

Pemrograman Web dengan PHP dan MySQL

Akhmad Sofwan

sofwan@sofwan.net

http://www.sofwan.net

Lisensi Dokumen:

Copyright © 2003-2016 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarluaskan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Pemrograman adalah sebuah pekerjaan menulis kode program untuk menghasilkan sebuah aplikasi atau software. Salah satu platform aplikasi adalah Web, selain desktop dan Mobile. Pemrograman di dalam platform Web sering disebut Pemrograman Web. Di dalam pemrograman web, terdapat bahasa pemrograman dan script yang dapat dipergunakan. HTML dan CSS adalah script untuk membuat tampilan halaman web. JavaScript adalah bahasa pemrograman yang biasa dipergunakan untuk membuat halaman web lebih interaktif. JavaScript dengan node.js nya juga dipergunakan sebagai Back-end. Sebagai Back-End, PHP, Python dan Java juga dapat dipergunakan. Pada tutorial ini, kita akan membahas penggunaan PHP sebagai bahasa pemrograman web, dengan disinggung juga HTML dan CSS sebagai pembuat halaman webnya. Penulis juga menggunakan MySQL sebagai database untuk pegolahan data.

Pendahuluan

Tutorial Pemrograman web ini, penulis gunakan untuk mengasuh kelas Web Programming di Perguruan Tinggi Raharja, Tangerang. Tutorial ini dibagi menjadi 14 bab dan pada bab 11 ke atas, terdapat sebuah proyek kecil PHP yang juga diserta source code. Source code pada tutorial ini disertai dengan nomor baris, untuk identitasnya.

Salah satu cara untuk menginstall PHP dengan Apache, MySQL dan Phpmyadmin, adalah dengan menggunakan Xampp (www.apachefriends.org). PHP versi terkini (Maret 2019) adalah PHP versi 7.3.2. Kita juga perlu menggunakan Text Editor untuk menulis kode PHP. Anda dapat menggunakan geany atau Notepad++ sebagai text editor PHP, disamping sejumlah Text Editor lainnya. Anda dapat juga menggunakan MariaDB yang memiliki kemiripan tinggi dengan MySQL dan pada versi terkini dari XAMPP, yang dulunya menggunakan MySQL, saat ini sudah menggunakan MariaDB. MariaDB sendiri adalah database open source yang dibuat oleh Developer MySQL. MySQL memiliki versi komersial dan community, namun MariaDB hanya memiliki lisensi open source.

BAB I

Pengertian Dasar Pemrograman Web dan PHP

I.1.Apa itu Pemrograman Web

Pemrograman Web adalah pemrograman komputer yang berjalan dengan menggunakan browser. Pemrograman komputer di lakukan dengan penerapan ilmu-ilmu komputer tertentu, seperti Algoritma dan Struktur data yang di terapkan di dalam sebuah bahasa pemrograman tertentu. Untuk Web programming, ada beberapa bahasa pemrograman yang dapat di gunakan, seperti : PHP, JSP, ASP dan Ruby. Di dalam kesempatan ini, kita akan menggunakan PHP untuk dalam Pemrograman Web.

I.2 Pengenalan PHP

PHP (PHP:Hypertext Preprocessor) adalah sebuah bahasa pemrograman yang umum di gunakan untuk pengembangan web yang di jalankan dalam sebuah browser dan di terjemahkan oleh Web Server. Web Server adalah sebuah perangkat keras atau perangkat lunak yang menyediakan layanan akses kepada pengguna melalui protokol komunikasi HTTP atau HTTPS atas file-file yang terdapat pada suatu situs web.

PHP di dalam penggunaan nya untuk pembuatan web, juga bekerja sama dengan html,css dan javascript serta database dalam proses pembuatan nya, yang peran PHP sendiri adalah untuk pemrograman di sisi server, yaitu pemrograman yang di eksesksi script nya oleh web server, sementara untuk design, di gunakan html,css,java script dan grafis.

I.3 Menggunakan PHP

Untuk menggunakan PHP, kita perlu menginstall Web Server dengan feature dapat mengakses PHP. Beberapa Web Server yang di gunakan oleh PHP, yaitu : Apache, IIS,Xitami dan Nginx. Untuk menggunakan database, kita dapat menggunakan sejumlah database yang dapat bekerja sama dengan PHP, beberapa di antara nya adalah : MySQL,Postgre SQL dan SQL Server. Kita juga dapat menggunakan software bundle instalasi PHP dan lingkungan nya, yaitu Xampp, yang di dalam nya langsung terinstall dan terkonfigurasi apache, mysql , proftpd , phpmyadmin,dll.

I.4 Pengenalan MySQL

MySQL adalah sebuah Database Open Source populer yang versi terbaru saat ini untuk versi Mysql Community Server adalah Mysql Community Server 5.1.49. Database ini dapat berjalan di beberapa platform atau Sistem Operasi, seperti Windows, Mac OSX, Linux dan OpenSolaris. Kita dapat men download MySQL di <http://dev.mysql.com/downloads>. Fungsi database sendiri adalah untuk menyimpan data.

I.5 Software dan script lainnya

Selain PHP dan Mysql, terdapat beberapa aplikasi dan scripting yang di butuhkan di dalam pemrograman PHP ini. Kita butuh Web Server dan yang akan kita gunakan adalah Web Server Apache. Web Server sendiri adalah sebuah program komputer untuk menjalan kan sebuah konten seperti halaman web, menggunakan Hyper Text Transfer Protocol (HTTP) di dalam World Wide Web atau Internet¹.

Di dalam pemrograman PHP, kita juga membutuhkan script lain selain PHP itu sendiri, yaitu HTML/XHTML untuk menampilkan halaman web, CSS untuk design dan Java Script untuk pemrograman berbasis client atau browser.

I.6 Penggunaan XAMPP

Untuk menjalankan script PHP dengan Mysql, kita membutuhkan aplikasi PHP dan Mysql serta web server telah terinstall di komputer kita. Kita dapat menginstall nya masing-masing aplikasi tersebut satu persatu secara terpisah, namun ada sebuah paket aplikasi free yang memuat, PHP, Apache, Mysql, PHPMyadmin,dll yang dapat dengan mudah kita install. Aplikasi paket tersebut adalah Xampp, yang dapat di download gratis di <http://www.apachefriends.org> .

I.7 PHP Editor

Salah satu yang dapat memudahkan kita di dalam pemrograman adalah pemilihan Editor atau Integrated Development Environment (IDE) yang baik dan tepat.

Beda editor dan IDE adalah, bahwa Editor adalah software untuk menulis dan mengedit kode kita, seperti Notepad di Windows dan Vim di Linux, sedangkan IDE adalah editor yang lebih canggih lagi, karena sudah dilengkapi berbagai feature, seperti Debugging, Code Highlighting dan pemunculan symbol class dan feature.

Beberapa Editor atau IDE yang bagus dan free untuk Windows adalah :

1. Notepad++ (<http://notepad-plus-plus.org>)
2. Crimson (<http://www.crimsoneditor.com>)
3. Html-Kit (<http://www.htmlkit.com>)

Beberapa Untuk Linux adalah :

1. gPHPEdit (<http://www.gphpedit.org>)
2. Quanta Plus (<http://quanta.kdewebdev.org>)
3. Kdevelop (www.kdevelop.org)

Beberapa untuk Linux dan Windows adalah :

1. Bluefish (<http://bluefish.openoffice.nl>)
2. Eclipse (www.eclipse.org)
3. Netbeans IDE (www.netbeans.org)
4. Geany (www.Geany.org)

Di samping itu juga terdapat php editor commercial.

