**LAPORAN PRAKTIKUM**

**PEMROGRAMAN BERBASIS WEB**

**“DASAR-DASAR PHP”**

Disusun untuk Memenuhi Tugas Matakuliah Praktikum Multimedia

yang Dibimbing oleh Bapak M. Jauharul Fuady, S.T. M.T.



Oleh:

Imilda Rima Kurniawati

110533430646

S1 PTI 2011 Offering E

**UNIVERSITAS NEGERI MALANG**

**FAKULTAS TEKNIK**

**JURUSAN TEKNIK ELEKTRO**

**OKTOBER 2013**

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;

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.

Contoh penulisannya :

<?php

          $a = 1.234;

         $b = 1.2e3;

        $c = 2E-6;

        $d = 1.2E8;

?>

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 **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.

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 (” “)

|  |  |
| --- | --- |
| 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.

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.

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

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

<?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"];

?>

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 seperti pada

Contoh berikut:

<?php

    # Nilai awal $foo adalan 10

    $foo = 10;

     # $foo akan menjadi NULL

    $foo = NULL;

    # Sama dengan memanggil unset()

    unset($foo);

?>

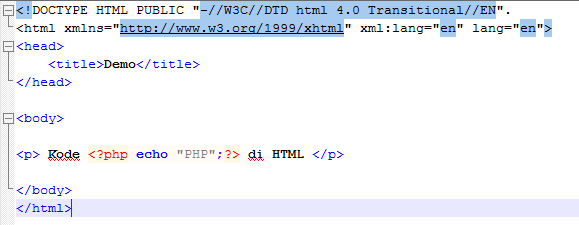
1. **Resources**

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.

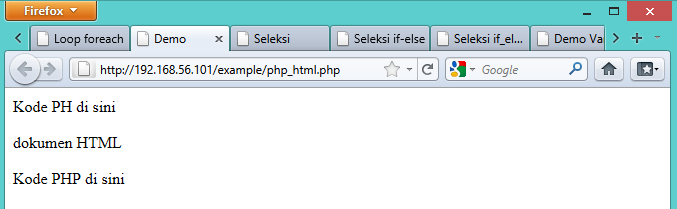
1. **LATIHAN**
2. **Program PHP**

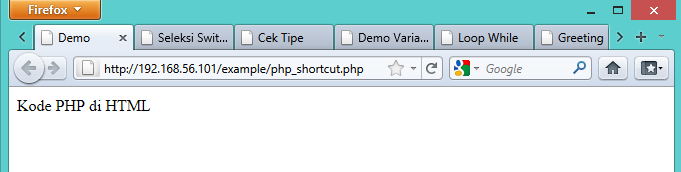
Script :





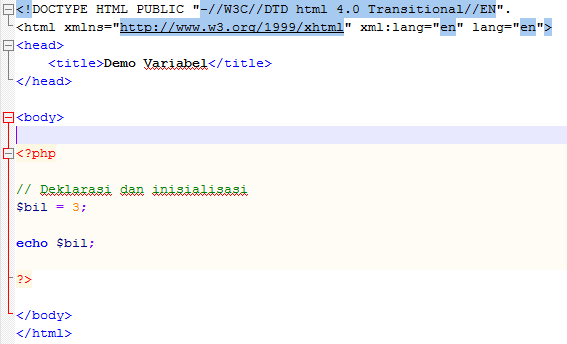
Hasil :



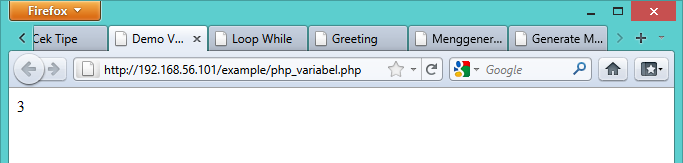


1. **Variabel**

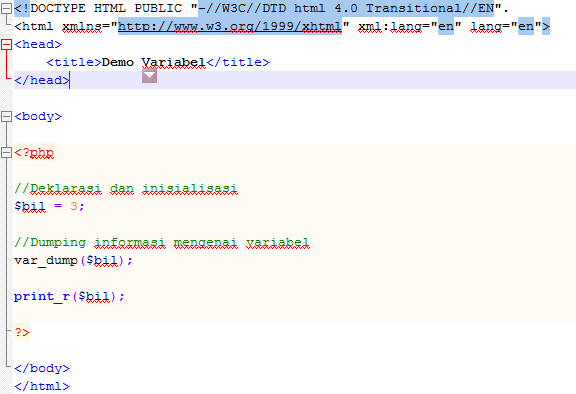
Script :

