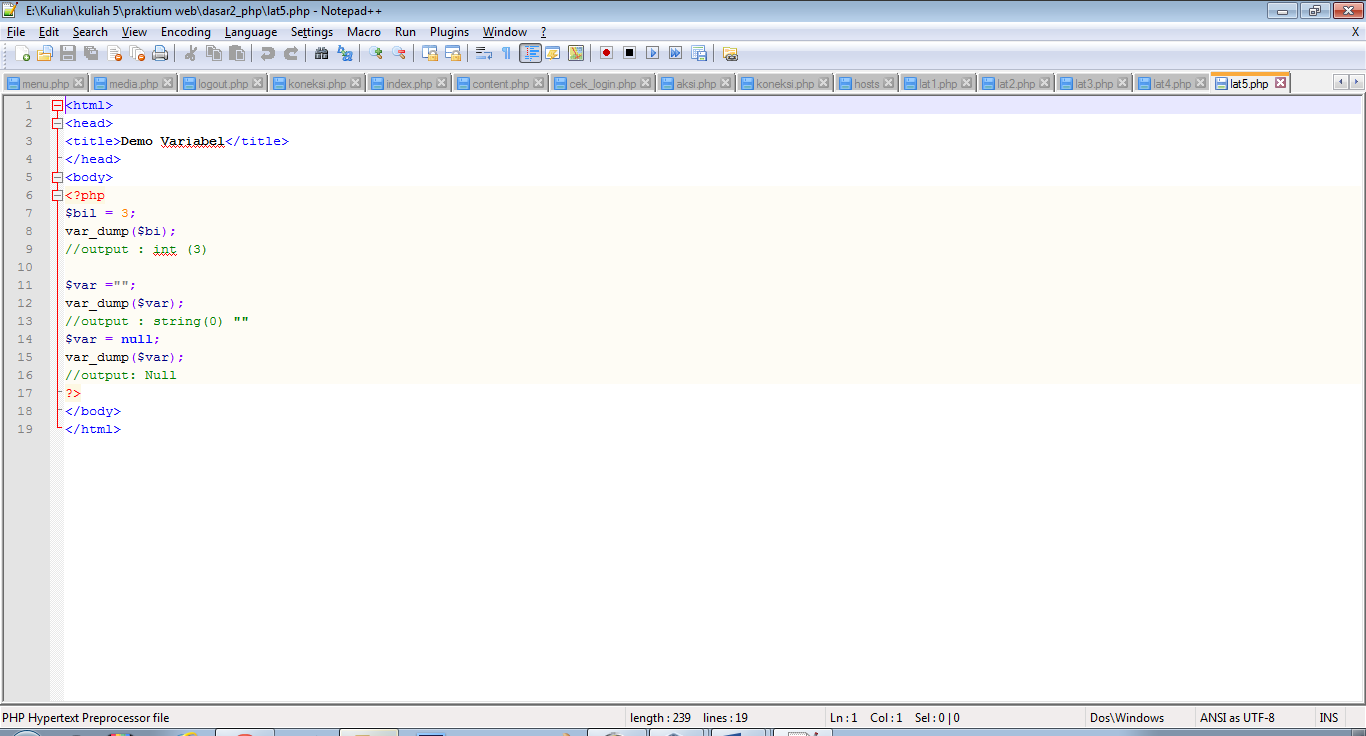
Hasil



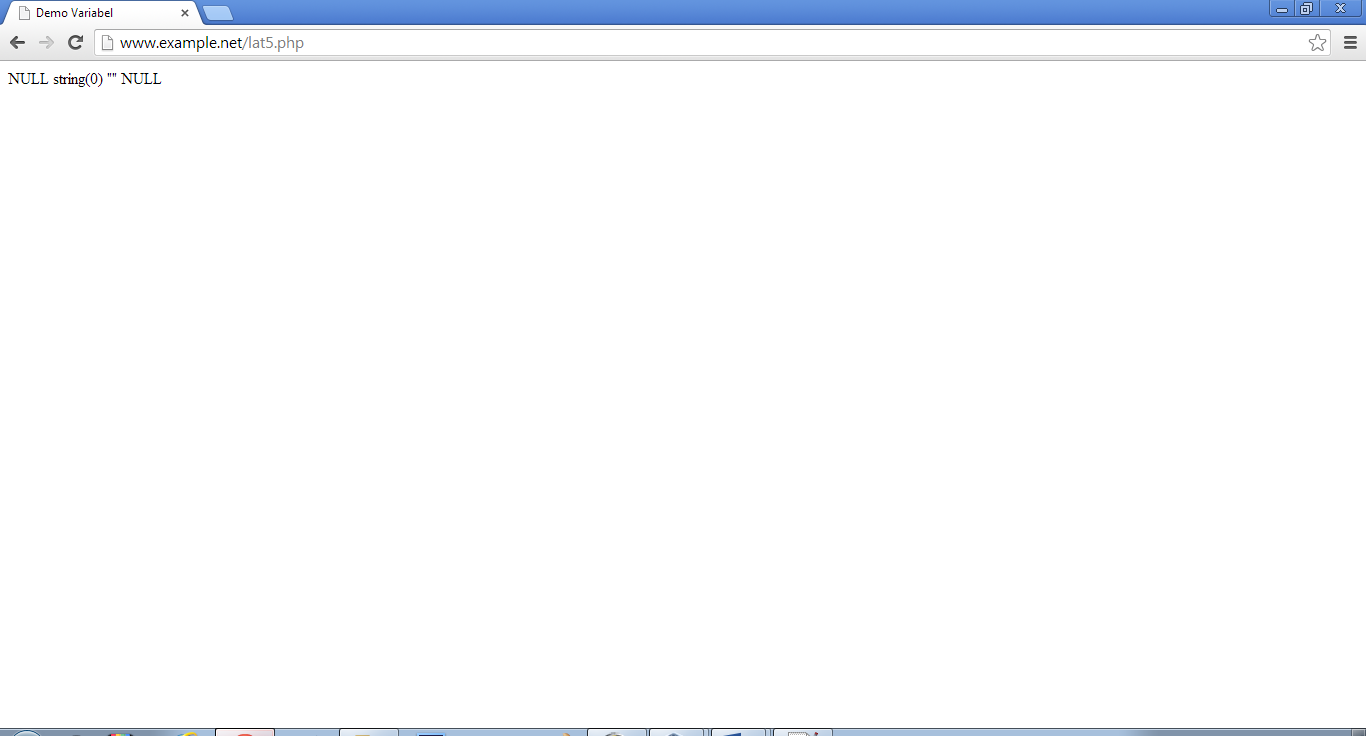
Perbedaan var\_dump dengan echo :

* Var\_dump menampilkan informasi detail dari sebuah variabel, antara lain tipe datanya, tipe sub-itemnya (apakah itu array atau objek).
* Print\_r menampilkan informasi dengan bahasa yang lebih mudah dimengerti, seperti : strings are not quoted, type information is omitted, array sizes aren't given, etc
* Echo lebih cepat 20% dari print
* Print dapat berlaku seperti sebuah fungsi artinya print memiliki return value berupa boolean jika sukses sedangkan echo tidak. Echo langsung meng-output-kan parameter yang diberikan
* Echo tidak mempunyai varian. Maksudnya echo hanya menampilkan string, jika kita menginputkan varibel berupa array atau object, echo hanya menampilkan keterangan bahwa variabel tersebut adalah array, atau object. Lain halnya dengan print, print memiliki varian print\_r() yang mampu memecah array dan menampilkan key-key (disebut juga indeks array) dan value dari array tersebut. Begitupula jika print\_r() kita beri variabel input berupa Object. Selain print\_r(), print memiliki varian printf($format, $string) untuk menampilkan string dengan format tertentu yang kita definisikan.

Latihan 5

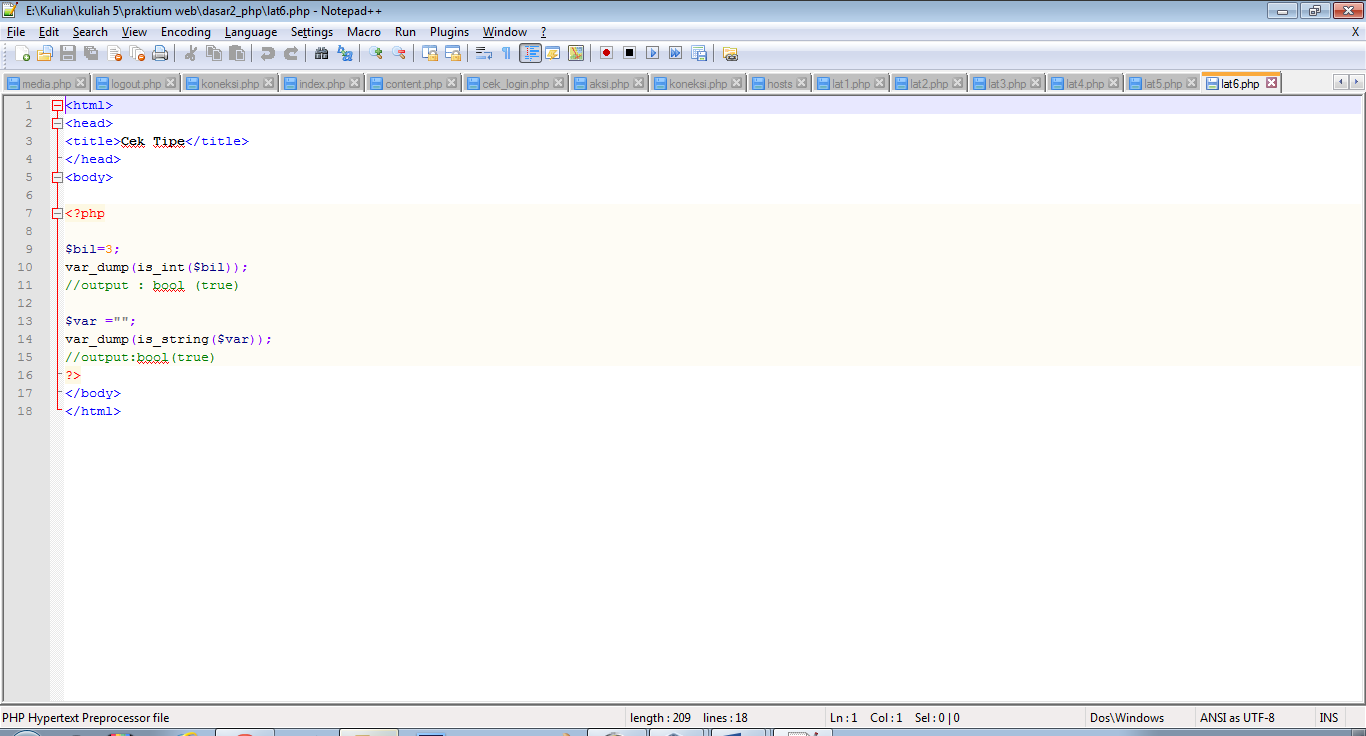


Hasil

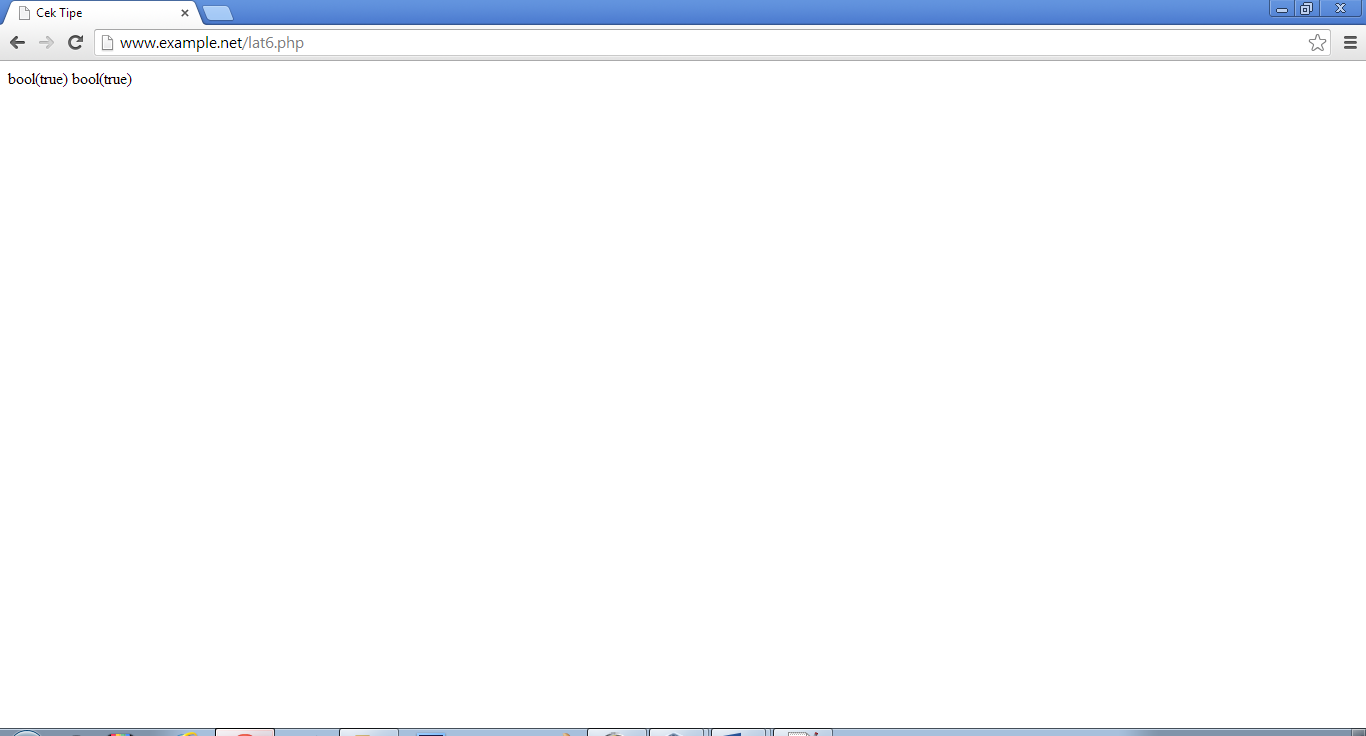


Script diatas hampir sama dengan latihan sebelumnya. namun pada latihan kali ini menggunakan *isset* yang digunakan untuk memeriksa apakah suatu variabel di-set nilainya atau tidak.