I.8 Web Browser

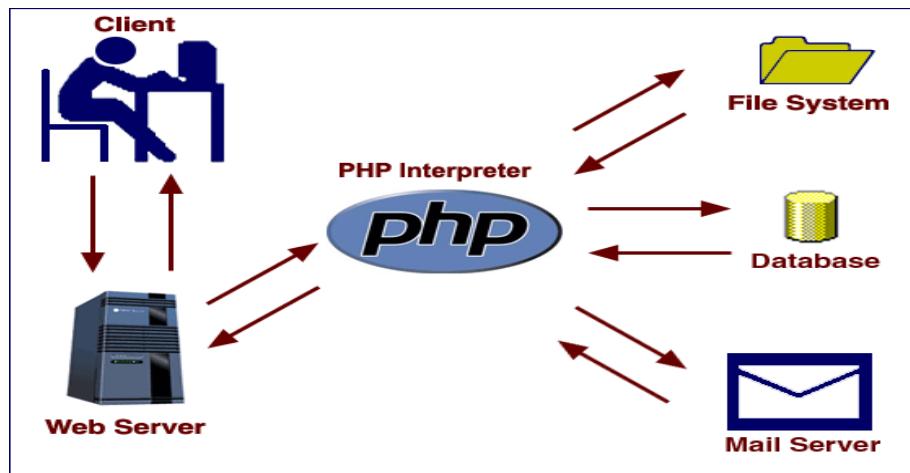
PHP berjalan di atas web server pada sebuah web browser. PHP dapat berjalan di semua web browser modern, seperti Internet Explorer, Mozilla Firefox, Opera, Konqueror dan Chrome. Jika Xampp kita telah berjalan dengan baik, dengan melakukan pengetikan <http://localhost> pada link address kita, maka browser dan xampp telah siap di gunakan.

Selanjut nya, kita cukup menaruh file-file php kita di dalam folder htdocs pada folder xampp. Untuk Windows pada folder <c:/xampp/htdocs> dan untuk Linux /opt/lampp/htdocs.

I.9 How PHP Works ?

Di dalam gambar tsb, client meminta kepada web server via browser sebuah page .php. Page tersebut di interpretasi kan oleh PHP Interpreter. PHP Interpreter, dapat merequest Database, File System dan Mail Server.

Untuk menggunakan PHP, anda harus mengenal HTML dan lebih utama, juga mengenal CSS. PHP digunakan , sebagai embedded language (bahasa yang disisipkan) di dalam HTML atau kebalikannya, HTML adalah script embedded pada script PHP. Ekstension PHP adalah .php.



Gambar I.1 : How PHP works [5]

Syntax dasar

Contoh :

1. Mencetak kata Hello World

```

1.<?php
2. echo "Hello World";
3.?>

```

2. Mencetak kata Hello World di dalam halaman web

```

1. <!doctype html>
2. <html>
3.   <head>
4.     <title>PHP Pertama ku</title>
5.   </head>
6.   <body>
7.     <?php echo "Hello World";
8.   </body>
9. </html>

```

3. Mengetahui konfigurasi php environment

```

1. <!doctype html>
2. <html>
3.   <head>
4.     <title>PHP Info</title>
5.   </head>
6.   <body>
7.     <?php echo phpinfo()
8.   </body>
9. </html>

```

BAB II

Dasar-dasar PHP

II.1 Pendahuluan

Dasar-dasar PHP yang akan di bahas pada buku ini adalah berupa syntax dasar, tipe data dan algoritma, yang di butuhkan di dalam pemrograman database.

II.2 Hello World

Biasanya, kita diminta menulis “Hello World” untuk awal mula belajar sebuah bahasa pemrograman. Demikian juga untuk kali ini, kita coba membuat kata “Hello World”

1. <?php
2. echo “Hello World”;
3. ?>

II.3 Tag awal dan penulisan PHP

Setelah kita berhasil menginstall XAMPP dan PHP Editor, maka kita sudah siap untuk mulai menulis Kode PHP kita. Kode PHP di awali dan di tutup dengan <?php ?> atau <? ?> atau <script language=“php”></script>.

Tetapi disarankan untuk menggunakan tag <?php ?>. Tag <? ?> kurang disarankan, karena dapat mirip pada tag script lain, sehingga dapat membuat web server kebingungan dalam mengakses script php, jika digabung dengan script lain yang menggunakan tag sejenis, seperti XML.

Di samping itu penggunaan tag <? ?> mengharuskan short_open_tag=On di php.ini.

II.4 Penggunaan echo dan print

Untuk menampilkan output, kita dapat menggunakan syntax **echo** dan **Print**.

Contoh : 1. <?php

2. echo “Belajar Pemrograman PHP
”;
3. echo (“Belajar Pemrograman PHP
”);
4. print “Belajar Pemrograman PHP
”;
5. print (“Belajar Pemrograman PHP
”);
6. ?>

Kedua fungsi tsb dapat digunakan dengan atau tanpa “()”. Syntax
 adalah syntax xhtml untuk turun ke bawah 1 baris.

II.5 Memberi Comment

Dengan : // atau /* * /

Contoh :

- 1.<?php
2. echo “Nama saya Sofwan
”; // Baris pertama
3. echo “Tinggal di Jakarta ”;
4. /* Syntax di atas menggunakan
5. echo dua-dua nya.
6. */
7. ?>

II.6 Insert Syntax PHP di dalam HTML

Seringkali di perlukan untuk insert code php di dalam html atau sebalik nya.

Contoh Insert PHP di dalam HTML:

```
1.<html>
2.  <head>
3.      <title>Contoh Program PHP </title>
4.  </head>
5.  <body>
6.      <p>Nama <?php echo "Andi" ;?> </p>
7.  </body>
8.</html>
```

Contoh insert html di dalam php.

```
1.<?php
2. // Membuat kata tebal pada sebuah kalimat
3. echo "<b>Belajar Database PHP</b>";
4. ?>
```

II.7 Tipe Data

Di setiap bahasa pemrograman di kenal ada nya tipe data, termasuk juga di PHP. Tipe data adalah jenis data yang di gunakan oleh PHP untuk menyimpan nilai di dalam sebuah variabel. Untuk menggunakan tipe data, di dalam PHP tidak di perlukan statement tipe data tertentu di depan variabel.

Tipe data yang ada di PHP adalah :

1. Booleans

Tipe data yang bernilai **True** atau **False** dan penulisan nilai nya dapat berupa case-insensitive.

Contoh :

```
1.<?php
2. $nilai1 = True ;
3. $nilai2 = False
4. ?>
```

2. Integer

Tipe data yang dapat di tulis dalam decimal (base 10), hexadesimal (base 16) atau octal (base 8). Untuk nilai hexadesimal di awalai angka : 0x dan octal di awali angka : 0

Contoh :

```
1.<?php
2. $nilai1=15; // Memberi nilai 15
3. $nilai2= -20; // Memberi nilai -20
4. $nilai3= 0123; // angka octal (sama dengan 83)
5. $nilai4= 0x1A; // angka hexadecimal (sama dengan 26)
6. ?>
```

Ukuran maximal sekitar 2 milyar (32 bit)

3. Float

Tipe float adalah tipe yang dapat menampung nilai di belakang koma.

