**Laporan Praktikum**

**Pemrograman Web**

Modul 1

Dasar-dasar PHP

Oleh

Aulia Rahmah

110533406967



**PRODI S1 PENDIDIKAN TEKNIK INFORMATIKA**

**JURUSAN TEKNIK ELEKTRO**

**FAKULTAS TEKNIK**

**UNIVERSITAS NEGERI MALANG**

**Oktober 2013**

**Dasar-Dasar PHP**

1. **Kompetensi Dasar**

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

1. **Alokasi Waktu**

2 JS (2X50menit)

1. **Dasar Teori**
   1. **PHP**

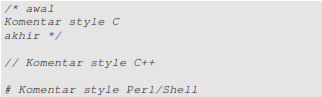
PHP (PHP: Hypertext Preprocessor) merupakan salah satu bahasa webscripting yang sangat powerful. Sejak pertama kali diperkenalkan, bahasa ini dimaksudkan untuk menghasilkan halaman-halaman web yang dinamis. Hingga saat ini, PHP banyak sekali digunakan dalam membuat aplikasi web—baik lokal maupun Internet—dinamis dan atraktif.

* 1. **Tipe Data**

PHP mendukung delapan tipe primitif, yang terdiri dari empat tipe skalar (boolean, integer, float/double, dan string), dua tipe gabungan (array dan object), dan sisanya adalah tipe khusus (NULL dan resource). Meskipun keberadaan tipe data dalam deklarasi variabel tidak penting, namun dalam tahap selanjutnya sangatlah penting.

* 1. **Komentar**

PHP mendukung tiga jenis sintaks untuk menuliskan baris komentar, meliputi style C, C++, dan Perl/Shell.



Meskipun ketiga jenis style di atas valid, namun upayakan untuk memilih

style yang paling sesuai. Umumnya, style C digunakan untuk komentar yang terdiri dari beberapa baris, sedangkan sisanya untuk satu baris.

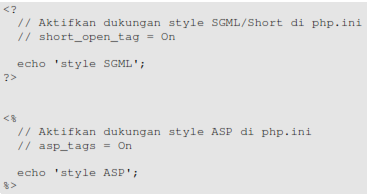
* 1. **Tag PHP**

Parser PHP bekerja dengan menguraikan instruksi-instruksi yang diletakkan di antara tag pembuka dan penutup. Ada empat jenis style tag yang dapat kita gunakan untuk menyatakan bahwa kode merupakan instruksi PHP. Dua dari empat jenis tag yang selalu tersedia tanpa memerlukan konfigurasi adalah style XML dan SCRIPT.



Dua style tag lainnya memerlukan aktivasi dukungan di file konfigurasi

php.ini.

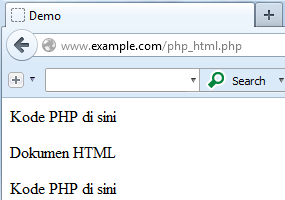


1. **Latihan**
   1. **Program PHP**

Dalam pembuatan aplikasi web, instruksi-instruksi PHP akan dikombinasikan dengan elemen-elemen HTML. Meskipun PHP dapat digunakan untuk membungkus keseluruhan dokumen HTML, namun sebaiknya hal ini tidak kita lakukan. Selain dapat menyulitkan pembacaan kode, langkah ini juga kurang efisien. Pendekatan yang disarankan dalam kombinasi ini adalah memisahkan kode program PHP dari dokumen HTML. Sebagai contoh, ketikkan kode program berikut dan simpan dengan nama php\_html.php (ingat, ekstensi yang digunakan adalah php).