Latihan 6



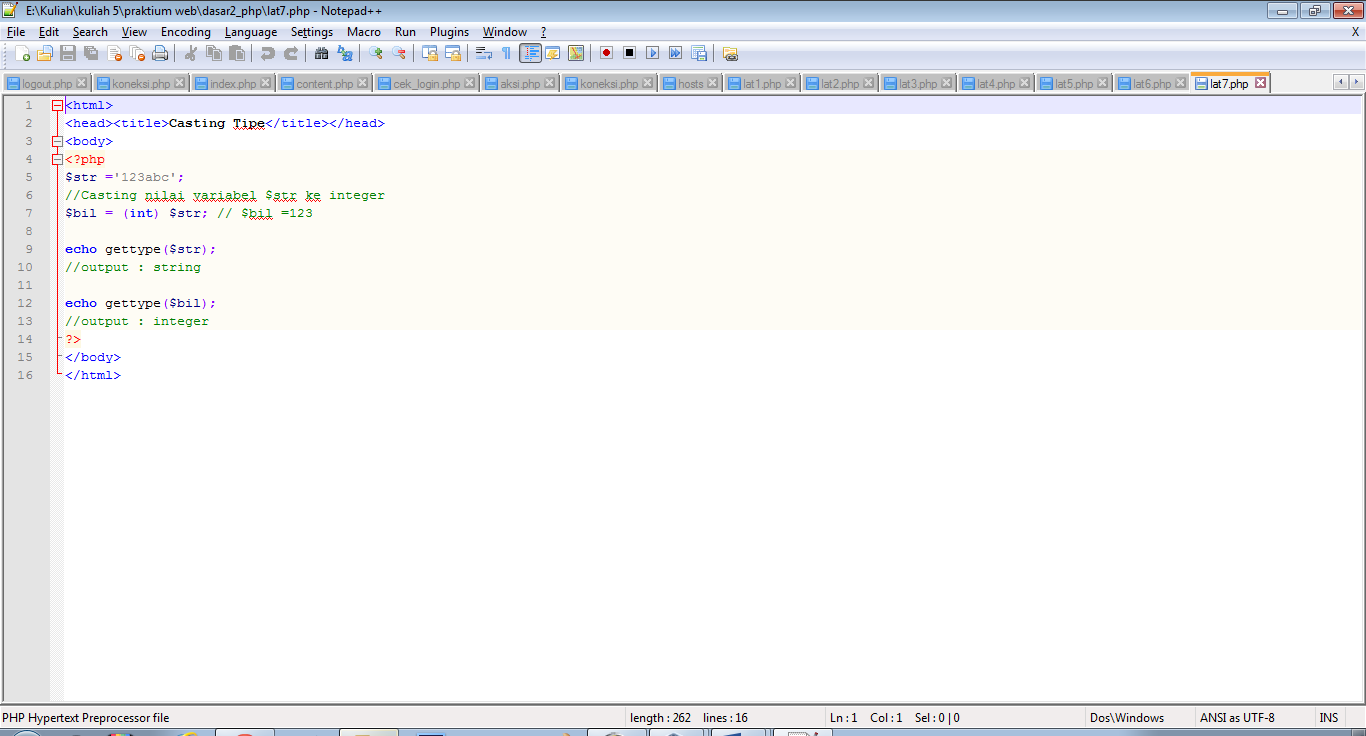
Hasil

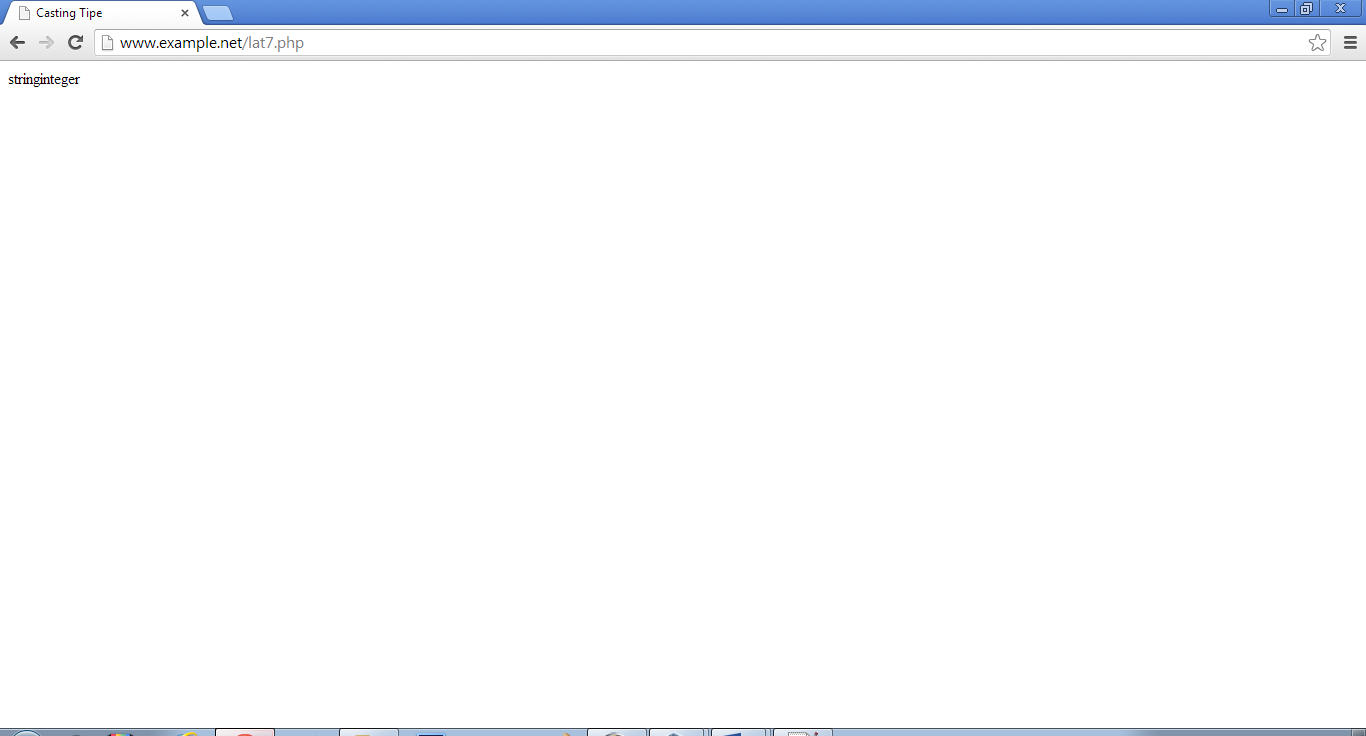


Keterangan

Pada script diatas, \_is digunakan untuk menguji tipe data pada suatu variabel. Kita dapat melihat apakah tipe data suatu variabel yang diuji sesuai dengan tipe data yang kita tuliskan.