Contoh :

```
1.<?php
2. $nilai1= 2.56;
3. $nilai2=1.2e3;
4. ?>
```

4. String

Kita dapat menggunakan koma satu dan koma dua :

- Koma satu

Contoh :

```

1.<?php
2. $kata1='Saya sedang belajar PHP';
3. $kata2='Nama saya Isma\'il'; // Cara menulis kata yang ada koma
4. //di atas nya
5. echo "Kata 1 : $kata1";
6. echo "<br />"; // Baris ke bawah
7. echo "Kata 2 : $kata2";
8. ?>

```

- Koma double

dengan menggunakan koma double, kita dapat menggunakan karakter tertentu, yaitu :

Sequence	Arti
\n	Baris ke bawah
\r	Mengembalikan carriage
\t	Tab Horizontal
\f	1 baris form
\\\	Backslash
\\$	Memberikan tanda dollar
\”	Koma double

Tabel : II.1 . Arti Sequence

Contoh :

```

1.<?php
2. $kata1 = "Saya memiliki uang \$ 15";
3. $kata2 = "Ibukota Indonesia adalah \" Jakarta \"";
4. echo "kata 1 : $kata1 <br />";
5. echo "kata 2 : $kata2";
6. ?>

```

Batas maksimal memory string, adalah penggunaan memory pada PHP.

5. Array

Array adalah sebuah variabel yang menyimpan banyak nilai di dalamnya secara teratur, yang setiap nilai memiliki alamat masing-masing secara unique.

Syntax : array().

- Array 1 dimensi

Adalah array biasa yang terdiri dari 1 dimensi atau daftar nilai.

Untuk mencetak sebuah nilai pada array di gunakan :

\$nama_array[alamat]

Contoh :

```

1.<?php
2. $hari = array ("Senin","Selasa","Rabu","Kamis","Jum'at",
3.                 "Sabtu","Minggu");
4. echo "Hari ini adalah hari : $hari [0]"
5. ?>

```

Awal alamat array dapat di ubah, seperti :

```
1. <?php
2. $hari = array (1=>"Senin","Selasa","Rabu","Kamis","Jum'at",
3.                 "Sabtu","Minggu");
4. print_r($hari); // Mencetak isi array $hari
5. echo "Hari ini adalah hari : $hari[1]"; // Hari senin
6. ?>
```

- Array 2 dimensi

Adalah sebuah nama array yang dapat menyimpan array lain di bawah nya.

Contoh :

```
1.<?php
2. $arr2=array("kelas_5"=>array(1=>"Andi","Umar","Ihsan"),
3.             "kelas_6"=>array (1=>"Nia","Siti","Amir"));
4. $data1=$arr2["kelas_5"][2]; // Umar
5. $data2=$arr2["kelas_6"][3]; // Amir
6. echo "Data 1 : $data1 <br/>";
7. echo "Data 2 : $data2 <br/>";
8. ?>
```

6. Object

Untuk membuat object baru, di gunakan syntax new.

Contoh :

```
1. <?php
2. class coba
3.     {function test()
4.      {
5.          echo "Testing ... testing";
6.      }
7.     }
8. $akses= new coba;
9. $akses → test();
10. ?>
```

7. NULL

NULL adalah type data yang berarti tidak mengandung nilai.

```
1. <?php
2. $nilai1=0;
3. $nilai2=null;
4. echo "Nilai 1 : $nilai1";
5. echo "Nilai 2 : $nilai2";
6. ?>
```

\$nilai1 mengandung nilai 0 dan \$nilai2 tidak ada nilai atau null.

Suatu variabel yang belum di beri nilai dapat di katakan, bahwa variabel tersebut tidak ada nilai atau null.

II.8 Variabel

Variabel adalah sesuatu yang di pergunakan untuk menyimpan sebuah nilai. Pemberian nama variabel di awali dengan tanda \$, lalu dapat berupa huruf atau under score dan tidak dapat dengan angka.

Contoh :

1. <?php
2. \$nama="Anwar"; // Tipe variabel string
3. \$angka=4; // Tipe variabel numerik
4. \$benar=true;
5. ?>

Variabel dapat menentukan sendiri tipe nya masing-masing secara otomatis tatkala di beri sebuah nilai, apakah berbentuk string, numerik atau float.

II.9 Operator

Operator di gunakan untuk menjalankan proses aritmatika, logika, perbandingan, string, array, type , incrementing dan decrementing dan lainnya.

1. Operator Aritmetika dan Sama dengan

Contoh	Deskripsi	Keterangan
-\$a	Negasi	Lawan dari \$a
\$a + \$b	Penjumlahan	Penjumlahan variabel \$a terhadap \$b
\$a - \$b	Pengurangan	Pengurangan \$a terhadap \$b
\$a * \$b	Perkalian	Perkalian \$a terhadap \$b
\$a / \$b	Pembagian	Pembagian \$a terhadap \$b
\$a % \$b	Modulus	Sisa pembagian \$a terhadap \$b
5 + 6 = 11 dan \$a = 5	Sama dengan	Hasil dari operasi Aritmetika atau pemberian nilai terhadap suatu variabel

Tabel II.2 : Operator Aritmetika dan Sama Dengan

2. Operator Incrementing dan Decrementing

Contoh	Deskripsi	Keterangan
++\$a	Pre-Increment	Menambah \$a dengan 1 dan mengembalikan nilai a
\$a++	Post-Increment	Mengembalikan \$a , kemudian menambah dengan 1
--\$a	Pre-decrement	Mengurangi \$a dengan 1 kemudian mengembalikan \$a
\$a -	Post-decrement	Mengembalikan \$a, kemudian mengurangi \$a dengan 1

Tabel II.3 : Operator Incrementing dan Decrementing

Contoh :

1. <?php
2. \$a=4;
3. echo \$a++ ;
4. echo \$a;
- 5.
6. \$a=4;
7. echo ++\$a;
8. echo \$a;

```

9. $a=4;
10. echo $a--;
11. echo $a;
12.
13. $a=4;
14. echo -$a;
15. echo $a;
16. ?>

```

3. Operator Logika

Contoh	Deskripsi	Hasil
\$a and \$b / \$a && \$b	And	Benar , jika \$a dan \$b benar
\$a or \$b / \$a \$b	Or	Benar, jika \$a atau \$b benar
\$a xor \$b	Xor	Benar, jika \$a atau \$b benar, tetapi tidak keduanya.
!\$a	Not	Benar, jika \$a tidak benar

Tabel II.4 : Operator Logika

Operator logika ini sering di gunakan bersamaan dengan feature **if .. endif**

Contoh :

```

1. <?php
2. $a=5;
3. $b=10;
4. if ($a < 7 && $b<7)
5.     { echo "Kondisi Benar"; }
6. else
7.     { echo "Kondisi salah";}
8.
9. if ($a < 7 || $a < 7)
10.    { echo "Kondisi Benar"; }
11. else
12.    { echo "Kondisi Salah" ; }
13. ?>

```

4. Operator String

Operator string memiliki 2 operator, yaitu . (titik) dan .= (titik sama dengan). Kedua nya berfungsi untuk menggabungkan 2 atau lebih string.