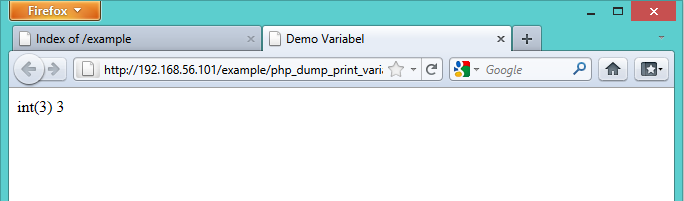


Hasil:

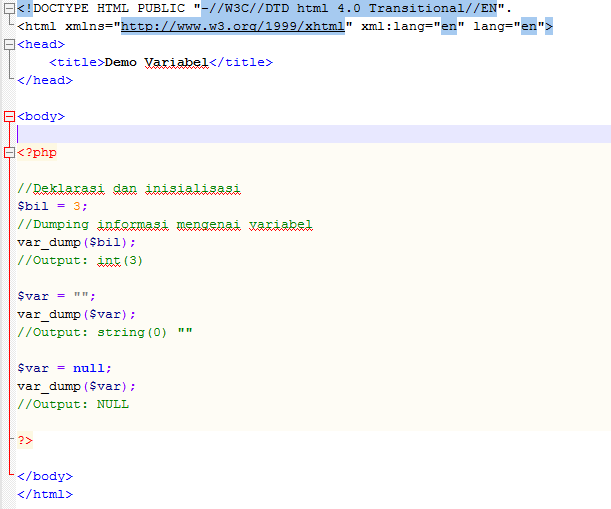


Fungsi var\_dump() atau print\_r

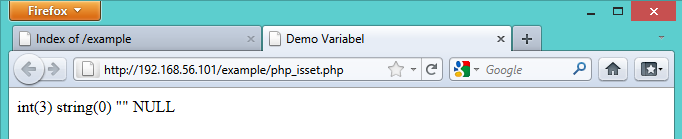


Hasil:

Konstruksi bahasa isset()

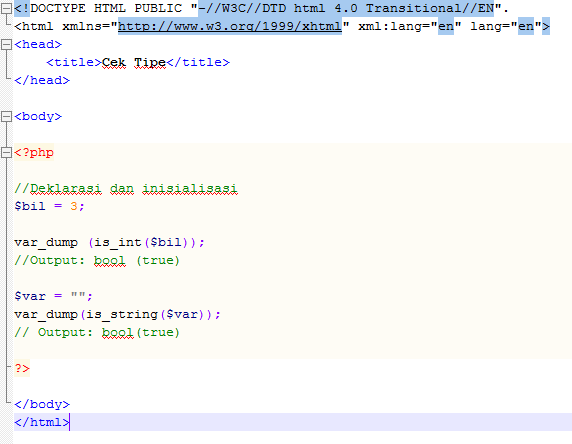


Hasil:

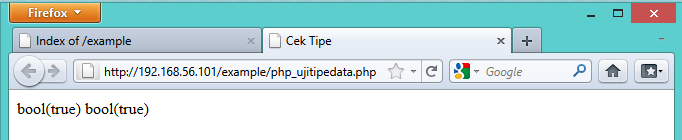


1. **Tipe Data dan Casting**

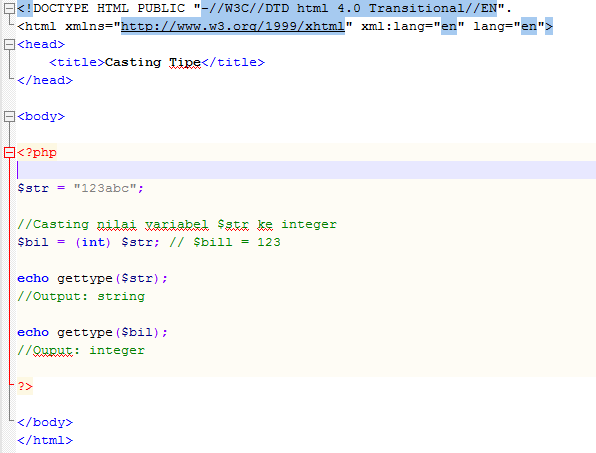
Script Menguji Tipe Data Suatu Variabel



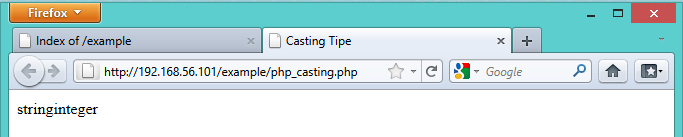
Hasil :