Latihan 7



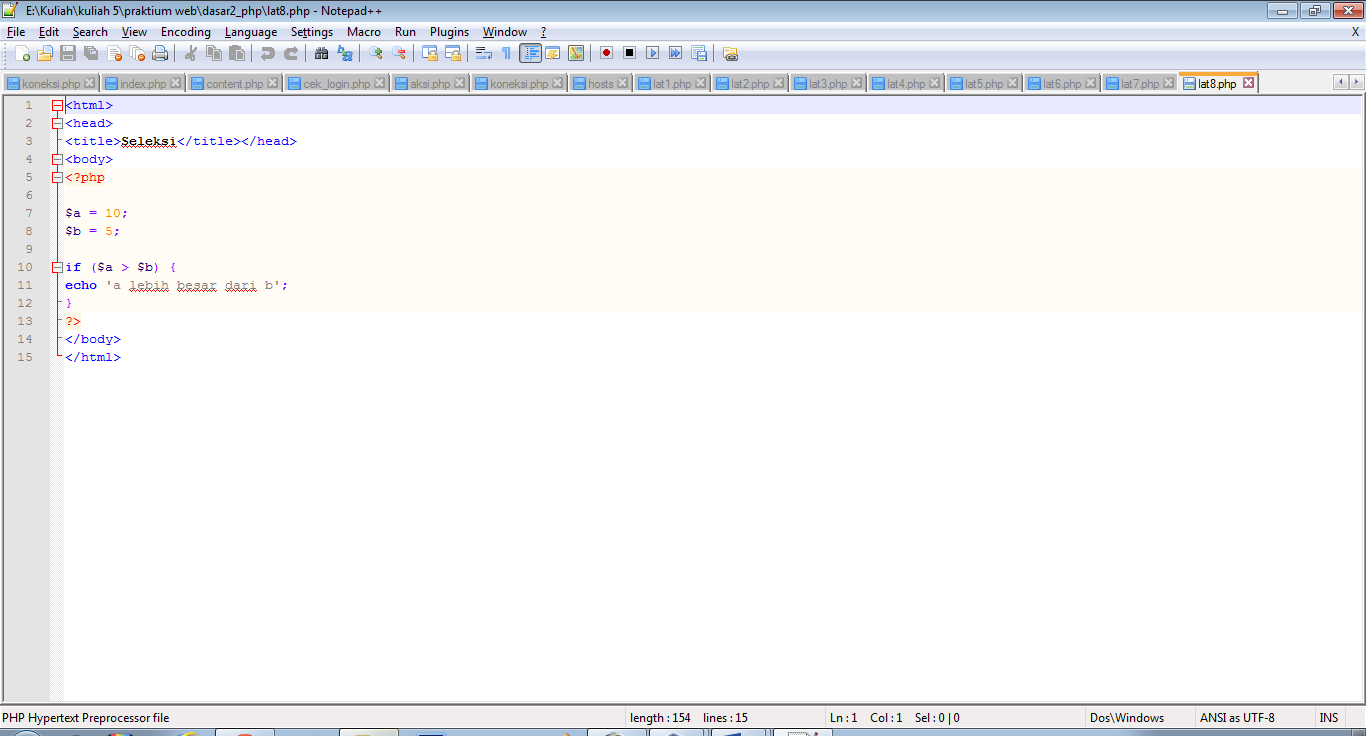


Hasil

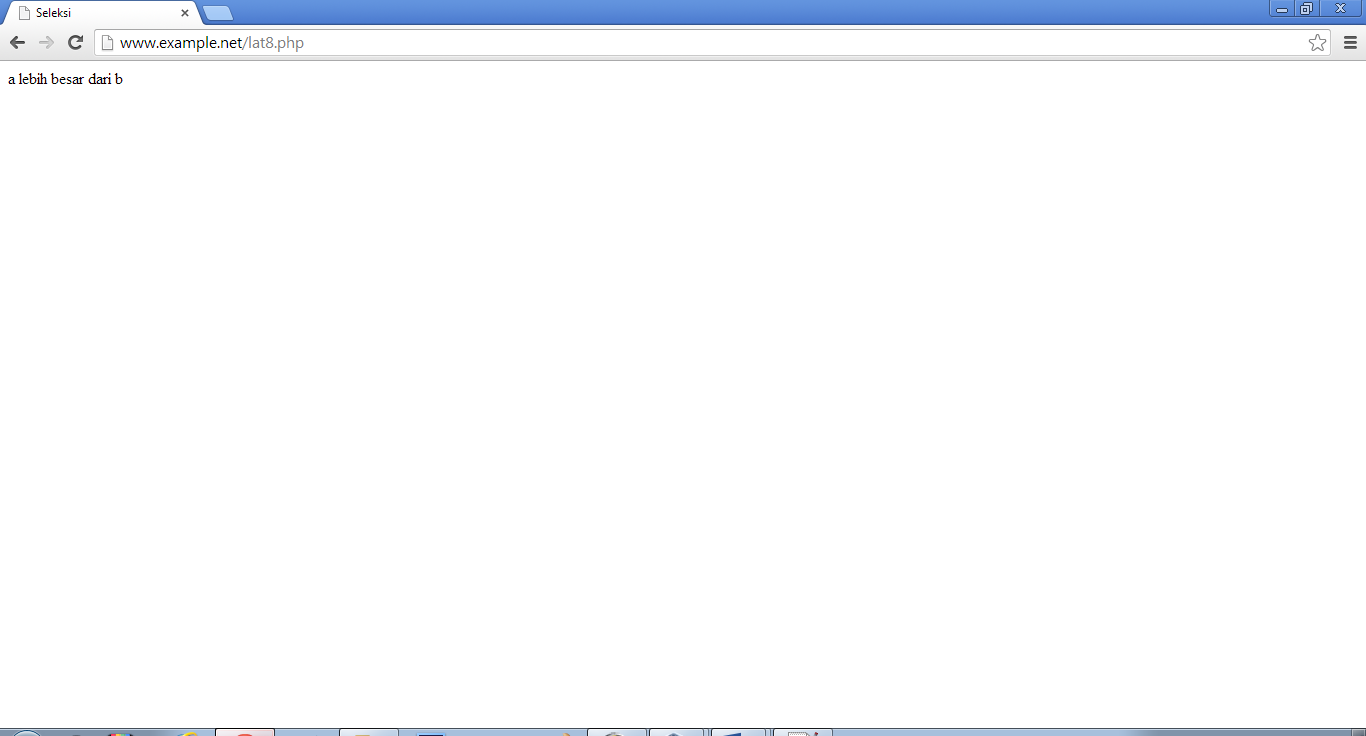
Keterangan

Script diatas hampir sama dengan latihan sebelumnya. disini kita mengujinya dengan *casting*. Kita dapat mengambil nilai pada suatu variabel dengan tipe data tertentu dan memasukkannya pada variabel dengan tipe data lain, namun nilai yang diambil hanya yang tipenya sama seperti variabel yang akan dimasukkan nilainya.

Latihan 8



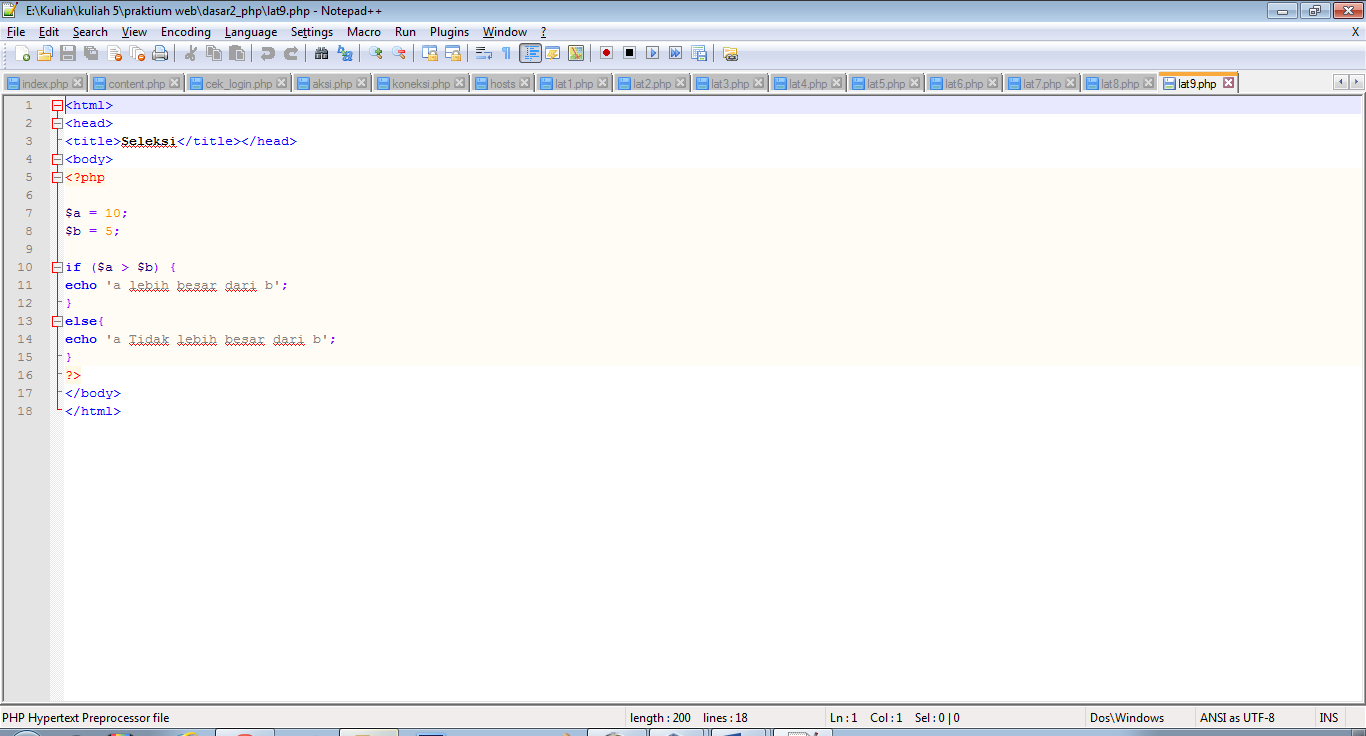
Hasil



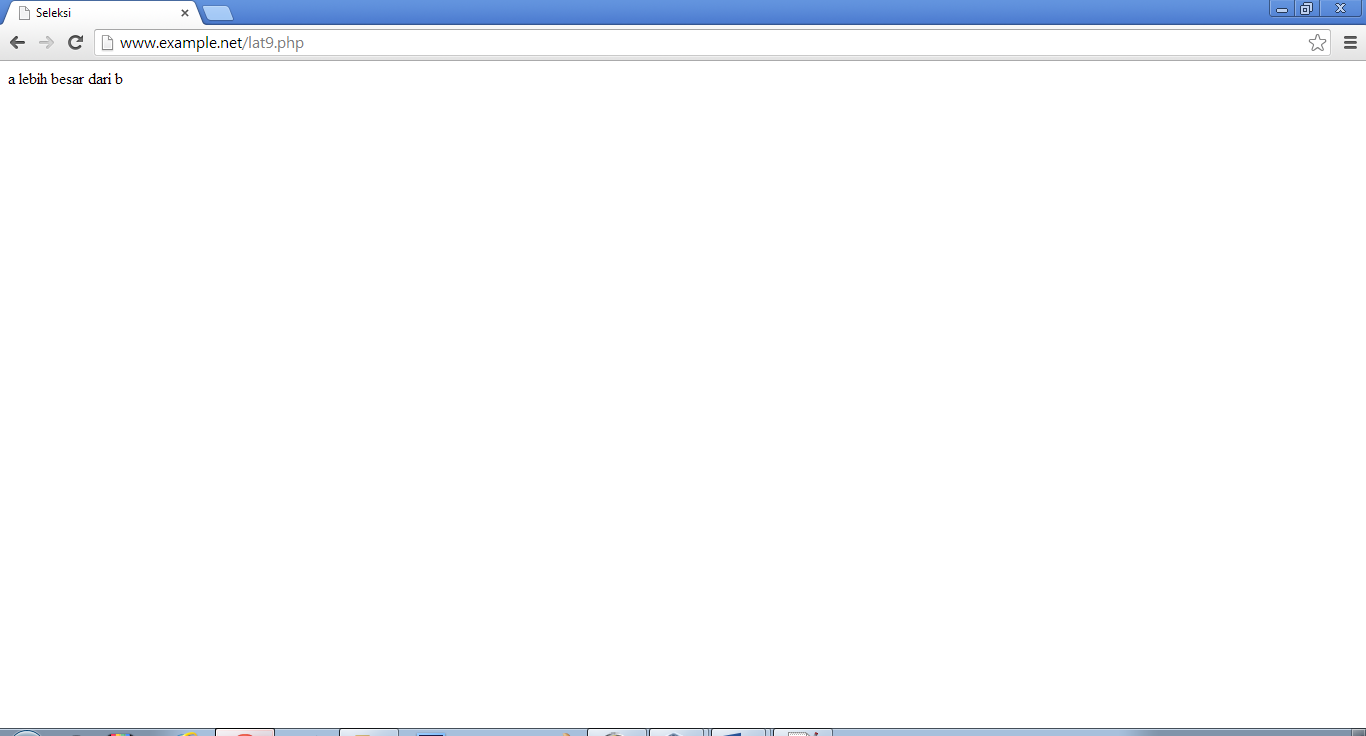
Keterangan

Script diatas menggunakan seleksi if. Dimana jika nilai pada variabel a lebih besar dari variabel b, maka kondisi di dalam if akan dijalankan.