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Demo</title>  </head>  <body>  <?php  echo 'Kode PHP di sini';  // ...  ?>  <p> Dokumen HTML </p>  <?php  echo 'Kode PHP di sini';  // ...  ?>  </body>  </html> |

Hasil



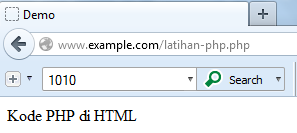
Berkenaan dengan kombinasi ini, tag-tag PHP juga menyediakan shortcut

untuk meringkas penulisan kode. Shortcut ini umumnya digunakan ketika

ingin menuliskan kode-kode kecil di dalam HTML.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Demo</title>  </head>  <body>  <p> Kode <?php echo "PHP";?> di HTML </p>  </body>  </html> |

Hasil

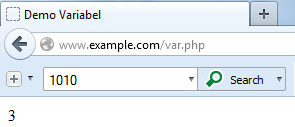


* 1. **Variabel**

Sebagaimana diketahui, variabel diidentifikasikan melalui karakter dollar ($) dan diikuti nama variabel. PHP mengizinkan kita untuk mendeklarasikan dan meng-assign suatu nilai sekaligus pada saat variabel akan digunakan.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Demo Variabel</title>  </head>  <body>  <?php  // Deklarasi dan inisialisasi  $bil = 3;  echo $bil;  ?>  </body>  </html> |

Hasil



Di tahap pengembangan, kita bisa memanfaatkan fungsi **var\_dump()** atau **print\_r()** untuk memudahkan pemeriksaan variabel.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Demo Variabel</title>  </head>  <body>  <?php  // Deklarasi dan inisialisasi  $bil = 3;  // Dumping informasi mengenai variabel  var\_dump($bil);  print\_r($bil);  ?>  </body>  </html> |

Hasil



Amati dan pahami hasil keluaran dari fungsi **var\_dump().** Apa bedanya **var\_dump()** dengan konstruksi bahasa echo? Jelaskan!

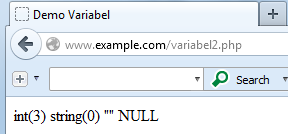
**Jawaban :**

Perbedaan var\_dump() dan dengan kostruksi bahasa echo adalah perintah echo juga dapat mencetak spesial karakter yang memiliki arti tertentu sedangkan var\_dump() dapat menampilkan atau menduplikat array.

Apabila Anda ingin memeriksa apakah suatu variabel di-set atau tidak, gunakan konstruksi bahasa **isset()**. Konstruksi bahasa ini juga dapat digunakan untuk memeriksa beberapa variabel sekaligus.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Demo Variabel</title>  </head>  <body>  <?php  $bil = 3;  var\_dump($bil);  // Output: int(3)  $var = "";  var\_dump($var);  // Output: string(0) ""  $var = null;  var\_dump($var);  // Output: NULL  ?>  </body>  </html> |

Hasil

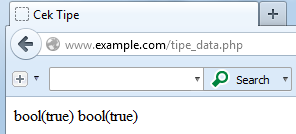


* 1. **Tipe Data dan Casting**

PHP menyediakan fungsi-fungsi berawalan is\_ yang dapat dimanfaatkan untuk menguji tipe data suatu variabel.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Cek Tipe</title>  </head>  <body>  <?php  $bil = 3;  var\_dump(is\_int($bil));  // Output: bool(true)  $var = "";  var\_dump(is\_string($var));  // Output: bool(true)  ?>  </body>  </html> |

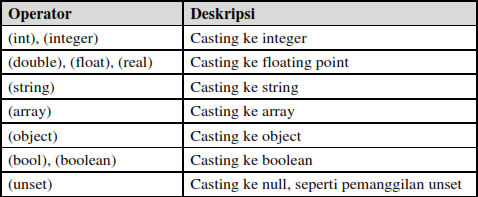
Hasil



Kecuali fungsi **is\_bool()**, penamaan fungsi-fungsi lainnya persis seperti tipe data terkait, misalnya **is\_integer()**, **is\_float(), is\_object(),** dan **is\_null()**.