Contoh :

```

1.<?php
2. $a = "Belajar";
3. $a=$a." PHP";
4. echo $a; // Belajar PHP
5. echo "<br />";
6. $a .=" PHP";
7. echo $a; // Belajar PHP
8.?>

```

5. Operator Perbandingan

Untuk membandingkan 2 buah nilai atau variabel. Kita dapat menggunakan operator perbandingan, yang terdiri dari :

Contoh	Deskripsi	Hasil
<code>\$a == \$b</code>	Sama dengan	Benar, jika \$a sama dengan \$b
<code>\$a === \$b</code>	Identik	Benar, jika \$a sama dengan \$b dan kedua nya bertipe sama
<code>\$a != \$b</code> atau <code>\$a <> \$b</code>	Tidak sama dengan	Benar, jika \$a tidak sama dengan \$b
<code>\$a !== \$b</code>	Tidak identik	Benar, jika \$a tidak sama dengan \$b atau kedua nya tidak identik
<code>\$a < \$b</code>	Kurang dari	Benar, jika \$a kurang dari \$b
<code>\$a > \$b</code>	Lebih besar dari	Benar, jika \$a lebih besar dari \$b
<code>\$a <= \$b</code>	Lebih kecil atau sama dengan	Benar, jika \$a lebih kecil atau sama dengan \$b.
<code>\$a >= \$b</code>	Lebih besar atau sama dengan	Benar, jika \$a lebih besar atau sama dengan \$b.

Tabel II.5 : Operator Perbandingan

Contoh :

```
1.<?php
2. $a=6;
3. $b=8;
4.
5. if ($a == $b)
6. { echo "Kedua variabel sama dengan"; }
7. else
8. { echo "Kedua variabel TIDAK sama dengan"; }
9.
10. if ($a > $b)
11. { echo "$a lebih besar dari $b"; }
12. else
13. { echo "$a tidak lebih besar dari $b"; }
14. ?>
```

BAB III

STRUCTURE CONTROL (1)

Struktur control sangat di perlukan di dalam pemrograman, termasuk web programming, untuk penanganan logika dan algoritma program.

Di bawah ini adalah beberapa Struktur Control yang ada di dalam PHP.

III.1. While

Syntax : While (expr)
 statement

Syntax ini akan menjalankan statement selama expr benar.

Contoh :

```
1.<?php  
2. $angka=5;  
3. while ($angka <=10)  
4. {  
5.     echo "Angka : $angka <br/>";  
6.     $angka++; // $angka=$angka+1;  
7. }  
8.?>
```

Statement mencetak akan di laksanakan selama variabel \$angka masih di bawah atau sama dengan 10.

III.2. Do-while

Syntax : do
 { statement ; }
 while (expr);

Contoh :

```
1. <?php  
2. $nilai=10;  
3. do  
4. { echo "Nilai : $nilai";  
5.     $nilai++;  
6. }  
7. while ($nilai<=10)  
8.?>
```

Statement untuk mencetak Nilai di jalankan terlebih dahulu, kemudian variabel \$nilai di tambah 1, baru setelah itu di lakukan cek, apakah variabel \$nilai sama dengan 10 atau tidak.

Statement akan terus di eksekusi selama kondisi \$nilai <= 10 benar.

III.3. for

Syntax : for (expr1; expr2; expr3)
 { statement; }

expr1 : Statement awal

expr2 : Cek Kondisi

expr3 : Penjumlahan atau pengurangan

Contoh :

```
1. <?php  
2. for ($angka=1;$angka<=10;$angka++)  
3. { echo "Angka : $angka <br />"; }  
4.?>
```

Perintah di atas akan mencetak statement angka sebanyak 10 kali.

III.4. foreach

Syntax : `foreach (array_expression as $value)
{ statement }`
array_expression : nama array
\$value : variabel untuk memuat nilai dari element array yang akan secara otomatis bergerak ke elemen array selanjut nya.

Contoh :

```
1. <?php  
2.     $data=array("Budiman","Iman","Nani","Susi","Umar");  
3.     foreach ($data as $nilai);  
4.         { echo "Nama Siswa : $nilai <br/>"; }  
5. ?>
```

Output :

```
Nama Siswa : Budiman  
Nama Siswa : Iman  
Nama Siswa : Nani  
Nama Siswa : Susi  
Nama siswa : Umar
```

Variabel \$nilai akan memuat elemen array dari elemen array pertama hingga terakhir, pointer array akan bergerak secara otomatis dari element array pertama hingga terakhir.

- Syntax : `foreach (array_expression as $kunci=>$value)
{ statement }`

Feature ini sama dengan foreach sebelum nya, beda nya hanya akan memuat nilai kunci array nya, yang secara default di mulai dari 0.

Contoh :

```
1. <?php  
2.     $data=array("Budiman","Iman","Nani","Susi","Umar");  
3.     foreach ($data as $kunci =>$nilai);  
4.         { echo "Urut :$kunci. Nama Siswa : $nilai <br/>"; }  
5. ?>
```

Output :

```
Urut : 0 Nama Siswa : Budiman  
Urut : 1 Nama Siswa : Iman  
Urut : 2 Nama Siswa : Nani  
Urut : 3 Nama Siswa : Susi  
Urut : 4 Nama siswa : Umar
```

Variabel \$kunci memuat key dari element array dan variabel \$nilai memuat element array nya.

III.5. Switch

Switch berfungsi hampir sama dengan if .. end if. Switch cocok di gunakan saat kita hendak membandingkan sebuah variabel yang sama dengan nilai-nilai yang berbeda dan mengeksekusi statement yang berbeda, yang tergantung dari kondisi nilai nya.

Setiap bagian, kita harus menyertakan break, untuk menghentikan eksekusi yang telah di eksekusi.