Latihan 9



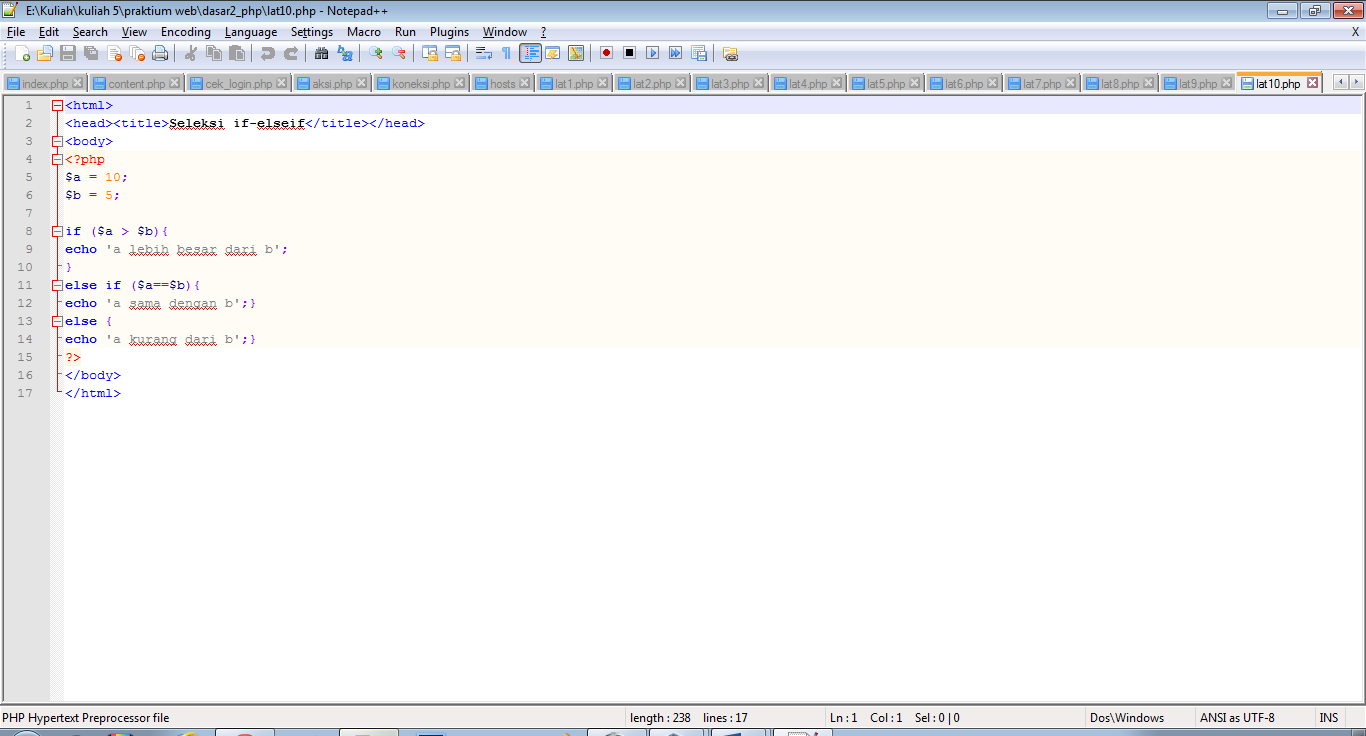
Hasil



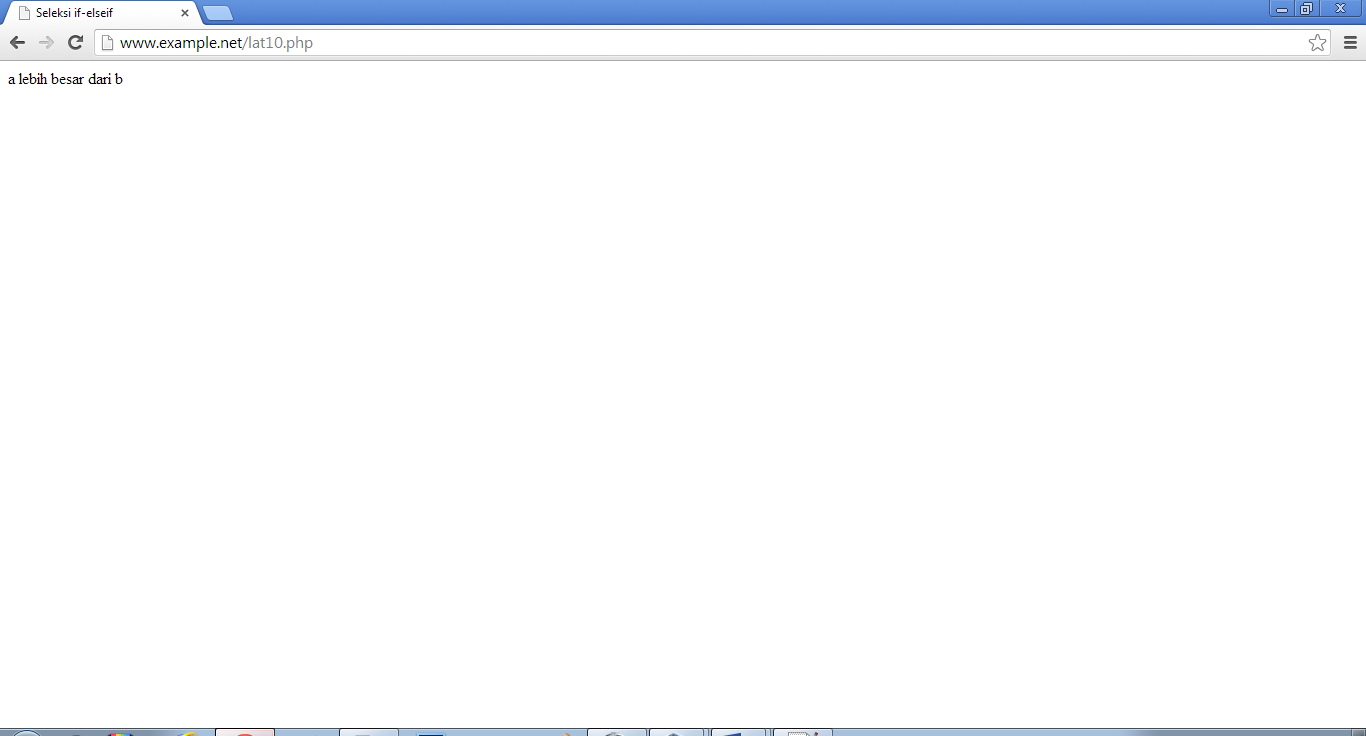
Keterangan

Script diatas hampir sama dengan latihan sebelumnya. pada script ini digunakan seleksi yang memiliki 2 kondisi. Jika kondisi 1 tidak terpenuhi maka kondisi 2 akan dijalankan

Latihan 10



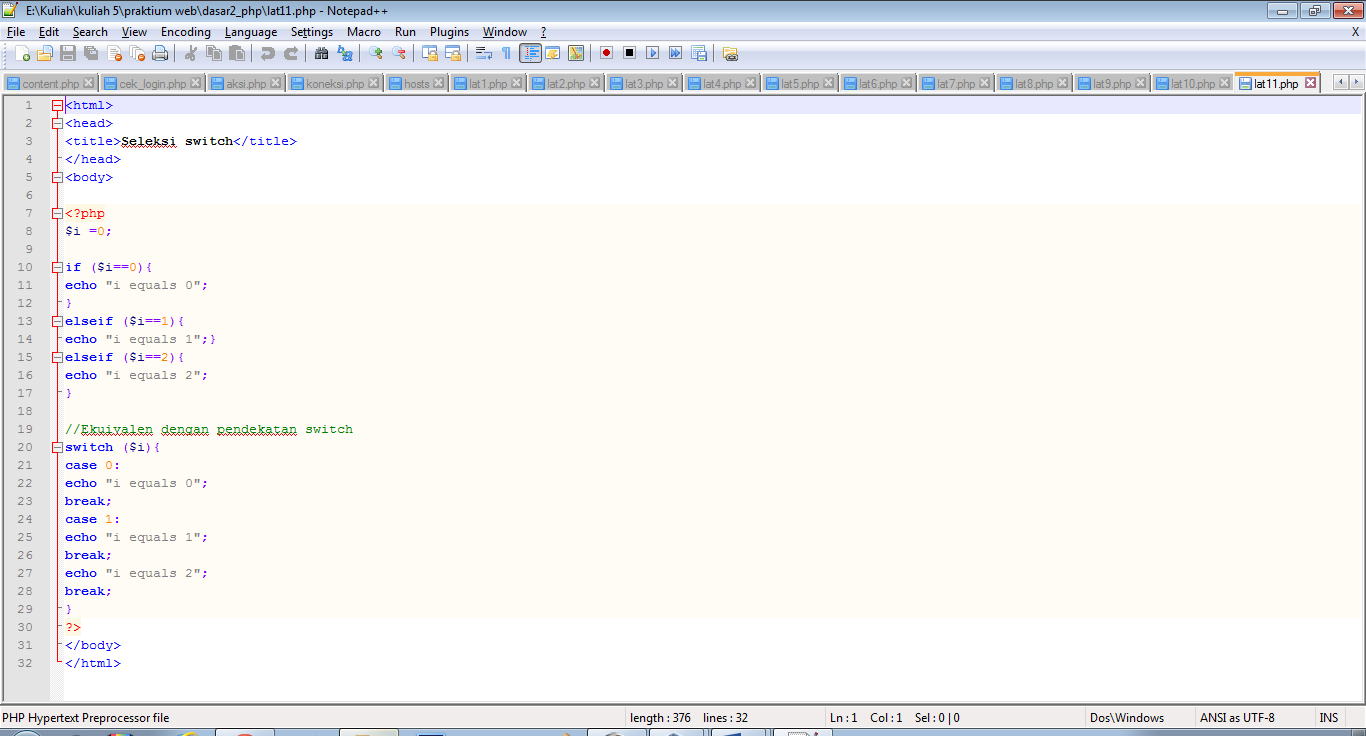
Hasil

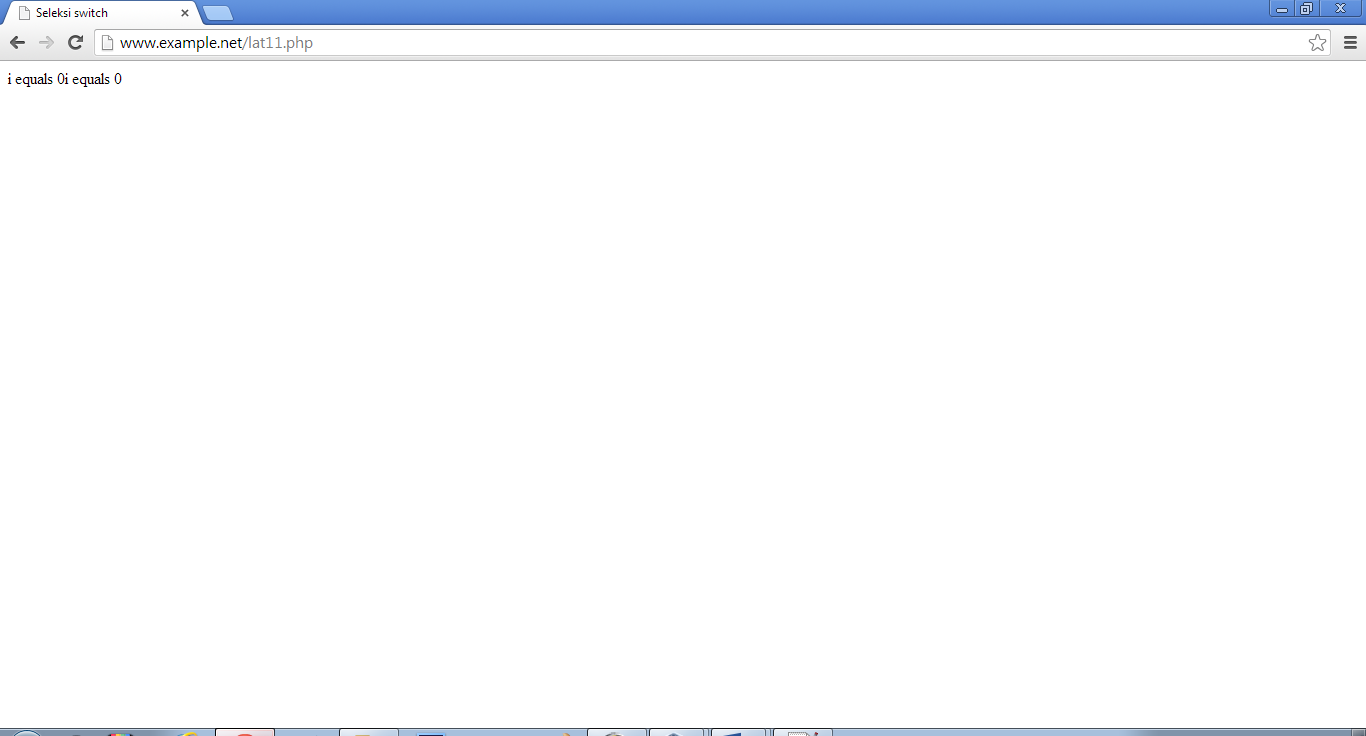


Keterangan

Script diatas hampir sama dengan latihan sebelumnya. pada script ini digunakan seleksi yang memiliki 2 kondisi. Jika kondisi 1 tidak terpenuhi maka kondisi 2 akan dijalankan