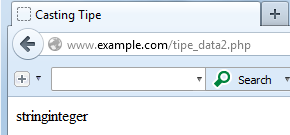
**Casting**

Untuk menguji aplikasi web, kita mempublikasikannya ke web server, baik secara lokal maupun Internet. Lingkungan lokal tentu merupakan pilihan yang efisien, khususnya ketika aplikasi masih dalam tahap pengembangan.



|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Casting Tipe</title>  </head>  <body>  <?php  $str = '123abc';  // Casting nilai vaiabel $str ke integer  $bil = (int) $str; // $bil = 123  echo gettype($str);  // Output: string  echo gettype($bil);  // Output: integer  ?>  </body>  </html> |

Hasil



* 1. **Pernyataan Seleksi**

Pernyataan seleksi untuk pengambilan keputusan merupakan bagian dari

bahasa pemrograman yang sangat penting. Pernyataan-pernyataan pengambilan keputusan di PHP diklasifikasikan ke dalam empat bagian:

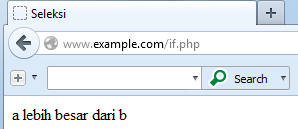
**if, if-else, if-elseif,** dan **switch**.

* **Pernyataan if**

Pernyataan if terdiri dari suatu ekspresi dan sebuah statemen atau blok statemen yang dieksekusi apabila ekspresi bernilai **true**.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Seleksi</title>  </head>  <body>  <?php  $a = 10;  $b = 5;  if ($a > $b) {  echo 'a lebih besar dari b';  }  ?>  </body>  </html> |

Hasil

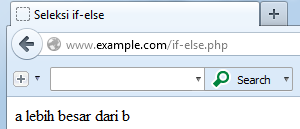


* **Pernyataan if-else**

Pernyataan ini sebenarnya merupakan ekspansi dari pernyataan if. Di sini blok pertama akan dieksekusi manakala ekspresi bernilai true, sebaliknya—jika ekspresi bernilai false—blok kedua yang dieksekusi.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Seleksi if-else</title>  </head>  <body>  <?php  $a = 10;  $b = 5;  if ($a > $b) {  echo 'a lebih besar dari b';  } else {  echo 'a TIDAK lebih besar dari b';  }  ?>  </body>  </html> |

Hasil

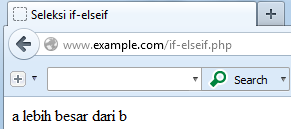


* **Pernyataan if-else if**

Pernyataan ini sebenarnya merupakan ekspansi dari if-else, di mana di ditambahkan lagi blok if-elseif. Bentuk pernyataan if-elseif memungkinkan kita untuk menciptakan seleksi yang lebih kompleks.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Seleksi if-elseif</title>  </head>  <body>  <?php  $a = 10;  $b = 5;  if ($a > $b) {  echo 'a lebih besar dari b';  } elseif ($a == $b) {  echo 'a sama dengan b';  } else {  echo 'a kurang dari b';  }  ?>  </body>  </html> |

Hasil

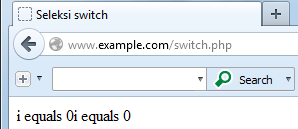


* **Pernyataan switch**

Pernyataan switch merupakan sebuah pernyataan control flow yang dimulai dengan suatu ekspresi dan mentransfer kontrol ke satu kasus berdasarkan nilai ekspresi.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Seleksi switch</title>  </head>  <body>  <?php  $i = 0;  if ($i == 0) {  echo "i equals 0";  } elseif ($i == 1) {  echo "i equals 1";  } elseif ($i == 2) {  echo "i equals 2";  }  // Ekuivalen, dengan pendekatan switch  switch ($i) {  case 0:  echo "i equals 0";  break;  case 1:  echo "i equals 1";  break;  case 2:  echo "i equals 2";  break;  }  ?>  </body>  </html> |

Hasil



* 1. **Pengulangan**

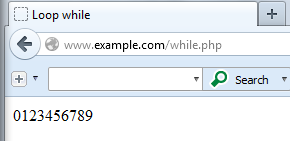
Bagian ini akan menguraikan jenis-jenis dan implementasi struktur kontrol perulangan.