Contoh :

```
1. <?php  
2.     $nilai=5;  
3.     switch ($nilai)  
4.         { case 1 :  
5.             echo "Nilai nya 1";  
6.             break;  
7.         case 2 :  
8.             echo "Nilai nya 2";  
9.             break;  
10.        case 3 :  
11.            echo "Nilai nya 3";  
12.            break;  
13.        case 4 :  
14.            echo "Nilai nya 4";  
15.            break;  
16.        case 5 :  
17.            echo "Nilai nya 5";  
18.            break;  
19.        case 6 :  
20.            echo "Nilai nya 6";  
21.            break;  
22.        case 7 :  
23.            echo "Nilai nya 7";  
24.            break;  
25.        case 8 :  
26.            echo "Nilai nya 8";  
27.            break;  
28.        case 9 :  
29.            echo "Nilai nya 9";  
30.            break;  
31.        case 10 :  
32.            echo "Nilai nya 10";  
33.            break;  
34.        case 11 :  
35.            echo "Nilai nya 11";  
36.            break;  
37.        case 12 :  
38.            echo "Nilai nya 12";  
39.            break;  
40.        case 13 :  
41.            echo "Nilai nya 13";  
42.            break;  
43.        case 14 :  
44.            echo "Nilai nya 14";  
45.            break;  
46.        case 15 :  
47.            echo "Nilai nya 15";  
48.            break;  
49.        case 16 :  
50.            echo "Nilai nya 16";  
51.            break;  
52.        case 17 :  
53.            echo "Nilai nya 17";  
54.            break;  
55.        case 18 :  
56.            echo "Nilai nya 18";  
57.            break;  
58.        case 19 :  
59.            echo "Nilai nya 19";  
60.            break;  
61.        case 20 :  
62.            echo "Nilai nya 20";  
63.            break;  
64.        case 21 :  
65.            echo "Nilai nya 21";  
66.            break;  
67.        case 22 :  
68.            echo "Nilai nya 22";  
69.            break;  
70.        case 23 :  
71.            echo "Nilai nya 23";  
72.            break;  
73.        case 24 :  
74.            echo "Nilai nya 24";  
75.            break;  
76.        case 25 :  
77.            echo "Nilai nya 25";  
78.            break;  
79.        case 26 :  
80.            echo "Nilai nya 26";  
81.            break;  
82.        case 27 :  
83.            echo "Nilai nya 27";  
84.            break;  
85.        case 28 :  
86.            echo "Nilai nya 28";  
87.            break;  
88.        case 29 :  
89.            echo "Nilai nya 29";  
90.            break;  
91.        case 30 :  
92.            echo "Nilai nya 30";  
93.            break;  
94.        case 31 :  
95.            echo "Nilai nya 31";  
96.            break;  
97.        case 32 :  
98.            echo "Nilai nya 32";  
99.            break;  
100.       case 33 :  
101.          echo "Nilai nya 33";  
102.          break;  
103.       case 34 :  
104.          echo "Nilai nya 34";  
105.          break;  
106.       case 35 :  
107.          echo "Nilai nya 35";  
108.          break;  
109.       case 36 :  
110.          echo "Nilai nya 36";  
111.          break;  
112.       case 37 :  
113.          echo "Nilai nya 37";  
114.          break;  
115.       case 38 :  
116.          echo "Nilai nya 38";  
117.          break;  
118.       case 39 :  
119.          echo "Nilai nya 39";  
120.          break;  
121.       case 40 :  
122.          echo "Nilai nya 40";  
123.          break;  
124.       case 41 :  
125.          echo "Nilai nya 41";  
126.          break;  
127.       case 42 :  
128.          echo "Nilai nya 42";  
129.          break;  
130.       case 43 :  
131.          echo "Nilai nya 43";  
132.          break;  
133.       case 44 :  
134.          echo "Nilai nya 44";  
135.          break;  
136.       case 45 :  
137.          echo "Nilai nya 45";  
138.          break;  
139.       case 46 :  
140.          echo "Nilai nya 46";  
141.          break;  
142.       case 47 :  
143.          echo "Nilai nya 47";  
144.          break;  
145.       case 48 :  
146.          echo "Nilai nya 48";  
147.          break;  
148.       case 49 :  
149.          echo "Nilai nya 49";  
150.          break;  
151.       case 50 :  
152.          echo "Nilai nya 50";  
153.          break;  
154.       case 51 :  
155.          echo "Nilai nya 51";  
156.          break;  
157.       case 52 :  
158.          echo "Nilai nya 52";  
159.          break;  
160.       case 53 :  
161.          echo "Nilai nya 53";  
162.          break;  
163.       case 54 :  
164.          echo "Nilai nya 54";  
165.          break;  
166.       case 55 :  
167.          echo "Nilai nya 55";  
168.          break;  
169.       case 56 :  
170.          echo "Nilai nya 56";  
171.          break;  
172.       case 57 :  
173.          echo "Nilai nya 57";  
174.          break;  
175.       case 58 :  
176.          echo "Nilai nya 58";  
177.          break;  
178.       case 59 :  
179.          echo "Nilai nya 59";  
180.          break;  
181.       case 60 :  
182.          echo "Nilai nya 60";  
183.          break;  
184.       case 61 :  
185.          echo "Nilai nya 61";  
186.          break;  
187.       case 62 :  
188.          echo "Nilai nya 62";  
189.          break;  
190.       case 63 :  
191.          echo "Nilai nya 63";  
192.          break;  
193.       case 64 :  
194.          echo "Nilai nya 64";  
195.          break;  
196.       case 65 :  
197.          echo "Nilai nya 65";  
198.          break;  
199.       case 66 :  
200.          echo "Nilai nya 66";  
201.          break;  
202.       case 67 :  
203.          echo "Nilai nya 67";  
204.          break;  
205.       case 68 :  
206.          echo "Nilai nya 68";  
207.          break;  
208.       case 69 :  
209.          echo "Nilai nya 69";  
210.          break;  
211.       case 70 :  
212.          echo "Nilai nya 70";  
213.          break;  
214.       case 71 :  
215.          echo "Nilai nya 71";  
216.          break;  
217.       case 72 :  
218.          echo "Nilai nya 72";  
219.          break;  
220.       case 73 :  
221.          echo "Nilai nya 73";  
222.          break;  
223.       case 74 :  
224.          echo "Nilai nya 74";  
225.          break;  
226.       case 75 :  
227.          echo "Nilai nya 75";  
228.          break;  
229.       case 76 :  
230.          echo "Nilai nya 76";  
231.          break;  
232.       case 77 :  
233.          echo "Nilai nya 77";  
234.          break;  
235.       case 78 :  
236.          echo "Nilai nya 78";  
237.          break;  
238.       case 79 :  
239.          echo "Nilai nya 79";  
240.          break;  
241.       case 80 :  
242.          echo "Nilai nya 80";  
243.          break;  
244.       case 81 :  
245.          echo "Nilai nya 81";  
246.          break;  
247.       case 82 :  
248.          echo "Nilai nya 82";  
249.          break;  
250.       case 83 :  
251.          echo "Nilai nya 83";  
252.          break;  
253.       case 84 :  
254.          echo "Nilai nya 84";  
255.          break;  
256.       case 85 :  
257.          echo "Nilai nya 85";  
258.          break;  
259.       case 86 :  
260.          echo "Nilai nya 86";  
261.          break;  
262.       case 87 :  
263.          echo "Nilai nya 87";  
264.          break;  
265.       case 88 :  
266.          echo "Nilai nya 88";  
267.          break;  
268.       case 89 :  
269.          echo "Nilai nya 89";  
270.          break;  
271.       case 90 :  
272.          echo "Nilai nya 90";  
273.          break;  
274.       case 91 :  
275.          echo "Nilai nya 91";  
276.          break;  
277.       case 92 :  
278.          echo "Nilai nya 92";  
279.          break;  
280.       case 93 :  
281.          echo "Nilai nya 93";  
282.          break;  
283.       case 94 :  
284.          echo "Nilai nya 94";  
285.          break;  
286.       case 95 :  
287.          echo "Nilai nya 95";  
288.          break;  
289.       case 96 :  
290.          echo "Nilai nya 96";  
291.          break;  
292.       case 97 :  
293.          echo "Nilai nya 97";  
294.          break;  
295.       case 98 :  
296.          echo "Nilai nya 98";  
297.          break;  
298.       case 99 :  
299.          echo "Nilai nya 99";  
300.          break;  
301.       case 100 :  
302.          echo "Nilai nya 100";  
303.          break;  
304.       case 101 :  
305.          echo "Nilai nya 101";  
306.          break;  
307.       case 102 :  
308.          echo "Nilai nya 102";  
309.          break;  
310.       case 103 :  
311.          echo "Nilai nya 103";  
312.          break;  
313.       case 104 :  
314.          echo "Nilai nya 104";  
315.          break;  
316.       case 105 :  
317.          echo "Nilai nya 105";  
318.          break;  
319.       case 106 :  
320.          echo "Nilai nya 106";  
321.          break;  
322.       case 107 :  
323.          echo "Nilai nya 107";  
324.          break;  
325.       case 108 :  
326.          echo "Nilai nya 108";  
327.          break;  
328.       case 109 :  
329.          echo "Nilai nya 109";  
330.          break;  
331.       case 110 :  
332.          echo "Nilai nya 110";  
333.          break;  
334.       case 111 :  
335.          echo "Nilai nya 111";  
336.          break;  
337.       case 112 :  
338.          echo "Nilai nya 112";  
339.          break;  
340.       case 113 :  
341.          echo "Nilai nya 113";  
342.          break;  
343.       case 114 :  
344.          echo "Nilai nya 114";  
345.          break;  
346.       case 115 :  
347.          echo "Nilai nya 115";  
348.          break;  
349.       case 116 :  
350.          echo "Nilai nya 116";  
351.          break;  
352.       case 117 :  
353.          echo "Nilai nya 117";  
354.          break;  
355.       case 118 :  
356.          echo "Nilai nya 118";  
357.          break;  
358.       case 119 :  
359.          echo "Nilai nya 119";  
360.          break;  
361.       case 120 :  
362.          echo "Nilai nya 120";  
363.          break;  
364.       case 121 :  
365.          echo "Nilai nya 121";  
366.          break;  
367.       case 122 :  
368.          echo "Nilai nya 122";  
369.          break;  
370.       case 123 :  
371.          echo "Nilai nya 123";  
372.          break;  
373.       case 124 :  
374.          echo "Nilai nya 124";  
375.          break;  
376.       case 125 :  
377.          echo "Nilai nya 125";  
378.          break;  
379.       case 126 :  
380.          echo "Nilai nya 126";  
381.          break;  
382.       case 127 :  
383.          echo "Nilai nya 127";  
384.          break;  