Latihan 11



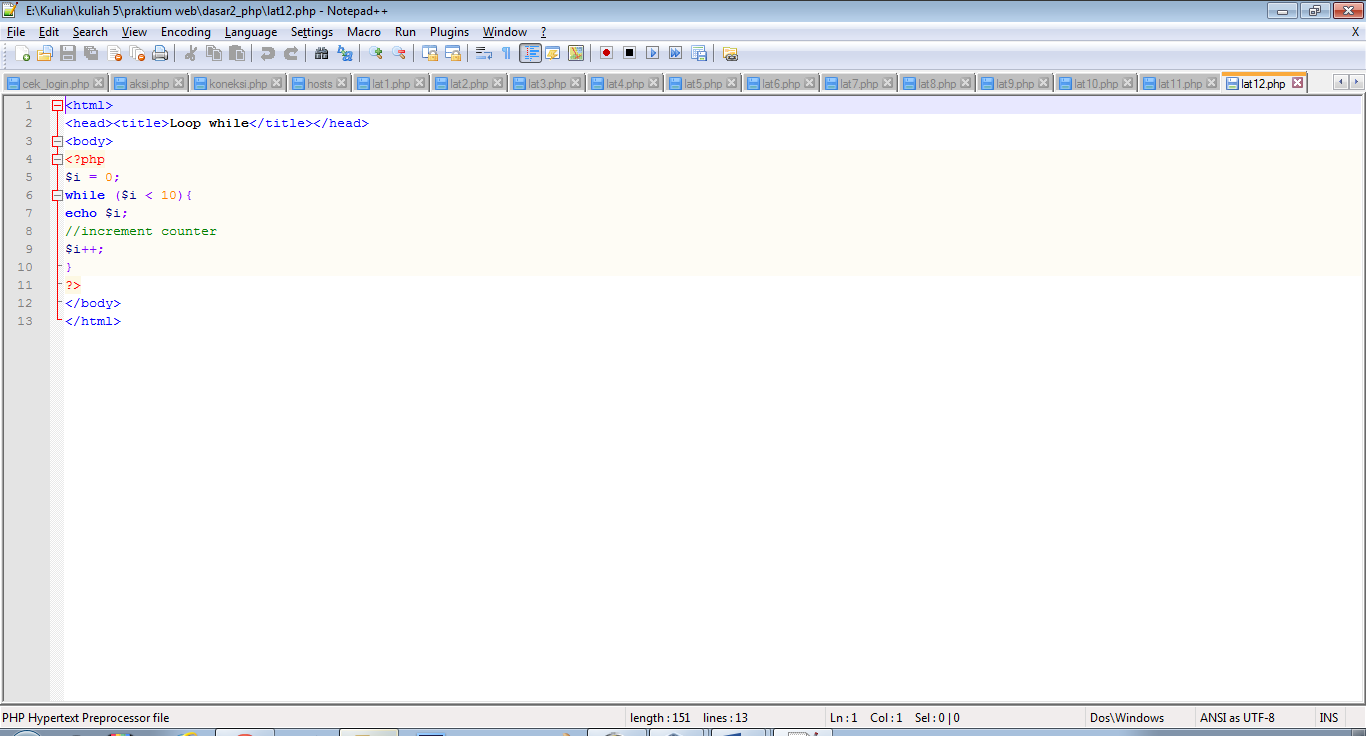


Hasil

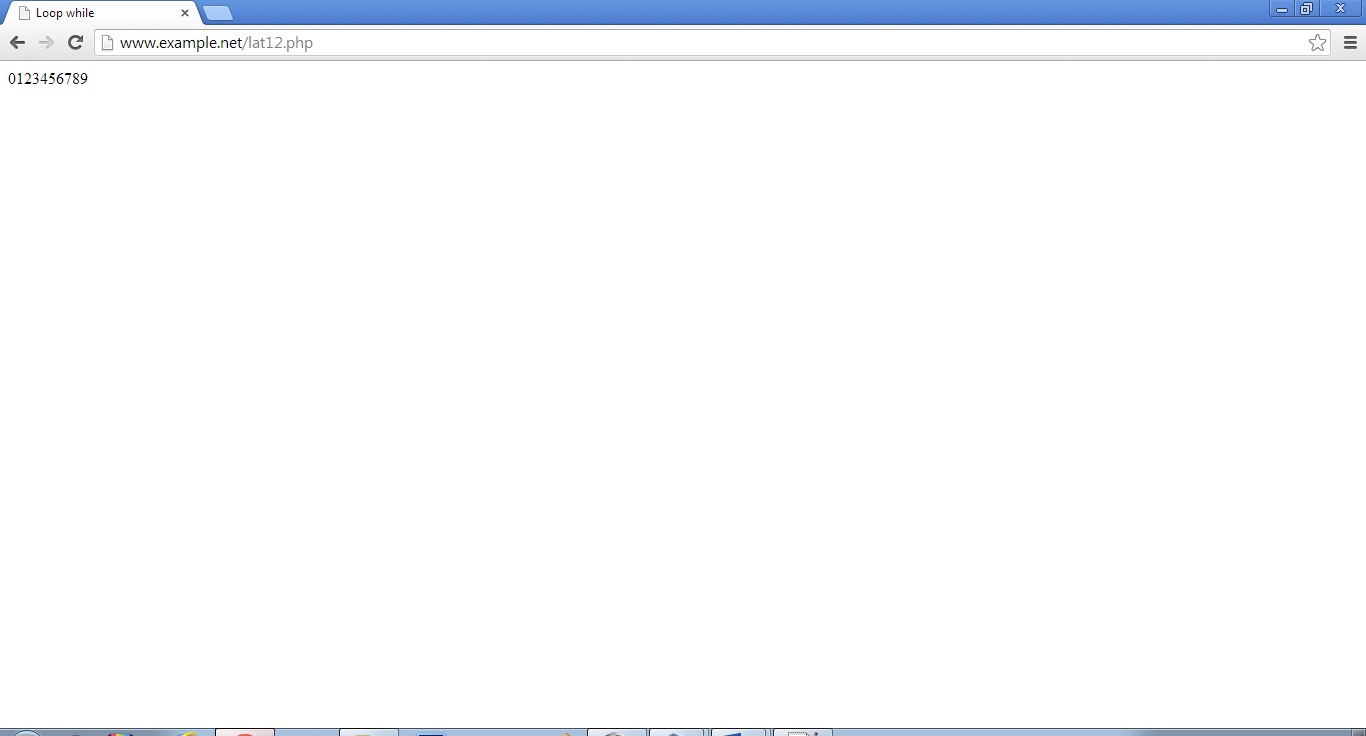
Keterangan

Script diatas menggunakan seleksi *switch*. Dimana terdapat beberapa pilihan, jika kondisi pertama dipilih, maka pernyataan pertama akan dijalankan, dan seterusnya.

Latihan 12



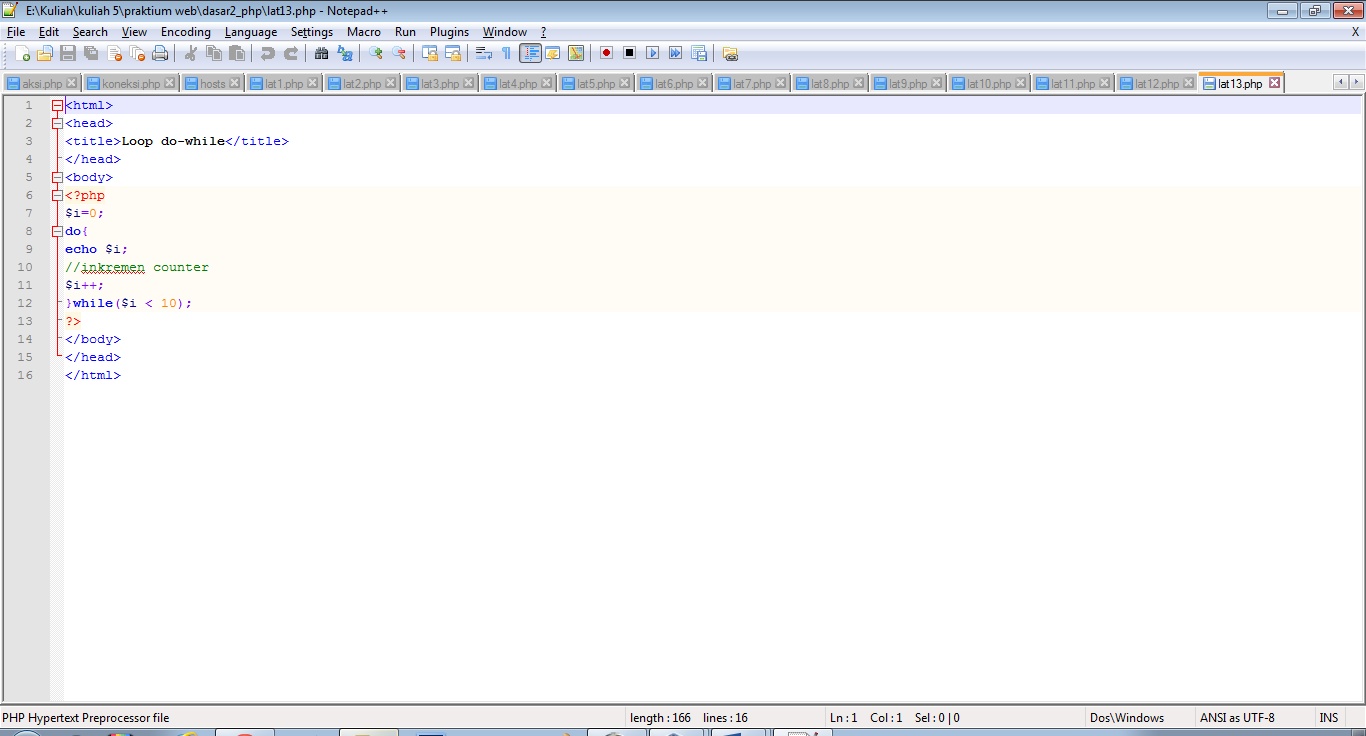
Hasil



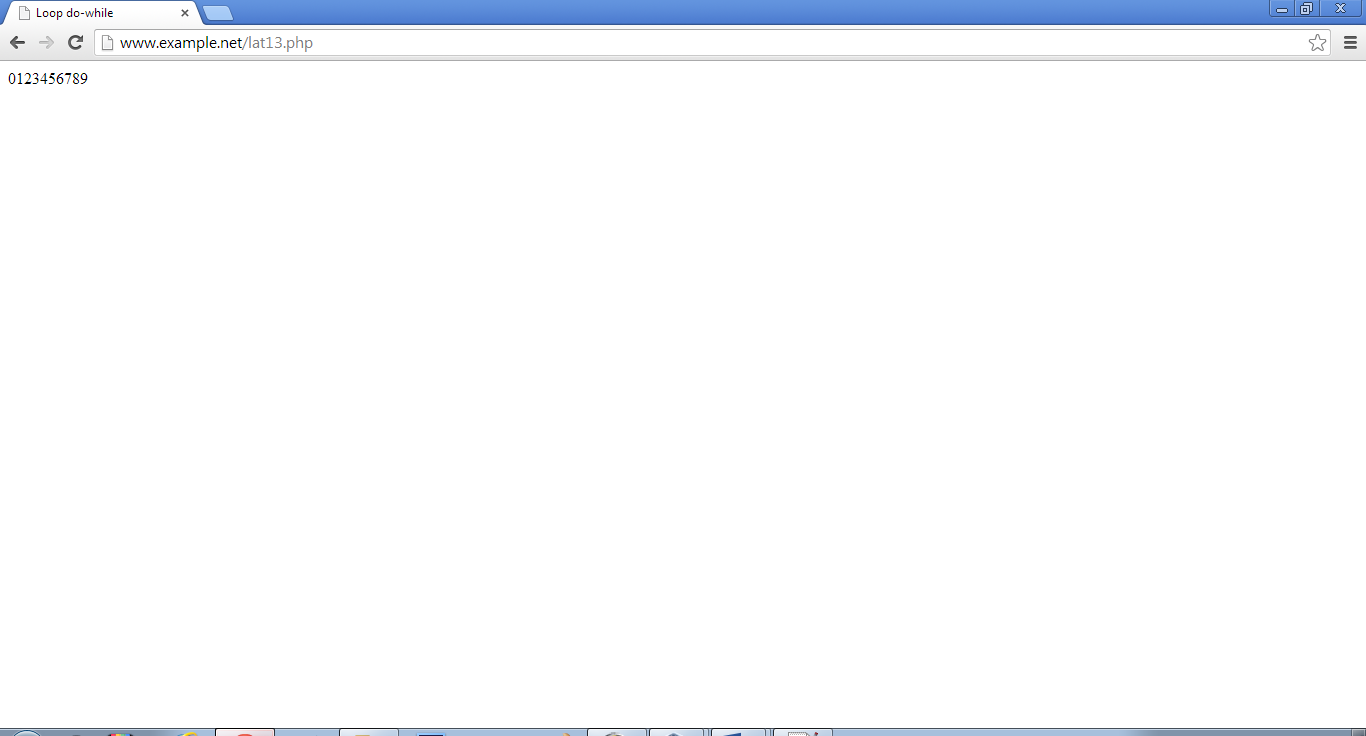
Keterangan

Script diatas menggunakan perulangan while. Dimana jika nilai pada variabel I kurang dari 10 maka nilai variabel I tersebut akan ditambahkan dengan angka 1 sampai memenuhi kondisi.