* **Pengulangan while**

Pada pernyataan ini, ekspresi akan dievaluasi dan pengulangan dieksekusi jika dan hanya jika ekspresi bernilai true.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Loop while</title>  </head>  <body>  <?php  $i = 0;  while ($i < 10) {  echo $i;  // Inkremen counter  $i++;  }  ?>  </body>  </html> |

Hasil

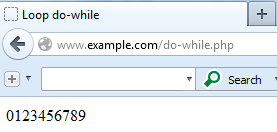


* **Pengulangan do-while**

Bentuk pengulangan ini mirip dengan while, kecuali bahwa ekspresi pengontrolan pengulangan dilakukan di akhir blok. Ini juga berarti bahwa blok pengulangan akan dieksekusi sedikitnya satu kali, meskipun ekspresi bernilai **false.**

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Loop do-while</title>  </head>  <body>  <?php  $i = 0;  do {  echo $i;  // Inkremen counter  $i++;  } while ($i < 10);  ?>  </body>  </html> |

**Hasil**

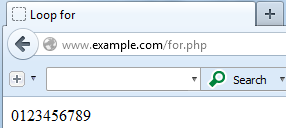


* **Pengulangan for**

Pernyataan pengulangan ini paling banyak digunakan di dalam program, khususnya ketika jumlah iterasinya sudah diketahui. Struktur pengulangan for terdiri dari tiga bagian: ekspresi inisialisasi, kondisi perulangan, dan ekspresi inkremen.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Loop for</title>  </head>  <body>  <?php  for ($i = 0; $i < 10; $i++) {  echo $i;  }  ?>  </body>  </html> |

Hasil

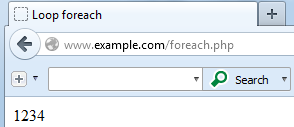


* **Pengulangan foreach**

Sebagai tambahan, PHP juga menyediakan konstruksi foreach yang dapat digunakan untuk melakukan iterasi di array atau koleksi.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Loop foreach</title>  </head>  <body>  <?php  $arr = array(1, 2, 3, 4);  foreach ($arr as $value) {  echo $value;  }  ?>  </body>  </html> |

Hasil



* 1. **Fungsi dan Prosedur**

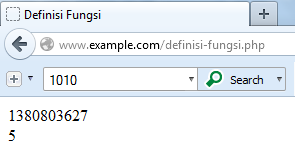
Keberadaan fungsi/prosedur sangat membantu dalam mengorganisir kode program dan menerapkan aspek guna ulang. Di PHP, pembuatan fungsi dan prosedur menggunakan sintaks sama, di mana yang membedakan hanya pengembalian nilai.

* **Definisi Fungsi/Prosedur**

Contoh definisi fungsi dan prosedur beserta cara pemanggilannya diperlihatkan sebagai berikut:

|  |
| --- |
| <?php  // Contoh prosedur  function do\_print() {  // Mencetak informasi timestamp  echo time();  }  // Memanggil prosedur  do\_print();  echo '<br />';  // Contoh fungsi penjumlahan  function jumlah($a, $b) {  return ($a + $b);  }  echo jumlah(2, 3);  // Output: 5  ?> |

Hasil

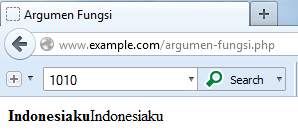


* **Argumen Fungsi/Prosedur**

Suatu fungsi dapat memiliki nol atau lebih argumen. Adapun jika diperlukan, juga bisa dideklarasikan argumen yang sifatnya opsional. Deklarasi ini sekaligus menginisialisasi nilai default pada argumen. Selain itu, argumen opsional harus diletakkan di urutan paling belakang.

|  |
| --- |
| <?php  /\*\*  \* Mencetak string  \* $teks nilai string  \* $bold adalah argumen opsional  \*/  function print\_teks($teks, $bold = true) {  echo $bold ? '<b>' .$teks. '</b>' : $teks;  }  print\_teks('Indonesiaku');  // Mencetak dengan huruf tebal  print\_teks('Indonesiaku', false);  // Mencetak dengan huruf reguler  ?> |

Hasil



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

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html xmlns="http://www.w3.org/1999/xhtml">  <head>  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />  <title>Studi Kasus 1 - Greeting </title>  </head>  <body>  <h4>\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*</h4>  <h2>\*\* Fungsi Greeting \*\*</h2>  <h4>\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*</h4>  <p>  <justify>  <h1><?php  $MyTime=08.00;  function greeting($jam){  if ($jam>=00.01 && $jam<10.00 ) {  echo "Selamat Pagi, Selamat Beraktifitas ";  }  else if ($jam>=10.00 && $jam<15.00) {  echo "Selamat Siang";  }  else if ($jam>=15.00 && $jam<18.00) {  echo "Selamat Sore";  }  else if ($jam>=18.00 && $jam<23.59) {  echo "Selamat Malam,";  }  }  greeting($MyTime);  ?>  </h1>  </justify>  </p>  </body>  </html> |