385.       case 128 :  
386.          echo "Nilai nya 128";  
387.          break;  
388.       case 129 :  
389.          echo "Nilai nya 129";  
390.          break;  
391.       case 130 :  
392.          echo "Nilai nya 130";  
393.          break;  
394.       case 131 :  
395.          echo "Nilai nya 131";  
396.          break;  
397.       case 132 :  
398.          echo "Nilai nya 132";  
399.          break;  
400.       case 133 :  
401.          echo "Nilai nya 133";  
402.          break;  
403.       case 134 :  
404.          echo "Nilai nya 134";  
405.          break;  
406.       case 135 :  
407.          echo "Nilai nya 135";  
408.          break;  
409.       case 136 :  
410.          echo "Nilai nya 136";  
411.          break;  
412.       case 137 :  
413.          echo "Nilai nya 137";  
414.          break;  
415.       case 138 :  
416.          echo "Nilai nya 138";  
417.          break;  
418.       case 139 :  
419.          echo "Nilai nya 139";  
420.          break;  
421.       case 140 :  
422.          echo "Nilai nya 140";  
423.          break;  
424.       case 141 :  
425.          echo "Nilai nya 141";  
426.          break;  
427.       case 142 :  
428.          echo "Nilai nya 142";  
429.          break;  
430.       case 143 :  
431.          echo "Nilai nya 143";  
432.          break;  
433.       case 144 :  
434.          echo "Nilai nya 144";  
435.          break;  
436.       case 145 :  
437.          echo "Nilai nya 145";  
438.          break;  
439.       case 146 :  
440.          echo "Nilai nya 146";  
441.          break;  
442.       case 147 :  
443.          echo "Nilai nya 147";  
444.          break;  
445.       case 148 :  
446.          echo "Nilai nya 148";  
447.          break;  
448.       case 149 :  
449.          echo "Nilai nya 149";  
450.          break;  
451.       case 150 :  
452.          echo "Nilai nya 150";  
453.          break;  
454.       case 151 :  
455.          echo "Nilai nya 151";  
456.          break;  
457.       case 152 :  
458.          echo "Nilai nya 152";  
459.          break;  
460.       case 153 :  
461.          echo "Nilai nya 153";  
462.          break;  
463.       case 154 :  
464.          echo "Nilai nya 154";  
465.          break;  
466.       case 155 :  
467.          echo "Nilai nya 155";  
468.          break;  
469.       case 156 :  
470.          echo "Nilai nya 156";  
471.          break;  
472.       case 157 :  
473.          echo "Nilai nya 157";  
474.          break;  
475.       case 158 :  
476.          echo "Nilai nya 158";  
477.          break;  
478.       case 159 :  
479.          echo "Nilai nya 159";  
480.          break;  
481.       case 160 :  
482.          echo "Nilai nya 160";  
483.          break;  
484.       case 161 :  
485.          echo "Nilai nya 161";  
486.          break;  
487.       case 162 :  
488.          echo "Nilai nya 162";  
489.          break;  
490.       case 163 :  
491.          echo "Nilai nya 163";  
492.          break;  
493.       case 164 :  
494.          echo "Nilai nya 164";  
495.          break;  
496.       case 165 :  
497.          echo "Nilai nya 165";  
498.          break;  
499.       case 166 :  
500.          echo "Nilai nya 166";  
501.          break;  
502.       case 167 :  
503.          echo "Nilai nya 167";  
504.          break;  
505.       case 168 :  
506.          echo "Nilai nya 168";  
507.          break;  
508.       case 169 :  
509.          echo "Nilai nya 169";  
510.          break;  
511.       case 170 :  
512.          echo "Nilai nya 170";  
513.          break;  
514.       case 171 :  
515.          echo "Nilai nya 171";  
516.          break;  
517.       case 172 :  
518.          echo "Nilai nya 172";  
519.          break;  
520.       case 173 :  
521.          echo "Nilai nya 173";  
522.          break;  
523.       case 174 :  
524.          echo "Nilai nya 174";  
525.          break;  
526.       case 175 :  
527.          echo "Nilai nya 175";  
528.          break;  
529.       case 176 :  
530.          echo "Nilai nya 176";  
531.          break;  
532.       case 177 :  
533.          echo "Nilai nya 177";  
534.          break;  
535.       case 178 :  
536.          echo "Nilai nya 178";  
537.          break;  
538.       case 179 :  
539.          echo "Nilai nya 179";  
540.          break;  
541.       case 180 :  
542.          echo "Nilai nya 180";  
543.          break;  
544.       case 181 :  
545.          echo "Nilai nya 181";  
546.          break;  
547.       case 182 :  
548.          echo "Nilai nya 182";  
549.          break;  
550.       case 183 :  
551.          echo "Nilai nya 183";  
552.          break;  
553.       case 184 :  
554.          echo "Nilai nya 184";  
555.          break;  
556.       case 185 :  
557.          echo "Nilai nya 185";  
558.          break;  
559.       case 186 :  
560.          echo "Nilai nya 186";  
561.          break;  
562.       case 187 :  
563.          echo "Nilai nya 187";  
564.          break;  
565.       case 188 :  
566.          echo "Nilai nya 188";  
567.          break;  
568.       case 189 :  
569.          echo "Nilai nya 189";  
570.          break;  
571.       case 190 :  
572.          echo "Nilai nya 190";  
573.          break;  
574.       case 191 :  
575.          echo "Nilai nya 191";  
576.          break;  
577.       case 192 :  
578.          echo "Nilai nya 192";  
579.          break;  
580.       case 193 :  
581.          echo "Nilai nya 193";  
582.          break;  
583.       case 194 :  
584.          echo "Nilai nya 194";  
585.          break;  
586.       case 195 :  
587.          echo "Nilai nya 195";  
588.          break;  
589.       case 196 :  
590.          echo "Nilai nya 196";  
591.          break;  
592.       case 197 :  
593.          echo "Nilai nya 197";  
594.          break;  
595.       case 198 :  
596.          echo "Nilai nya 198";  
597.          break;  
598.       case 199 :  
599.          echo "Nilai nya 199";  
600.          break;  
601.       case 200 :  
602.          echo "Nilai nya 200";  
603.          break;  
604.       case 201 :  
605.          echo "Nilai nya 201";  
606.          break;  
607.       case 202 :  
608.          echo "Nilai nya 202";  
609.          break;  
610.       case 203 :  
611.          echo "Nilai nya 203";  
612.          break;  
613.       case 204 :  
614.          echo "Nilai nya 204";  
615.          break;  
616.       case 205 :  
617.          echo "Nilai nya 205";  
618.          break;  
619.       case 206 :  
620.          echo "Nilai nya 206";  
621.          break;  
622.       case 207 :  
623.          echo "Nilai nya 207";  
624.          break;  
625.       case 208 :  
626.          echo "Nilai nya 208";  
627.          break;  
628.       case 209 :  
629.          echo "Nilai nya 209";  
630.          break;  
631.       case 210 :  
632.          echo "Nilai nya 210";  
633.          break;  
634.       case 211 :  
635.          echo "Nilai nya 211";  
636.          break;  
637.       case 212 :  
638.          echo "Nilai nya 212";  
639.          break;  
640.       case 213 :  
641.          echo "Nilai nya 213";  
642.          break;  
643.       case 214 :  
644.          echo "Nilai nya 214";  
645.          break;  
646.       case 215 :  
647.          echo "Nilai nya 215";  
648.          break;  
649.       case 216 :  
650.          echo "Nilai nya 216";  
651.          break;  
652.       case 217 :  
653.          echo "Nilai nya 217";  
654.          break;  
655.       case 218 :  
656.          echo "Nilai nya 218";  
657.          break;  
658.       case 219 :  
659.          echo "Nilai nya 219";  
660.          break;  
661.       case 220 :  
662.          echo "Nilai nya 220";  
663.          break;  
664.       case 221 :  
665.          echo "Nilai nya 221";  
666.          break;  
667.       case 222 :  
668.          echo "Nilai nya 222";  
669.          break;  
670.       case 223 :  
671.          echo "Nilai nya 223";  
672.          break;  
673.       case 224 :  
674.          echo "Nilai nya 224";  
675.          break;  
676.       case 225 :  
677.          echo "Nilai nya 225";  
678.          break;  
679.       case 226 :  
680.          echo "Nilai nya 226";  
681.          break;  
682.       case 227 :  
683.          echo "Nilai nya 227";  
684.          break;  
685.       case 228 :  
686.          echo "Nilai nya 228";  
687.          break;  
688.       case 229 :  
689.          echo "Nilai nya 229";  
690.          break;  
691.       case 230 :  
692.          echo "Nilai nya 230";  
693.          break;  
694.       case 231 :  
695.          echo "Nilai nya 231";  
696.          break;  
697.       case 232 :  
698.          echo "Nilai nya 232";  
699.          break;  
700.       case 233 :  
701.          echo "Nilai nya 233";  
702.          break;  
703.       case 234 :  
704.          echo "Nilai nya 234";  
705.          break;  
706.       case 235 :  
707.          echo "Nilai nya 235";  
708.          break;  
709.       case 236 :  
710.          echo "Nilai nya 236";  
711.          break;  
712.       case 237 :  
713.          echo "Nilai nya 237";  
714.          break;  
715.       case 238 :  
716.          echo "Nilai nya 238";  
717.          break;  
718.       case 239 :  
719.          echo "Nilai nya 239";  
720.          break;  
721.       case 240 :  
722.          echo "Nilai nya 240";  
723.          break;  
724.       case 241 :  
725.          echo "Nilai nya 241";  
726.          break;  
727.       case 242 :  
728.          echo "Nilai nya 242";  
729.          break;  
730.       case 243 :  
731.          echo "Nilai nya 243";  
732.          break;  
733.       case 244 :  
734.          echo "Nilai nya 244";  
735.          break;  
736.       case 245 :  
737.          echo "Nilai nya 245";  
738.          break;  
739.       case 246 :  
740.          echo "Nilai nya 246";  
741.          break;  
742.       case 247 :  
743.          echo "Nilai nya 247";  
744.          break;  
745.       case 248 :  
746.          echo "Nilai nya 248";  
747.          break;  
748.       case 249 :  
749.          echo "Nilai nya 249";  
750.          break;  
751.       case 250 :  
752.          echo "Nilai nya 250";  
753.          break;  
754.       case 251 :  
755.          echo "Nilai nya 251";  
756.          break
```

```

8.     echo "Nilai nya 2";
9.     break;
10.    case 3:
11.        echo "Nilai nya 3";
12.        break;
13.    default :
14.        echo "Selain dari nilai di atas";
15.    }
16. ?>