Latihan 13



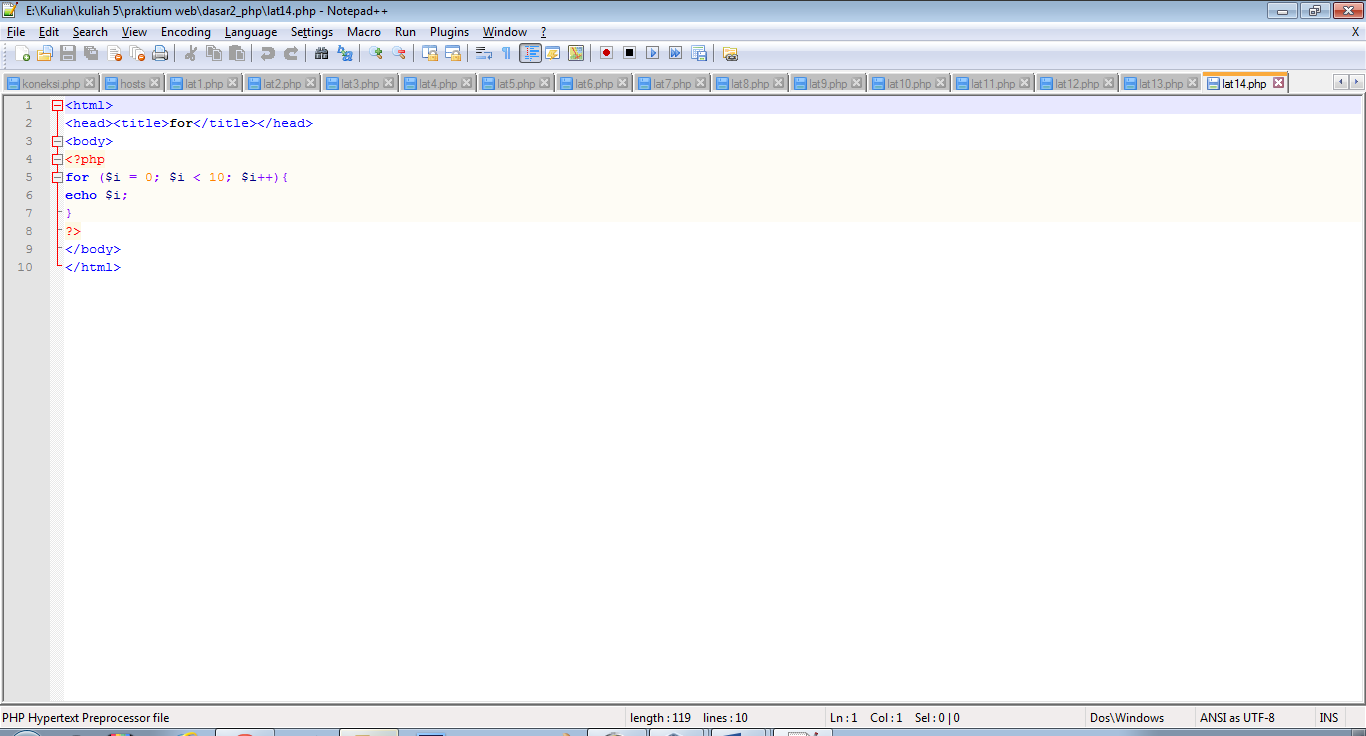
Hasil



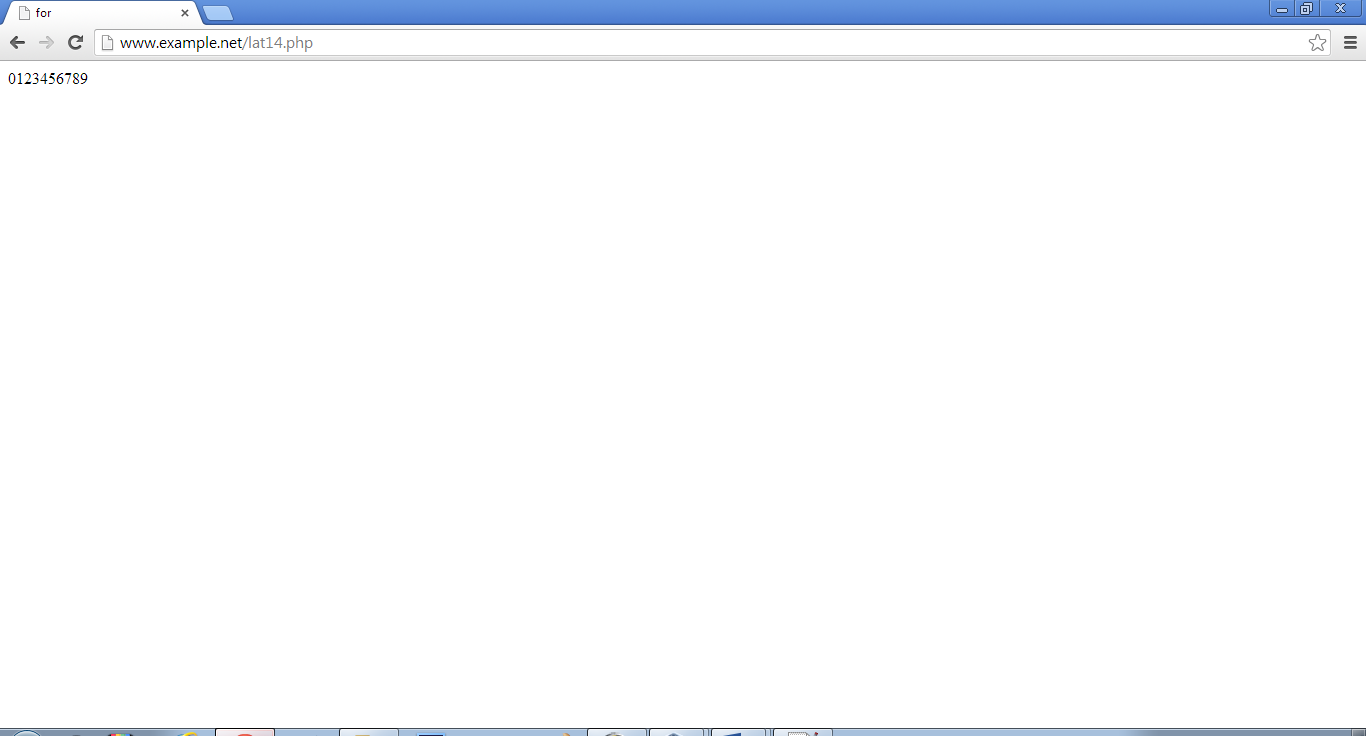
Keterangan

Hampir sama dengan latihan sebelumnya, Do-while akan menjalankan perintah paling tidak satu kali meskipun kondisi bernilai salah.

Latihan 14



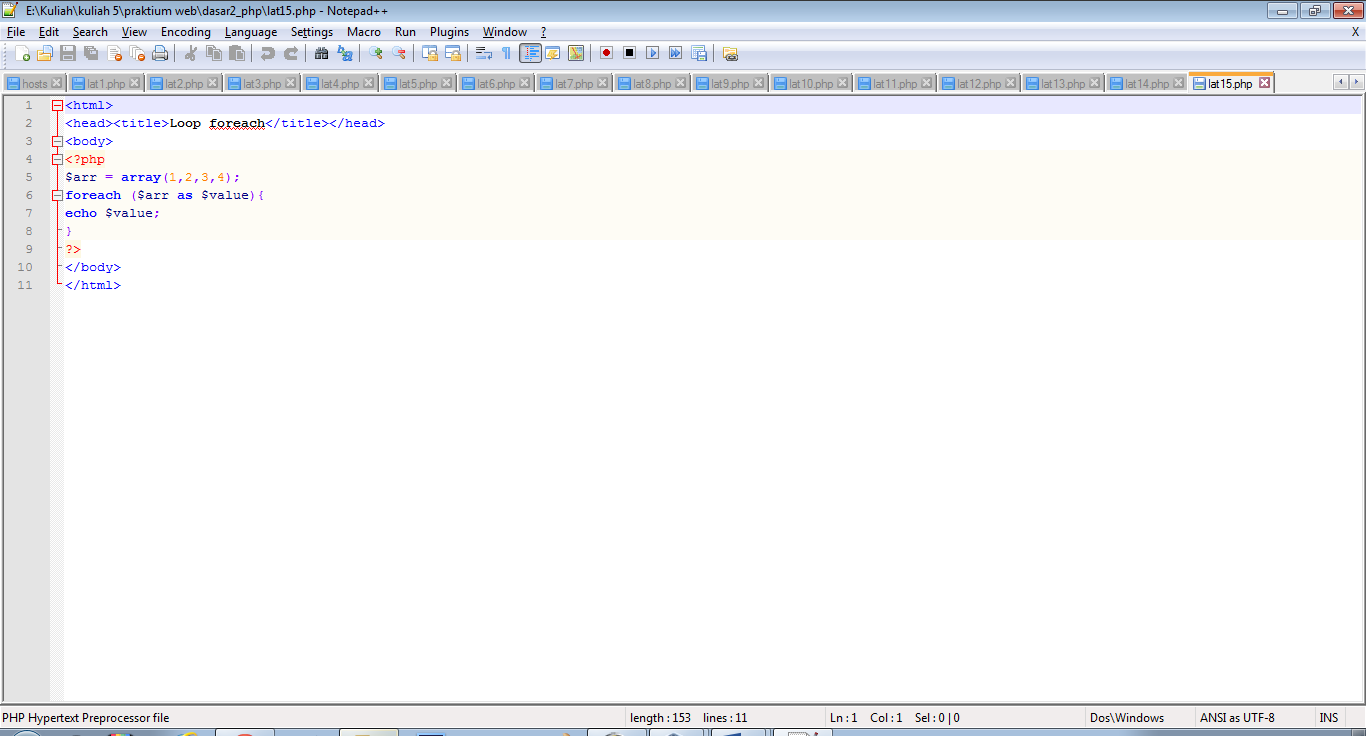
Hasil



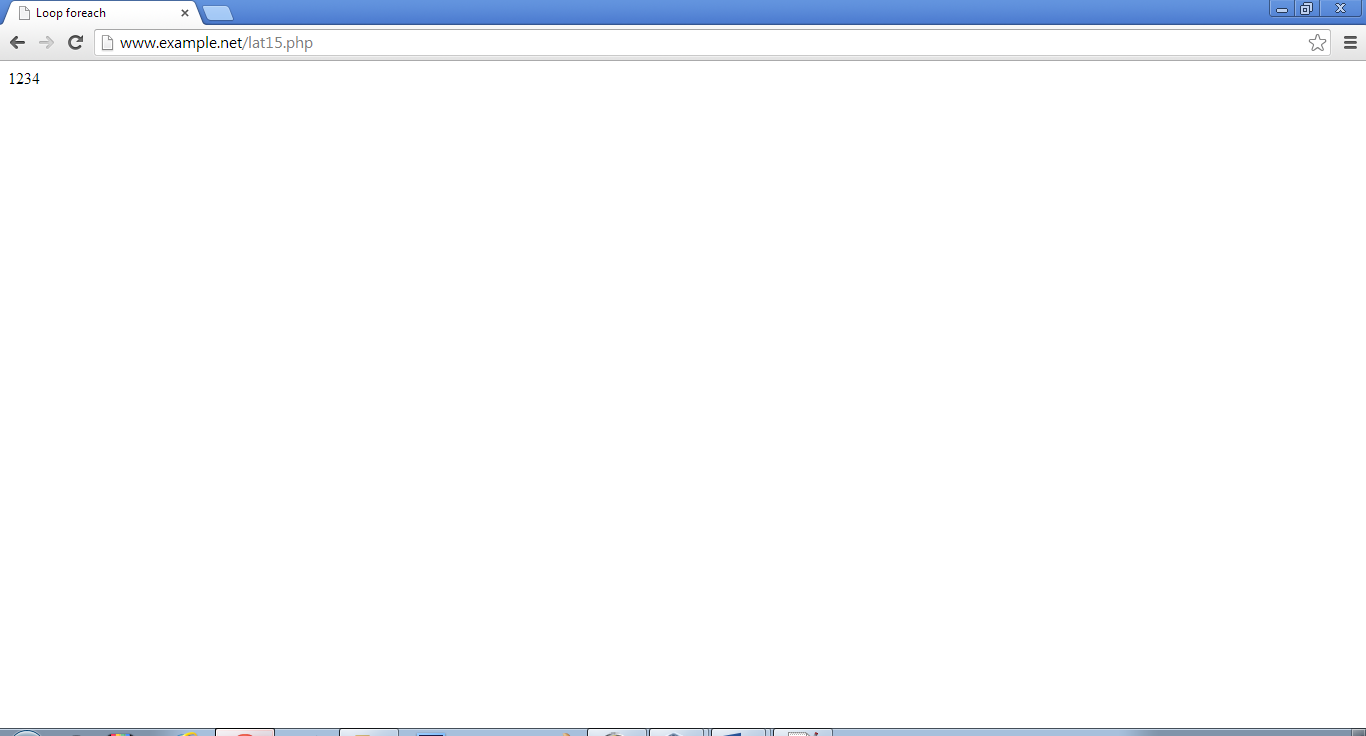
Keterangan

Pada script ini, pengulangan for paling banyak digunakan pada program khususnya ketika jumlah pengulangannya sudah diketahui.