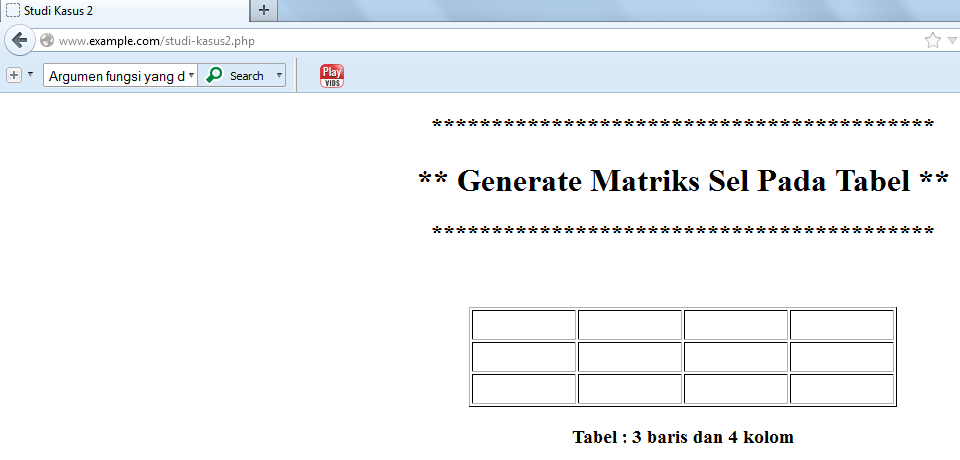
Hasil



* 1. Buat fungsi sederhana untuk men-generate matriks sel pada tabel. Misalkan diberikan argumen 3 (baris) dan 4 (kolom), maka program akan menampilkan tabel berisi 3 baris dan 4 kolom sel.

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html xmlns="http://www.w3.org/1999/xhtml">  <head>  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />  <title>Studi Kasus 2</title>  </head>  <body bgcolor="white">  <center>  <h2>\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*</h2>  <h1>\*\* Generate Matriks Sel Pada Tabel \*\*</h1>  <h2>\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*</h2>  <br>  <br>  <?php  function buat\_table($cell, $row)  {  echo("<table border=\"1\">\n");  for($b = 0; $b < $cell; $b++) {  echo("<tr height=\"30px\">\n");  for($k = 0; $k < $row; $k++) {  echo("<td width=\"100px\"></td>\n");  }  echo("</tr>\n");  }  echo("</table>\n");  echo("<h3>Tabel : $cell baris dan $row kolom</h3>");  }  buat\_table(3,4);  ?>  </center>  </body>  </html> |

Hasil



1. **Tugas Praktikum**
   1. Uraikan secara ringkas mengenai *passing* argumen di PHP dan berikan contoh *passing by value* dan *by reference*.

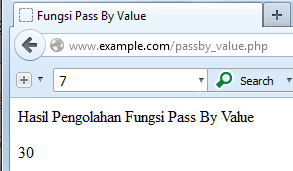
**Pass By Value**

Argumen fungsi yang dilewatkan secara pass by value (default dalam php) berarti membuat copy dari argumen yang asli sehingga argumen asli tersebut tidak ikut berubah dengan adanya proses pada fungsi terhadap argumen tersebut.