```

III.6. Break

Break mengakhiri eksekusi dari structure for,foreach,while, do-while atau switch. Break menerima sebuah nomor pilihan yang merupakan angka seberapa banyak structure yang ia keluar.

Contoh :

```

1.<?php
2. for ($i=1;$i<=10;$i++)
3.     { echo $i ."<br />";
4.         if ($i==7)
5.             { break; } // atau break 1
6.     }
7. echo "<br />";
8. for ($i=1;$i<=10;$i++)
9.     {for ($j=1;$j<=5;$j++)
10.         {
11.             echo $i ;
12.             echo "<br />";
13.             echo $j;
14.             if ($i==6)
15.                 { break 2; } // Break untuk for $i dan $j
16.         }
17.     }
18. ?>

```

III.7. Continue

Berfungsi untuk melewati/skip suatu looping

Contoh :

```

1.<?php
2. for ($i=1;$i<=5;$i++)
3.     {
4.         echo $i;
5.         echo "<br />";
6.         if ($i==3)
7.             { continue; }
8.     }
9. ?>

```

BAB IV

STRUKTUR CONTROL (2), FUNGSI DAN PROCEDURE

IV.1. If .. End if

Feature **If .. End If** berfungsi untuk menjalankan suatu syntax tertentu jika kondisi tertentu terpenuhi.

Contoh :

```
1. <?php  
2.     $nilai1=10;  
3.     if ($nilai1==10)  
4.         { echo "Nilai sesuai "; }  
5. ?>
```

IV.2. If ... Else ... End If

Feature **If .. Else .. End If** berfungsi untuk menjalankan suatu syntax tertentu jika kondisi tertentu terpenuhi dan ada kondisi lain.

Contoh :

```
1.<?php  
2. $nilai1=10;  
3. if ($nilai1==10)  
4.     { echo "Nilai sesuai"; }  
5. else  
6.     { echo "Nilai tidak sesuai"; }  
7.?>
```

IV.3. If .. Elseif .. Endif

Contoh :

```
1.<?php  
2. $nilai1=10;  
3. $nilai2=15;  
4. if ($nilai1 < $nilai2)  
5.     { echo "Pernyataan benar"; }  
6. elseif ($nilai1>$nilai2)  
7.     { echo "Pernyataan salah"; }  
8. else  
9.     { echo "No Comment"; }  
10.?>
```

IV.4. Function

Jika ada sebuah baris coding yang akan di ulang pada beberapa bagian program atau kita hendak memisahkan suatu fungsi tertentu dari program utama, kita dapat menggunakan function dan procedure.

Syntax :

```
function nama_function(exp1,exp2,...)  
{  
    statement  
}  
exp1: variabel (optional)
```

Contoh :

```
1.<?php  
2. function bilangan($nilai1,$nilai2)  
3. {  
4.     $hasil=$nilai1+$nilai2;  
5.     echo "Hasil penjumlahan : $hasil";  
6. }  
7.
```

```
8. echo "Hasil Function nya nich <br />";
9. $bil1=4;
10. $bil2=17;
11. bilangan($bil1,$bil2); // Output adalah 21
12. ?>
```

1. Menggunakan return

Return dapat di gunakan di dalam sebuah function untuk mengembalikan sebuah nilai yang dapat bertipe apa saja.