Latihan 15



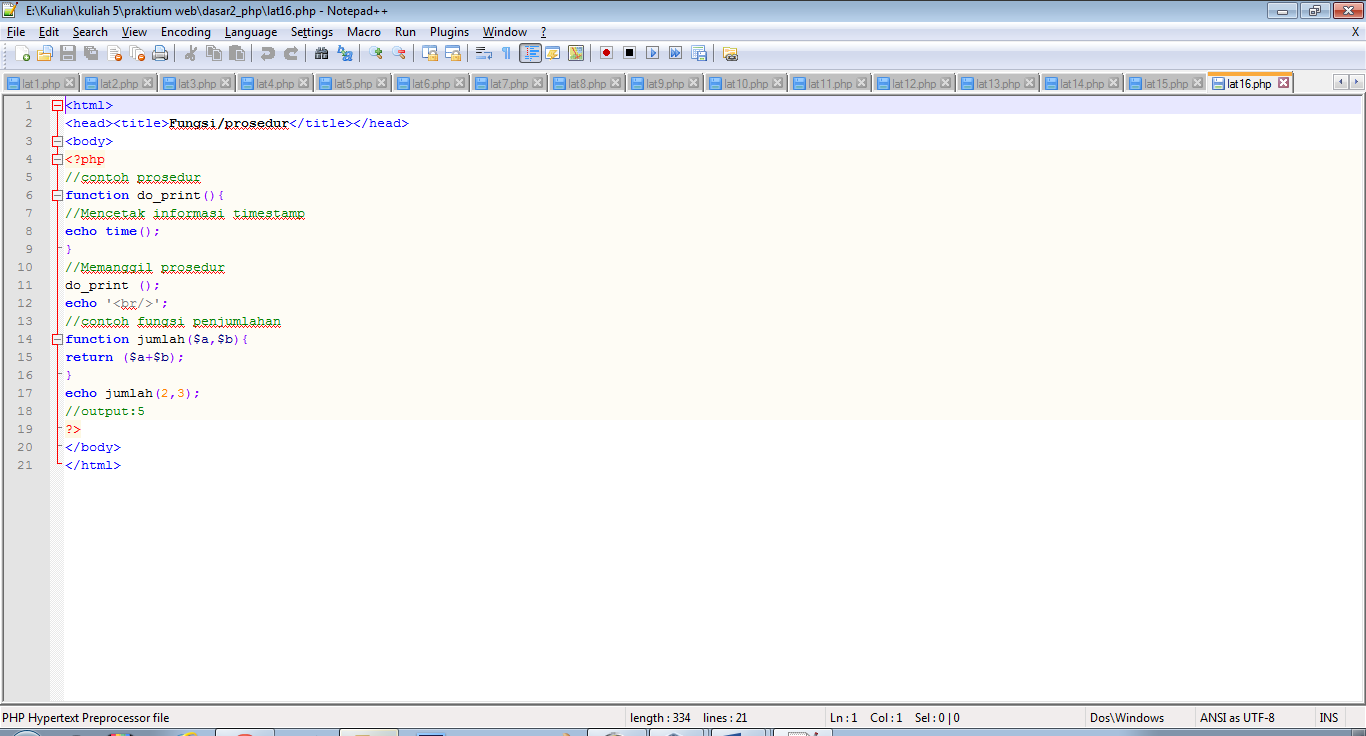
Hasil



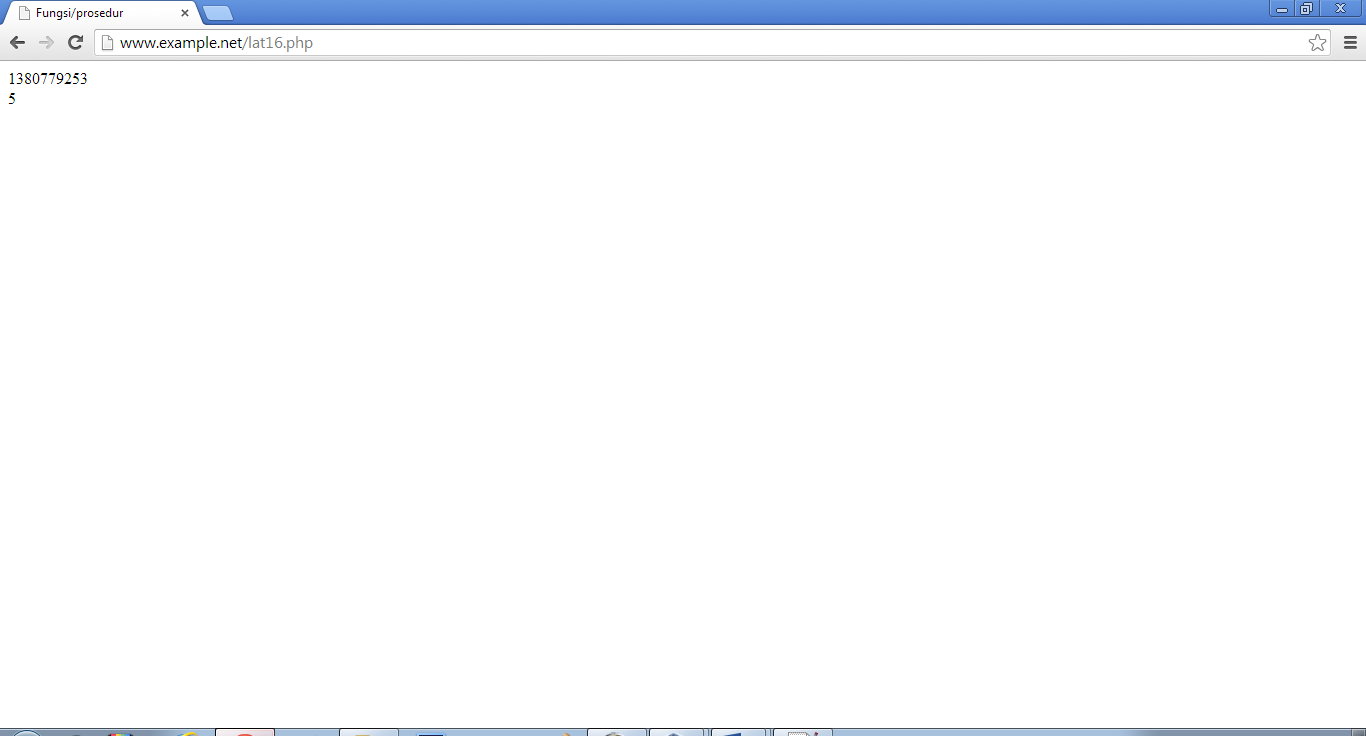
Keterangan

Perintah foreach diatas dapat digunakan untuk melakukan perulangan pada array.

Latihan 16



Hasil



Keterangan

operasi perhitungan dilakukan didalam fungsi, sehingga kita hanya perlu memanggil fungsi tersebut dan memberikannya nilai argumen

**STUDI KASUS**

1. Buat fungsi greeting yang menerima argument integer jam (format 24 jam) dan mengembalikan nilai string (Selamat Pagi, Selamat Siang, atau Selamat Malam). Jika nilai tidak memenuhi maka akan mengembalikan

Coding

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 4.0 Transitional//EN">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>Greeting</title>

</head>

<body>

<?php

function greeting()

{

$date = date ("G:i");

if ($date>=0 and $date <10) {

echo '<h1>Selamat Pagi</h1>';

} else if ($date>=10 and $date<15) {

echo '<h1>Selamat Siang</h1>';

} else if ($date>=18 and $date<24) {

echo "Selamat Malam";

}else echo "";

}

?>

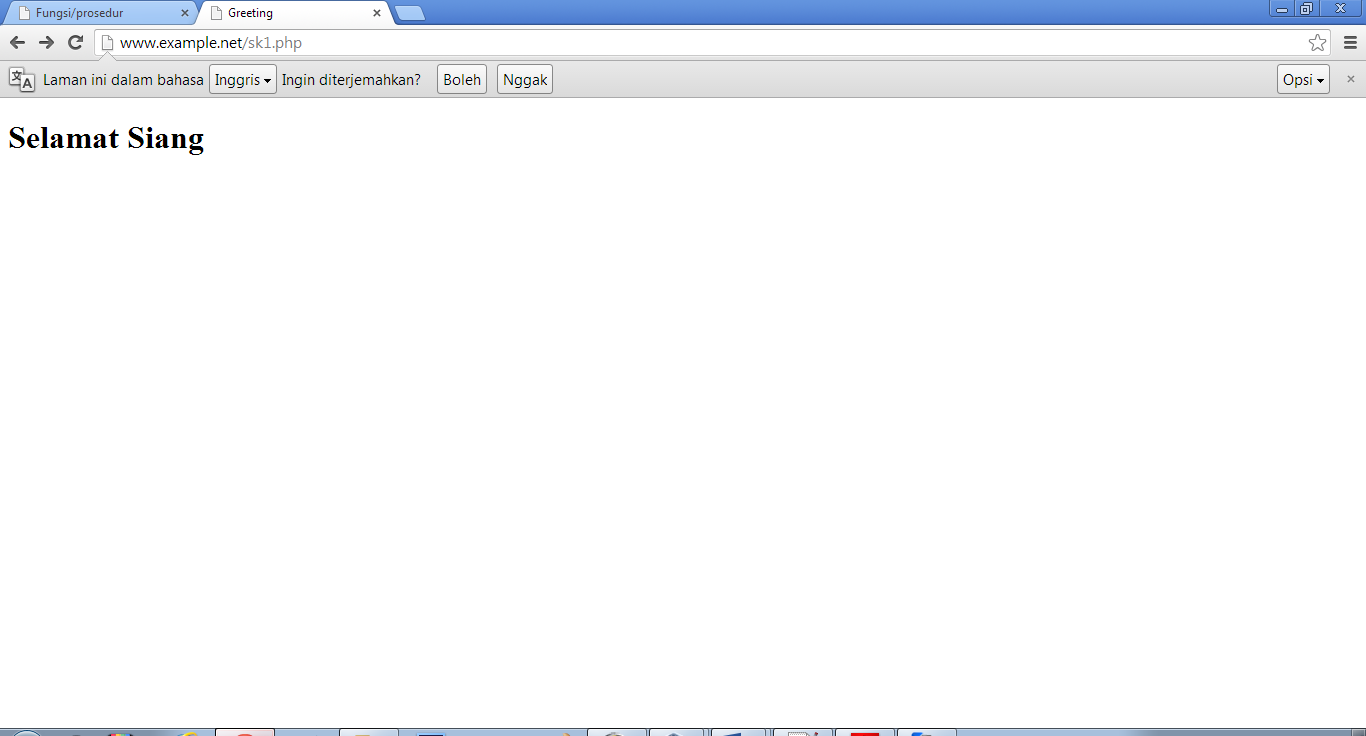
<?php greeting(); ?><br>

<p align="center"><font face="arial" size="5" color="red">

</body>

</html>

Hasil



Script menggunakan perintah kondisi if-else didalamnya untuk menyeleksi argumen yang masuk, lalu mengembalikan nilainya sesuai dengan kondisi

1. Buat fungsi sederhana meng-genarate matriks sel pada tabel

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 4.0 Transitional//EN">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>Generate Matrix Cell</title>

</head>

<body>

<?php

function Tabel($baris, $kolom)

{

echo '<h2 align="center">Tabel '. $baris .' baris dan ' . $kolom . ' kolom</h2>';

echo '<table align="center" width = 40% border=2>';

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

echo "<tr>\n";

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

echo "<td height=30 width=10%></td>\n";

}

echo "</tr>\n";

}

echo "</table>\n";

}

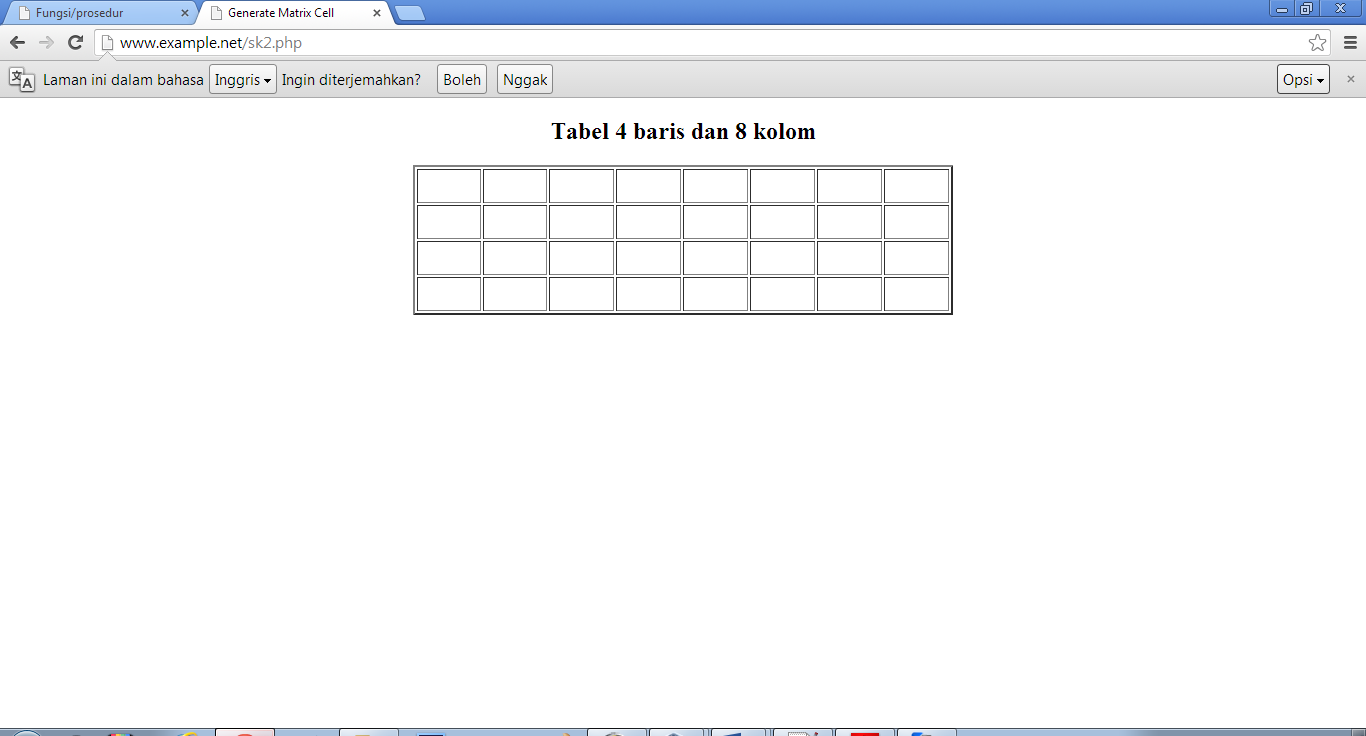
Tabel(4,8); //memanggil fungsi BuatTabel

?>

</body>

</html>

Hasil



Penejelasan

Script diatas digunakan untuk membuat tabel dengan memanggil fungsi “tabel”. Fungsi tabel memiliki dua argumen yaitu “baris” dan “kolom” untuk mementukan batas for

**TUGAS PRAKTIKUM**

1. **Tugas Praktikum 1**

* Passing By Value

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 4.0 Transitional//EN">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>Passing By Reference</title>

</head>

<body>

<?php

function angka(&$angka1) {

$angka1++;

}

$angka2=7;

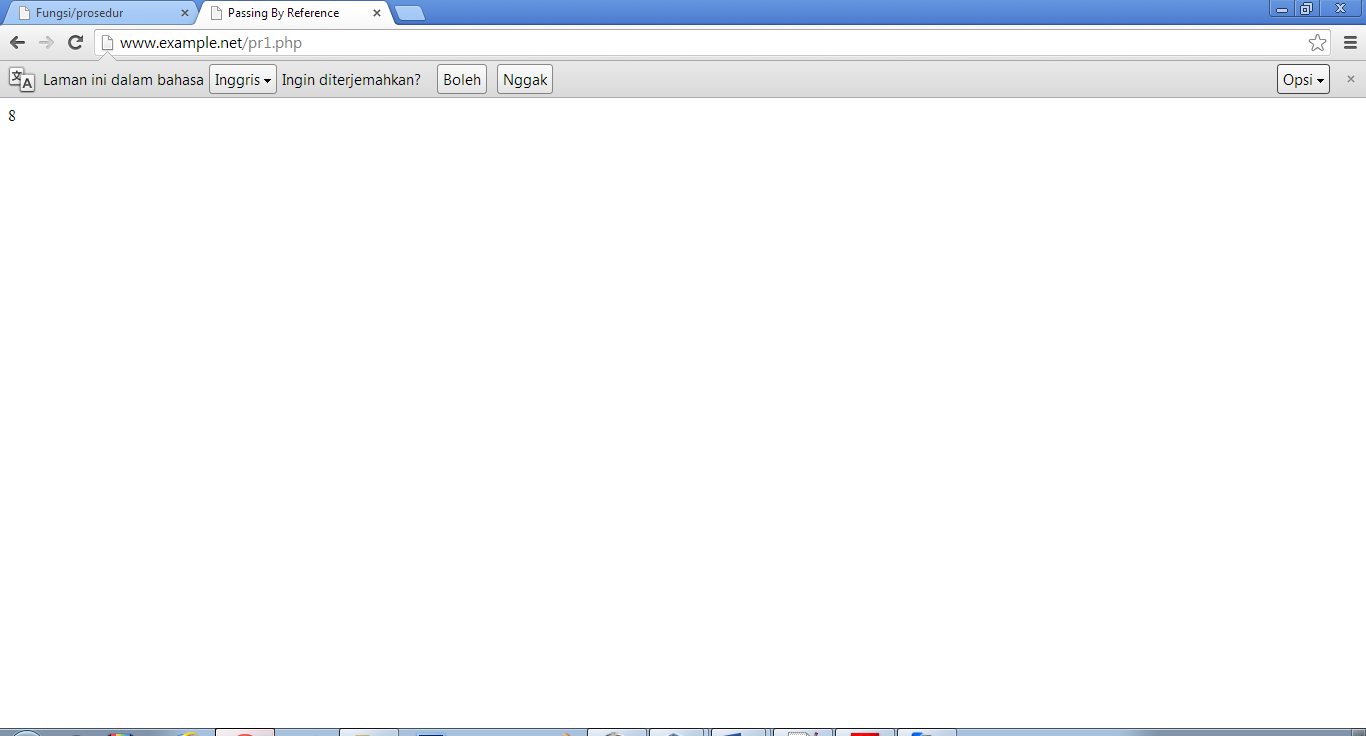
angka($angka2);

echo $angka2;

?>

</html>

Hasil



Keterangan

Variable yang diberikan oleh parameter hanya akan diolah didalam fungsi, variable yang berada diluar fungsi akan tetap.

Passing By Reference

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 4.0 Transitional//EN">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>Passing By Reference</title>

</head>

<body>

<?php

function angka(&$angka1) {

$angka1++;

}

$angka2=7;

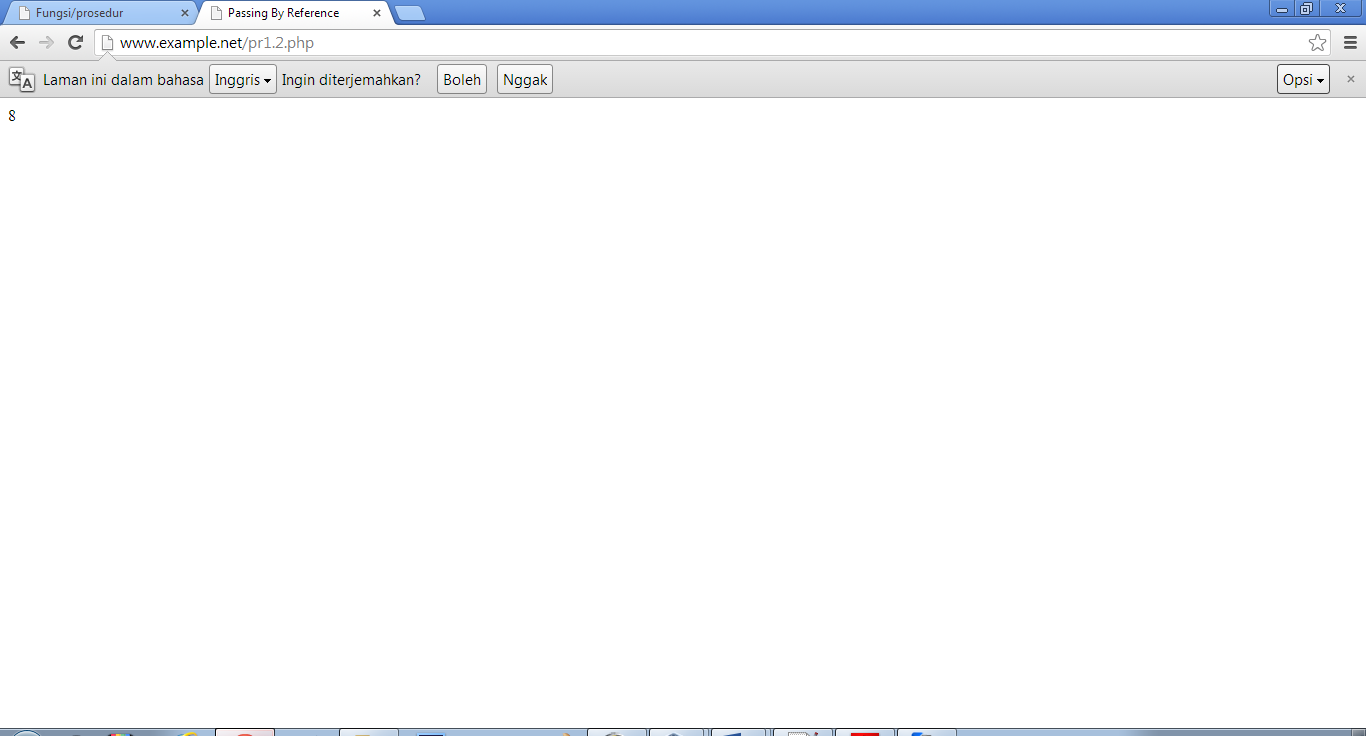
angka($angka2);

echo $angka2;

?>

</html>

Hasil



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 variable didalam fungsi.

**Tugas Praktikum 2**

**Coding**

<?php

$baris = 10;

$kolom = 3;

echo"

BARIS: $baris

<br/>

KOLOM: $kolom

<br/>

";

$s = $baris/$kolom;

$w = $baris%$kolom;

echo"<table>";

$num = 1;

for($a=1; $a<=$s; $a++){

echo"<tr>";

for($b=1; $b<=$kolom; $b++){

echo"<td> $num <td>";

$num++;

}

echo"</tr>";

}

if($w>0)

{

echo"<tr>";

for($b=1; $b<=$w; $b++){

echo"<td>

$num <td>";

$num++;

}

echo"</tr>";

}

echo"</table>";

?>

<style>

td{

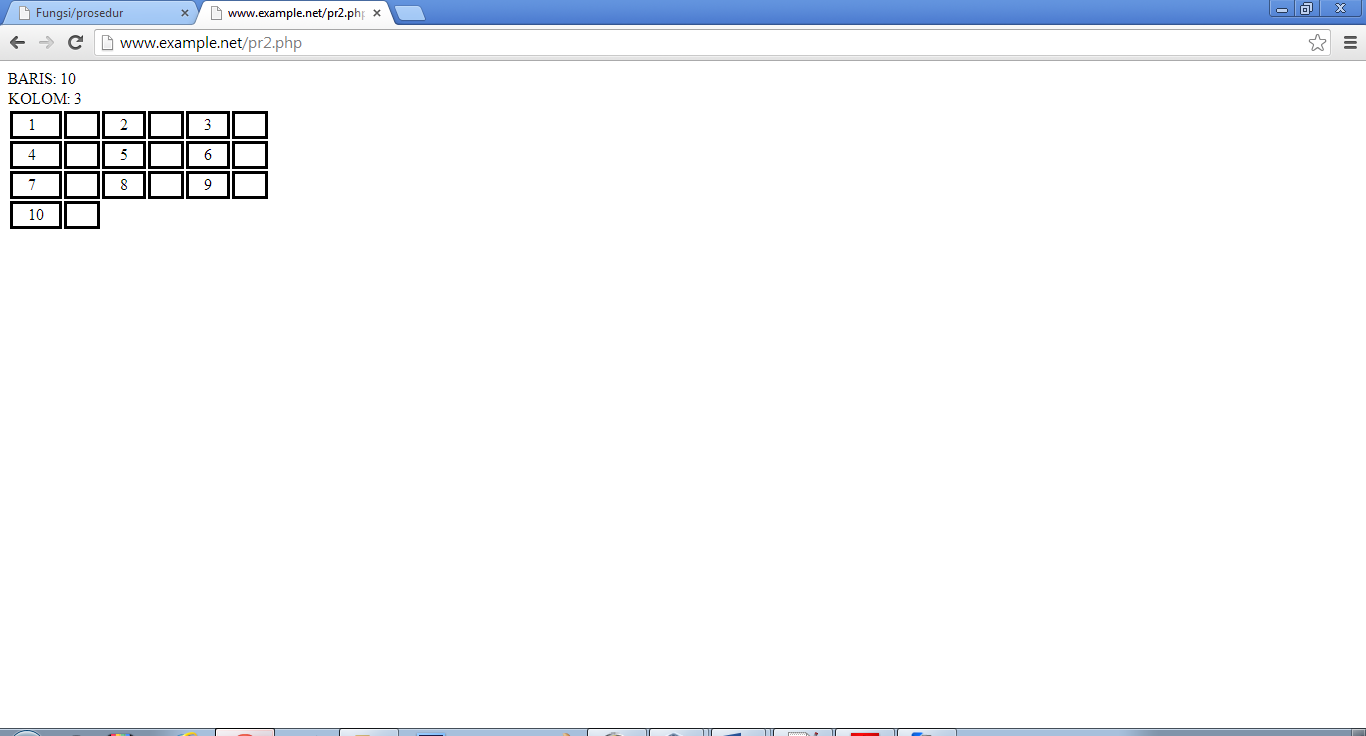
border: solid 3px;

padding: 1px 15px 1px 15px;

}

</style>

Hasil



Penejelasan

Script diatas digunakan untuk membuat tabel dengan cell yang dinamis (fleksibel).

**Kesimpulan**

PHP adalah singkatan dari "PHP: Hypertext Prepocessor". **php** adalah bahasa pemrograman server side yang sudah banyak digunakan pada saat ini, terutama untuk pembuatan website dinamis. Untuk hal-hal tertentu dalam pembuatan web, bahasa pemrograman PHP memang diperlukan, misalnya saja untuk memproses data yang dikirimkan oleh pengunjung web.

Dalam PHP tipe data tidak dideklarasikan oleh programmer akan tetapi ditentukan secara otomatis oleh Intepreter PHP. Jadi anda tidak perlu deklarasikan tipe data tersebut jika ingin memberikan nilai pada suatu variabel. Anda dapat menuliskan **fungsi gettype()**untuk mengetahui type data dari nilai yang anda berikan.

**Daftar Pustaka**

* **id.wikipedia.org/wiki/PHP (online tanggal 3 oktober 2013 jam 15.00 WIB)‎**
* **http://www.carawebs.info/2013/04/apa-itu-php.html (online tanggal 3 oktober 2013 jam 15.00 WIB)‎**
* **modul 2 praktikum web S1 Pendidikan Teknik Informatika UM 2013**