Ketika pass-by-values terjadi, method membuat sebuah salinan dari nilai variable yang dikirimkan ke method. Walaupun demikian, method tidak dapat secara langsung memodifikasi nilai variabel pengirimnya meskipun parameter salinannya sudah dimodifikasi nilainya di dalam method.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Fungsi Pass By Value</title>  </head>  <body>  <p>Hasil Pengolahan Fungsi Pass By Value</p>  <?php  function jumlah($nilai) {  $nilai++;  }  $input=30;  jumlah($input);  echo $input;  ?>  </body>  </html> |

Hasil

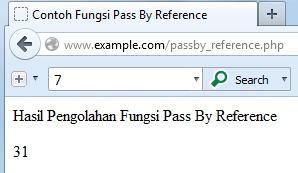


**Pass By Reference**

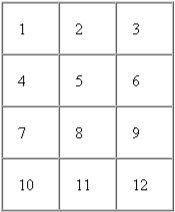
Argumen fungsi yang dilewatkan secara pass by reference berarti membawa argumen asli ke dalam fungsi sehingga argumen asli tersebut akan ikut berubah dengan adanya proses pada fungsi terhadap argumen tersebut. Untuk mengubah jadi passing by reference, kita hanya cukup menambahkan operator & pada argumennya.

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  <head>  <title>Contoh Fungsi Pass By Reference</title>  </head>  <body>  <p>Hasil Pengolahan Fungsi Pass By Reference</p>  <?php  function jumlah(&$nilai){  $nilai++;  }  $input=30;  jumlah($input);  echo $input;  ?>  </body>  </html> |

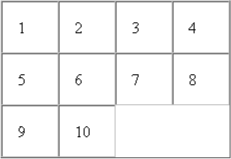
Hasil



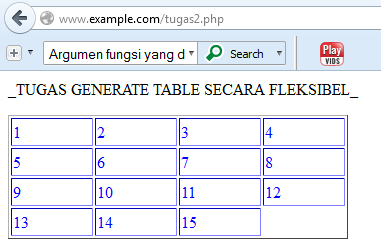
* 1. Buat program sederhana untuk men-*generate* sel tabel secara fleksibel. Tekniknya, buat sebuah fungsi yang menerima argumen berupa jumlah sel dan jumlah kolom. Jadi, pembentukan sel tabel didasarkan pada nilai jumlah sel dan jumlah kolom yang diberikan. Sebagai contoh, hasil program dengan argumen 12 (sel) dan 3 (kolom) diperlihatkan seperti gambar dibawah ini.



Untuk contoh argumen 10 dan 4 diperlihatkan pada gambar dibawah ini



|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html xmlns="http://www.w3.org/1999/xhtml">  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />  <title>Tabel Generat- Tugas Praktikum 2</title>  <style type="text/css">  #apDiv1 {  width:78px;  height:24px;  z-index:1;  left: 284px;  top: 189px;  color: #0000ff;  }    .style1 {  font-weight: bold;  }  </style>  <body>  \_TUGAS GENERATE TABLE SECARA FLEKSIBEL\_  <br />  <p></p>  <?php  $cell=15;  $kolom=4;  function generator($jum\_cell, $k)  {  echo "<table border='1'>";  $x=1;  for ($i=1; $i <=ceil($jum\_cell/$k); $i++) {  echo "<tr>";  for ($j=1; $j <=$k ; $j++) {  if ($x<=$jum\_cell) {  echo"<td id='apDiv1'>".$x."</td>";  }  else  {  echo " ";  }    $x++;  }  // $x=$x+$k;  echo "</tr>";  }  echo "</table>";  }  generator($cell, $kolom);  ?>  </body>  </html> |



1. **Kesimpulan**

* PHP *(Hypertext Preprocessor)* merupakan salah satu bahasa *web-scripting* yang sangat *powerfull*.
* PHP mendukung delapan tipe primitif, yang terdiri dari empat tipe skalar *(booloean, integer, float / double,* dan *string*), dua tipe gabungan (array dan object), dan sisanya adalah tipe khusus (*NULL* dan resource).
* PHP mendukung tiga jenis sintaks untuk menuliskan baris komentar, meliputi style C, C++, dan Perl/Shell.
* Paeser PHP bekerja dengan menguraikan instruksi – instruksi yang diletakkan di antar tag pembuka dan penutup.

1. **Daftar Rujukan**

Modul Praktikum Pemrograman Web, Modul 1 : *Dasar-dasar PHP*. Jurusan Teknik Elektro, Fakultas Teknik, Universitas Negeri Malang.