Contoh :

```
1.<?php
2. function bilangan($nilai1,$nilai2)
3. {
4.     $hasil=$nilai1+$nilai2;
5.     return $hasil;
6. }
7.
8. echo "Hasil Function nya nich <br />";
9. $bil1=4;
10. $bil2=17;
11. $hasilnya=bilangan($bil1,$bil2);
12. echo $hasilnya; // Output : 21
13. ?>
```

IV.5 Procedure

Procedure adalah fungsi yang tidak mengembalikan nilai.

Contoh :

```
1.<?php
2.function jumlah()
3. {
4.     $hasil=4+6;
5.     echo $hasil;
6. }
7.?>
```

Contoh :

```
1.<?php
2.jumlah();
3.function jumlah()
4. { $hasil=4+6;
5.     echo $hasil;
6. }
7.?>
```

BAB V

ARRAY DAN OBJECT ORIENTED (1)

Array adalah sebuah variabel yang menyimpan banyak nilai di dalamnya secara teratur, yang setiap nilai memiliki alamat masing-masing secara unique.

Syntax : array().

V.1 Array 1 dimensi

Adalah array biasa yang terdiri dari 1 dimensi atau daftar nilai. Untuk mencetak sebuah nilai pada array di gunakan :

```
$nama_array[alamat]
```

Contoh :

```
1.<?php  
2. $hari = array ("Senin","Selasa","Rabu","Kamis","Jum'at","Sabtu","Minggu");  
3. echo "Hari ini adalah hari : $hari [0]";  
4. ?>
```

Awal alamat array dapat di ubah, seperti :

```
1. <?php  
2. $hari = array (1=>"Senin","Selasa","Rabu","Kamis","Jum'at",  
3. "Sabtu","Minggu");  
4. print_r($hari); // Mencetak isi array $hari  
5. echo "Hari ini adalah hari : $hari[1]"; // Hari senin  
6. ?>
```

V.2 Array 2 dimensi

Adalah sebuah nama array yang dapat menyimpan array lain di bawah nya.

Contoh :

```
1. <?php  
2. $arr2=array("kelas_5"=>array(1=>"Andi","Umar","Ihsan"),  
3. "kelas_6"=>array (1=>"Nia","Siti","Amir"));  
4. $data1=$arr2["kelas_5"][2]; // Umar  
5. $data2=$arr2["kelas_6"][3]; // Amir  
6. echo "Data 1 : $data1 <br/>";  
7.echo "Data 2 : $data2 <br/>";  
8. ?>
```

V.3 Object Oriented Programming (OOP) (1)

Selain menerapkan procedural programming, PHP sejak versi 5.0 juga sudah mensupport OOP Programming secara penuh. PBO sendiri dapat di artikan sebagai pengelompokan modul-modul pemrograman di dalam class-class. Manfaat dari OOP yang terpenting adalah, agar seorang programmer terhindar dari mengulangi kode yang sama untuk tugas yang sama. Programmer cukup menggunakan kode yang sama untuk tugas yang sama dengan hanya memanggil method yang terdapat di dalam class. Di dalam procedural programming, method, dapat di samakan dengan function/procedure.Object Object adalah sebuah paket code yang berhubungan dengan state dan behaviour. Object sering di hubungkan dengan object di dalam kehidupan sehari-hari. Di dalam dunia nyata atau dalam sebuah program, Object memiliki 2 karakteristik, yaitu state dan behaviour. State adalah keadaan dari sebuah Object, sedangkan behaviour adalah kelakuan dari object tersebut.

Contoh :

Object : Mobil
State : warna,merk, tahun pembuatan, model
Behaviour : melaju, berbelok, membunyikan klakson

Sebuah Object yang di bentuk dari class biasa di sebut instance, arti nya Object tersebut adalah wujud nyata dari sebuah class. Variabel dan method dari instannce ini di sebut variabel instance dan methode instance. Setiap instance menyimpan variabel nya sendiri-sendiri, jadi nilai variabel untuk tiap instance bisa berbeda.

State di dalam OOP di sebut dengan properties dan Behaviour di sebut dengan method.

- Membuat class

```
1.<?php
2. class MyClass
3. {
4.     // Di isi dengan property class dan method
5. }
6. ?>
```

Class yang baru di buat, dapat di instantiate (Memberi contoh) dan di simpan ke dalam sebuah variabel dengan menggunakan keyword new.

```
$obj = new MyClass;
```

- Mendefinisikan property class

Property di dalam sebuah class , dapat juga di sebut dengan variabel di luar OOP. Fungsi nya sama dengan variabel, yakni untuk menyimpan value, namun property terikat dengan dengan sebuah object, sehingga hanya dapat di akses lewat sebuah object.

```
1.<?php
2. class MyClass
3. {
4.     public $nama="Akhmad Sofwan";
5. }
6. $obj = new MyClass;
7. ?>
```

Untuk dapat mengakses property \$nama, contoh nya, ingin mencetak di layar, nilai dari property \$nama, maka di gunakan kode : echo \$obj →nama;

- Mendefinisikan method class

Method di dalam OOP, sama dengan function / procedure di dalam procedural programming Hanyasaja, sama seperti dengan property, method di akses lewat object dengan menggunakan tanda panah (→).

Contoh:

```
<?php
class MyClass
{
    public $nama="Akhmad Sofwan";
    public function hello()
    {
        return $this→nama;
    }
}
$obj = new MyClass;
echo $obj → hello();
?>
```

Output :
Akhmad Sofwan

- Constructor dan Destructor

Ketika sebuah object di instantiasi, seringkali di perlukan untuk menset sejumlah hal secara otomatis pada awal proses. Dalam hal ini, PHP menyediakan magic method `__construct()`, yang secara otomatis di panggil kapanpun ketika sebuah object baru di buat.

Contoh:

```
1.<?php
2.class MyClass
3.{
4.    public $nama="Akhmad Sofwan";
5.    public function __construct()
6.    {
7.        echo 'The class "','"__CLASS__," was initiated !<br/>';
8.    }
9.
10.   public function hello()
11.   {
12.       return $this->nama;
13.   }
14.   $obj = new MyClass;
15.   echo $obj -> hello();
16. ?>
```

Output :
The class MyClass was initiated
Akhmad Sofwan

Pada source code di atas, method `__construct` di jalankan secara otomatis tatkala class `MyClass` di initiasi, baru kemudian di cetak method `hello`, yang berupa pencetakan property `$nama` dengan value "Akhmad Sofwan".

Untuk memanggil sebuah function ketika object di destroy, PHP menyediakan magic method `__destruct()`. Fungsi nya untuk class cleanup (Misal : Menutup koneksi database)

Contoh:

```
1.<?php
2.class MyClass
3.{
4.    public $nama="Akhmad Sofwan";
5.    public function __construct()
6.    {
7.        echo 'The class "','"__CLASS__," was initiated !<br/>';
8.    }
9.
10.   public function __destruct()
11.   {
12.       echo 'The class "','"__CLASS__," was destroyed !<br/>';
13.   }
14.
15.   public function hello()
16.   {
17.       return $this->nama;
18.   }
19.   $obj = new MyClass;
20.   echo $obj -> hello();
21. ?>
```

Output :

The class MyClass was initiated
Akhmad Sofwan
The class MyClass was destroyed

Magic method `__destruct` secara otomatis di jalankan, saat aplikasi berakhir. Untuk menjalankan Magic method `__destruct` secara eksplisit, dapat di nyatakan dengan `unset(nama_object)` [3].