**Casting**



Hasil:

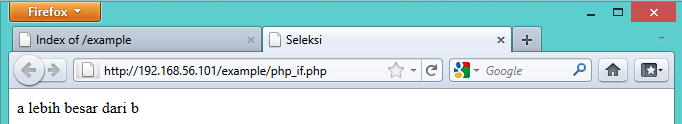


1. **Pernyataan Seleksi**

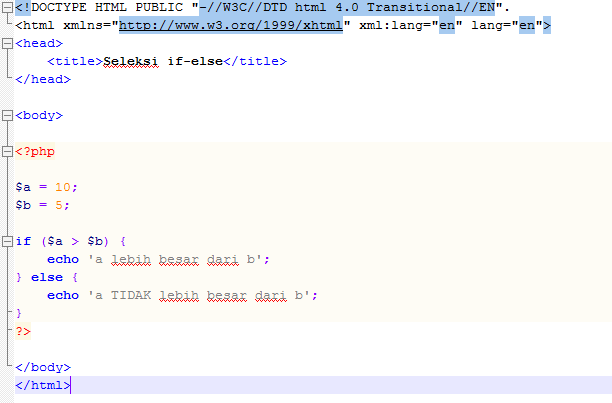
*Pernyataan If*



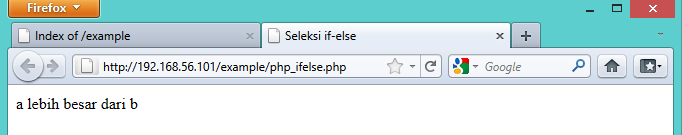
Hasil:



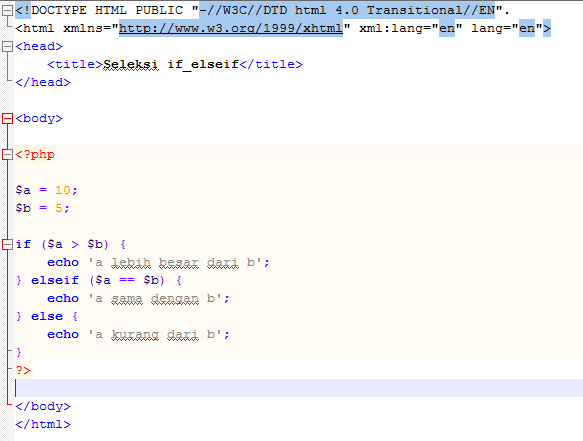
*Pernyataan If-Else*



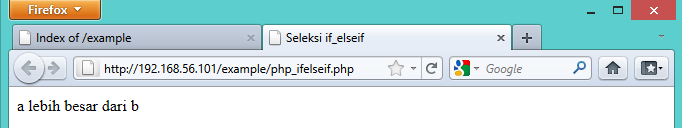
Hasil:



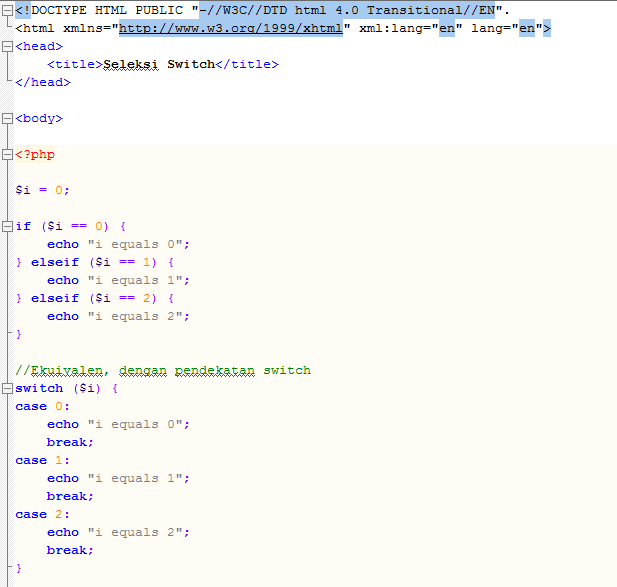
*Pernyataan If-Elseif*



Hasil:

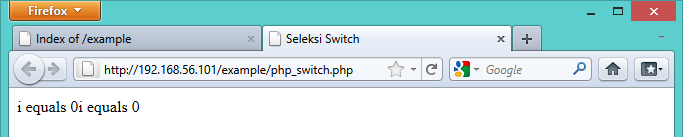


*Pernyataan Switch*



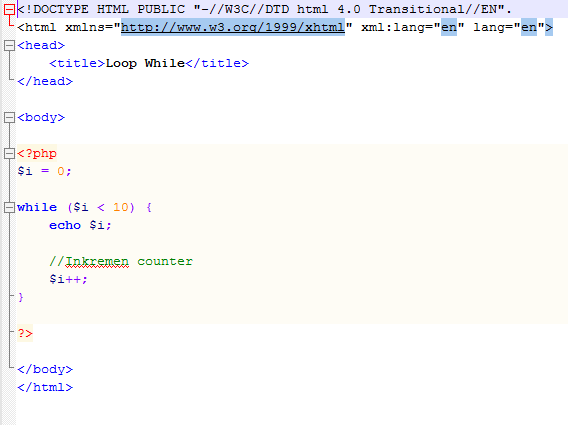


Hasil:

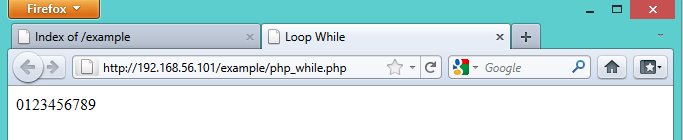


1. **Pengulangan**

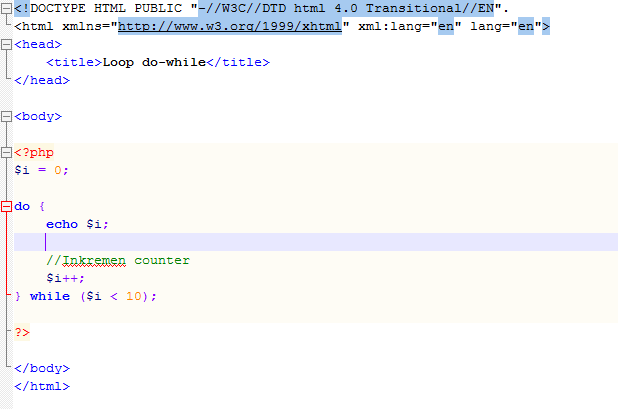
*Pengulangan While*



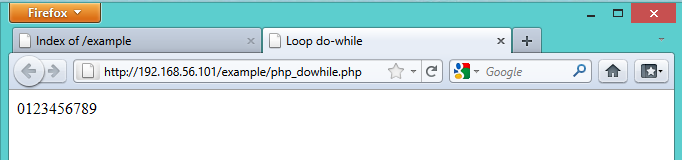
Hasil:



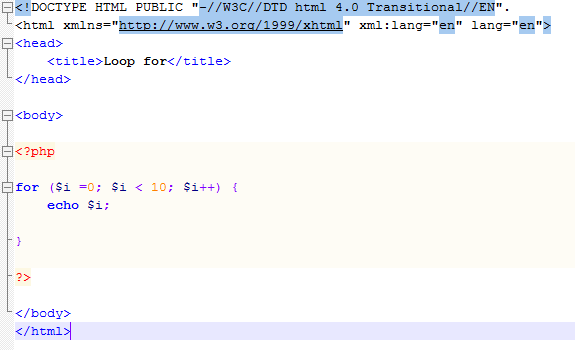
*Pengulangan Do-while*



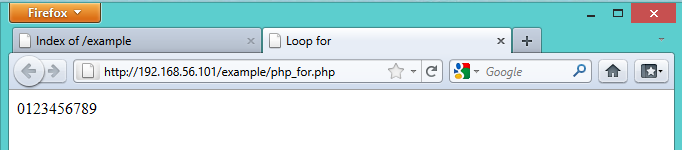
Hasil:



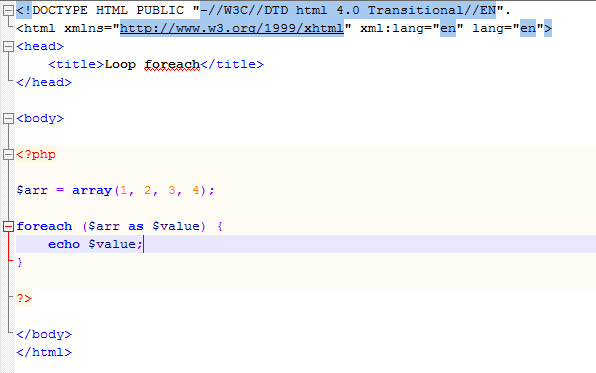
*Pengulangan For*



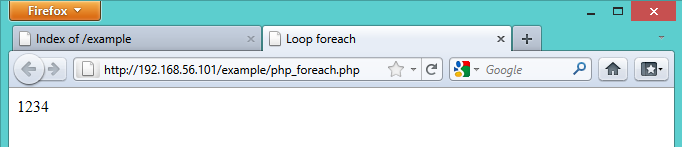
Hasil:



*Pengulangan Foreach*



Hasil:

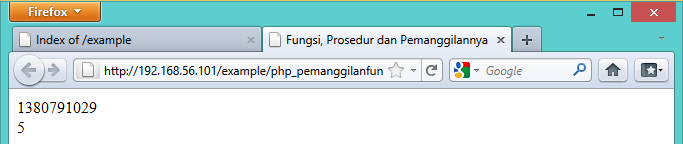


1. **Fungsi Prosedur**

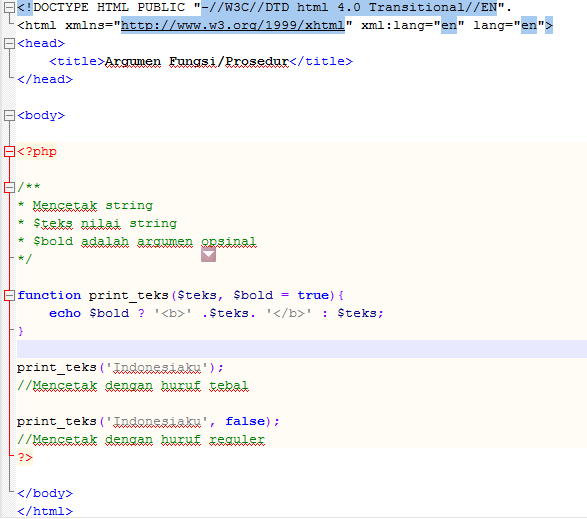
*Definnisi Fungsi dan Prosedur*



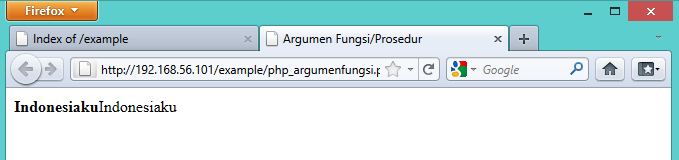
Hasil :



*Argumen Fungsi/Prosedur*



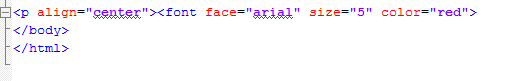
Hasil :



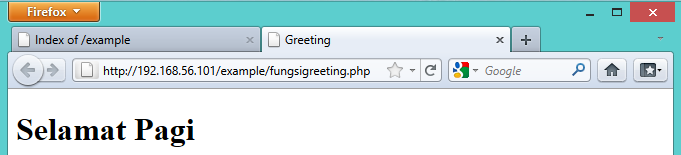
1. **STUDI KASUS**
2. **Fungsi Greeting**

Script :





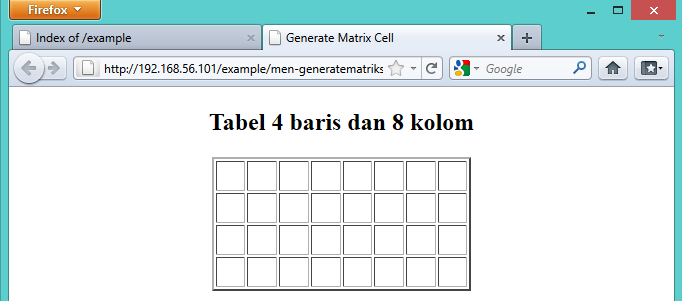
Hasil :



1. **Men-generate Matriks Sel**



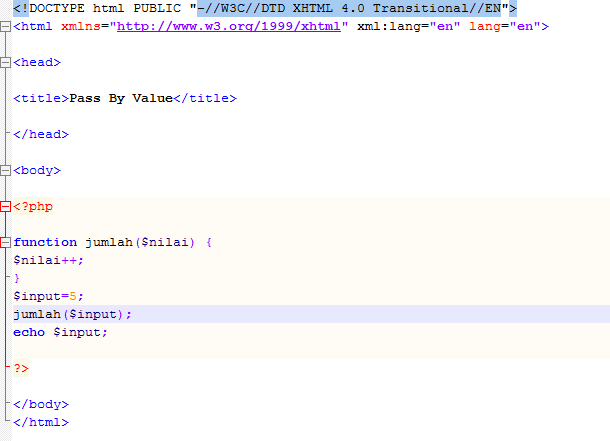
Hasil:



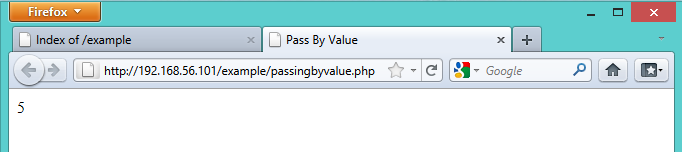
1. **TUGAS PRAKTIKUM**
2. Uraian:

Secara *default*, argumen dikirimkan ke fungsi adalah berupa nilai saja, jadi tidak bisa mengubah nilai variabel yang dipakai sebagi argumen di dalam fungsi. Jika mengubah nilai argumen di dalam fungsi, maka nilai asli pada variabel tetap. Agar dapat mengubah nilai asli variabel di dalam fungsi, maka harus ada pengiriman berdasarkan *reference*, caranya dengan menambah tanda "&" di depan variabel saat memanggil fungsi.

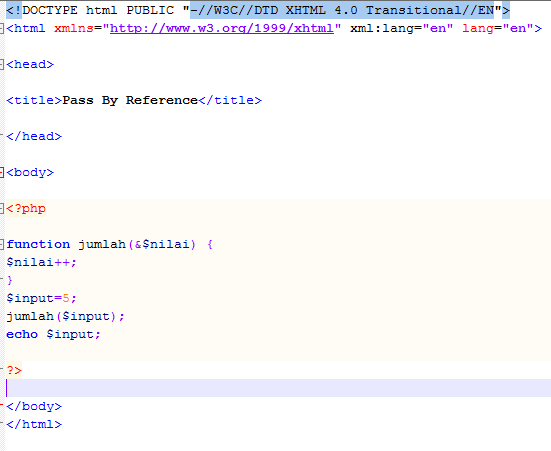
*Pass By Value*



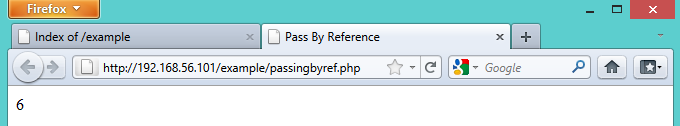
Hasil :



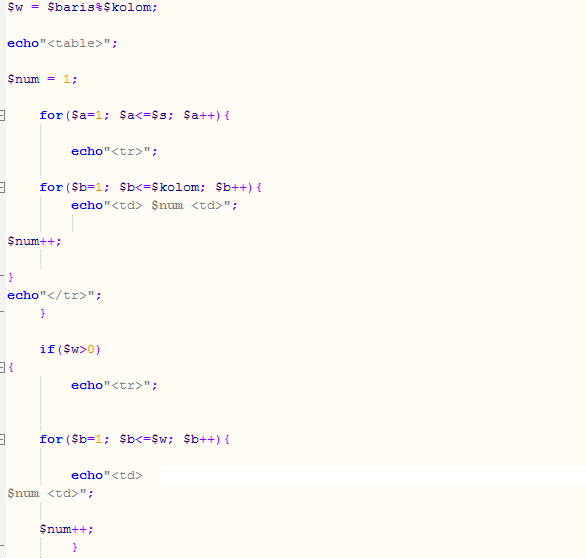
*Pass By Reference*

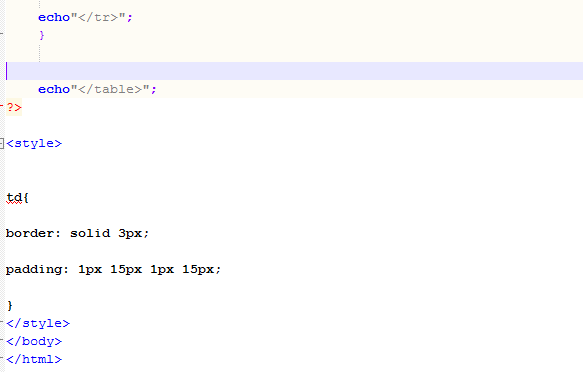


Hasil:



1. **Meng-generate sel tabel**





Hasil:

