

# MODUL PROGRAMMING WEB 2



OLEH :

ENDAR NIRMALA, S.Kom, MT

JURUSAN TEKNIK INFORMATIKA  
UNIVERSITAS PAMULANG

---

## KATA PENGANTAR

Web merupakan salah satu sumber informasi yang banyak digunakan. Berbagai aplikasi web dibuat dengan tujuan agar pemakai dapat berinteraksi dengan penyedia informasi dengan mudah dan cepat melalui dunia Internet. Aplikasi web tidak lagi terbatas sebagai pemberi informasi yang bersifat statis, melainkan juga mampu memberikan informasi yang berubah secara dinamis, dengan melakukan koneksi terhadap database.

Sehubungan dengan kebutuhan untuk membangun aplikasi web yang bersifat dinamis modul ini dibuat. Harapan yang diinginkan agar pembaca khususnya mahasiswa dapat belajar sendiri untuk mengimplementasikannya. Dengan pendekatan "belajar dengan mencoba dan berlatih", modul ini dapat membantu mahasiswa memiliki kemampuan dasar dalam membuat aplikasi web yang dinamis.

Modul ini mencakup hal-hal dasar tentang PHP hingga cara mengakses database dengan MySQL. Semua contoh latihan sudah diuji di lingkungan Windows dengan menggunakan Apache Server. Untuk lebih melengkapi modul ini anda dapat mendownload sendiri situs-situs tutorial PHP yang berada di Internet.

Akhir kata semoga modul ini dapat memberikan manfaat.

Jakarta, Mei 2011

Penulis

---

## DAFTAR ISI

<b>KATA PENGANTAR</b>	<b>1</b>
DAFTAR ISI.....	3
BAB I PENGENALAN PHP .....	4
1.1.    Pendahuluan .....	4
1.2.    Struktur Penulisan PHP .....	5
BAB II ELEMEN-ELEMEN DASAR PHP.....	10
2.1 Operator.....	10
2.2 Tipe Data .....	11
BAB III STRUKTUR KONTROL .....	16
3.1 Struktur Seleksi .....	16
3.2 Struktur Perulangan .....	19
BAB IV MANIPULASI BERKAS.....	23
4.1. Prinsip Pemrosesan Berkas .....	23
4.2. Mode Dalam Berkas .....	23
BAB V ARRAY .....	26
5.1 Insialisasi Array .....	26
5.2 Fungsi dalam Array.....	28
BAB VI FUNCTION.....	34
6.1 Penulisan Function.....	34
6.2 Jenis-jenis fungsi Buit-In.....	37
BAB VII DATABASE.....	43
7.1 Operasi Database .....	43
7.2 Fungsi PHP – MySql.....	44
BAB VIII MANIPULASI DATABASE .....	50
BAB IX SESSION DAN COOKIES .....	57
9.1 Session.....	57
9.2 Cookies .....	59
BAB X HANDLING ERRORS.....	62
DAFTAR PUSTAKA .....	74

---





## BAB I PENGENALAN PHP

### 1.1. Pendahuluan

PHP singkatan dari *Hypertext Preprocessor*, merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server, hasilnya dikirim ke klien tempat pemakai menggunakan browser. Ketika seseorang mengunjungi web berbasis PHP, web server akan memproses code-code PHP. Beberapa perintah atau code dari PHP tersebut selanjutnya ada yang diterjemahkan ke dalam HTML dan beberapa ada yang disembunyikan (misalnya proses kalkulasi dan operasi). Setelah diterjemahkan ke dalam HTML, web server akan mengirim kembali ke web browser pengunjung tersebut. Secara khusus, PHP dirancang untuk membentuk web dinamis, artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini.

#### Aplikasi Yang Diperlukan

Untuk dapat bekerja dengan PHP, berikut ini adalah beberapa aplikasi yang diperlukan:

-  Web server (Apache, IIS, Personal Web Server/PWS)
-  PHP Parser (dapat didownload di PHP.net)
-  Database server (MySQL, Interbase, MS SQL, dll)
-  Web Editor (Dreamweaver, Frontpage, dll)

Saat ini telah ada program terbundle seperti PHPTriad ataupun Apache2Triad dan XAMPP yang di dalamnya sudah terdapat PHP parser, Web server Apache dan database MySQL dalam satu paket. PHP parser adalah program yang digunakan untuk menterjemahkan (intepreter) code script dan kemudian mengeksekusinya. PHP juga termasuk dalam HTML-embedded, artinya code php dapat kita sisipkan pada sebuah halaman HTML. Untuk mengetahui bagaimana basis - baris HTML merupakan suatu script php maka digunakan pasangan tag.

---

## 1.2.Struktur Penulisan PHP

Tag yang dapat digunakan untuk menyatakan script php adalah:

<code>&lt;?php</code>	<code>&lt;?</code>	<code>&lt;%</code>	<code>&lt;script language="PHP"&gt;</code>
...	...	...	...
...	...	...	...
<code>?&gt;</code>	<code>?&gt;</code>	<code>%&gt;</code>	<code>...&lt;/script&gt;</code>

Diantara keempat ini tag pertama merupakan tag yang paling amaran karena memberikan tanda yang lebih jelas dan pada umumnya di mengerti oleh web server.

Contoh HTML embeded PHP :

```
<HTML>
<HEAD>
<TITLE>PHP Embeded </TITLE>
</HEAD>
<BODY>

<?php
echo "<B>File PHP Pertama saya <BR> Ini PHP didalam
HTML </B>";
?>
</BODY>
</HTML>
```

### ❖ Penulisan Komentar

Komentar merupakan bagian program yang tidak akan dieksekusi. Fungsi dari komentar ini adalah sebagai dokumentasi program atau berupa penjelasan dari program. Penulisan komentar dalam PHP ada dua macam :

1. Komentar satu baris diawali dengan tanda //
2. Komentar lebih dari satu baris diawali dengan /\* dan diakhiri dengan \*/

Contoh :

```
<?php
/*
echo "Kalimat ini tidak akan dicetak";
*/
?>

<?php
echo "Kalimat ini akan dicetak"; // Bagian ini
hanya merupakan komentar
?>
```

#### ❖ Variabel

Variabel dalam PHP didefinisikan dengan diawali tanda \$. Atribut penamaan variabel dalam PHP pada umumnya serupa dengan variabel pada umumnya, yaitu dengan aturan sbb :

- Nama variabel bisa terdiri dari angka, huruf, dan underscore. Selain itu tidak boleh digunakan.
- Huruf pertama nama variabel tidak boleh berupa angka.

Pada PHP nama variabel bersifat case-sensitive yaitu memperhatikan huruf besar dan kecil.

Contoh :

```
//penamaan berikut benar
$nama = "wawan";
$test123 = "Test";
$_TEST = "Test";
$Test123 = "Test";

//penamaan yang salah
$1Huruf = "a saja"; //diawali dengan angka
$sw3ar-it = "Again"; //menggunakan karakter "-"
$Tom&Jerry = "Kucing"; // menggunakan karakter "&"
```

PHP memiliki 3 jenis scope variabel yaitu :

1. Lokal Variable yaitu variabel yang hanya dapat digunakan di dalam fungsi
2. Global variabel yaitu variabel yang dapat digunakan di luar dan di dalam fungsi
3. Static Variabel yaitu variabel yang hanya ada dalam lingkup lokal suatu fungsi

#### Latihan 1 :

```
<?php
$A = 123 ; // variable global
function Test() {
$A = "Test"; // variable local
echo "Nilai A dalam fungsi = $A \n";
}
Test();
echo "Nilai A luar fungsi = $A \n";
?>
```

Jika dijalankan maka akan tercetak :

Nilai A dalam fungsi = "Test"

Nilai A luar fungsi = 123

#### Latihan 2 :

```
<?php
$A = 123 ; // variable global
function Test() {
global $A ; // variable local
echo "Nilai A dalam fungsi = $A \n";
}
Test();
echo "Nilai A luar fungsi = $A \n";
?>
```

Jika dijalankan maka akan tercetak :

Nilai A dalam fungsi = 123

Nilai A luar fungsi = 123

### Latihan 3 :

```
<html>
<head><title>Variabel Static</title></head>
<body><h1>Variabel Static</h1>
<?php
    Function test()
    {
        Static $a=0; // dengan static
        Echo "Nilai a : ";
        Echo $a;
        Echo "<br>";
        $a++;
    }
    Test();
    Test();
    Test();
    Test();
    Test();
?>
</body>
</html>
```

#### ❖ Konstanta

Konstanta mirip dengan variabel. Bedanya nilai konstanta tidak pernah berubah sejak di deklarasikan. Untuk mendeklarasikan konstanta digunakan fungsi `define()`.

**Sintak : `define("nama_konstanta",nilai);`**

### Latihan 4 :

```
<?php
Define("judul", "Menghitung luas lingkaran");
define("phi",3.14) ;
$r = 5;
$luas = phi * $r * $r ;
Echo Judul;
echo "Luas = $luas" ;
?>
```



**Latihan 5 :**

```
<?php
// Ini adalah komentar dalam satu baris

/* Kalau yang ini, komentar
dalam banyak baris, yang baru
akan selesai setelah diakhiri
dengan */
?>

<html>
<head>
<title>Test Penyisipan PHP Pada HTML</title>
</head>
<body>
Kapal Asing, Silakan identifikasikan diri Anda! <br>
<?php
// Berikut ini adalah inisiasi beberapa variabel
$namad = "Jean";
$namat = "Luc";
$namab = "Piccard";
$nilai1=25;
$nilai2=50;
$hasil=$nilai1*$nilai2;
$a=2;
$b=3;
$hsl=pow($a,$b);
?>
<b>Ini adalah kapal Federasi Planet USS Enterprise.<br>
<?php
echo "Saya $namab, $namad $namat $namab, kapten
kapal.</b><br>";
echo "$nilai1 x $nilai2 = $hasil<br>";
echo "$a ^ $b = $hsl";
?>
</body>
</html>
```

## BAB II ELEMEN-ELEMEN DASAR PHP

### 2.1 Operator

Operators digunakan untuk melakukan oprasi pada suatu nilai.

#### ❖ Operator Arithmetik

Operator	Description	Contoh	Hasil
+	Penjumlahan	$x=2$ $x+2$	4
-	Pengurangan	$x=2$ $5-x$	3
*	Perkalian	$x=4$ $x*5$	20
/	Pembagian	$15/5$ $5/2$	3 2.5
%	Modulus (sisa pembagian)	$5\%2$ $10\%8$ $10\%2$	1 2 0
++	penambahan dengan 1	$x=5$ $x++$	$x=6$
--	Pengurangan dengan 1	$x=5$ $x--$	$x=4$

#### ❖ Operator Assignment

Operator	Contoh	Sama dengan
=	$x=y$	$x=y$
+=	$x+=y$	$x=x+y$
-=	$x-=y$	$x=x-y$
*=	$x*=y$	$x=x*y$
/=	$x/=y$	$x=x/y$
.=	$x.=y$	$x=x.y$
%=	$x\%=y$	$x=x\%y$

#### ❖ Operator Pembandingan

Operator	Description	Contoh
==	sama dengan	$5==8$ menghasilkan false(salah)
!=	tidak sama dengan	$5!=8$ menghasilkan true(benar)
>	lebih besar dari	$5>8$ menghasilkan false(salah)
<	lebih kecil dari	$5<8$ menghasilkan true(benar)
>=	besar atau sama dengan	$5>=8$ menghasilkan false(salah)
<=	lebih kecil atau sama dengan	$5<=8$ menghasilkan true(benar)

## ❖ Operator Logika

Operator	Description	Contoh
&&	And	x=6 y=3(x < 10 && y > 1) returns true
	Or	x=6 y=3(x==5    y==5) returns false
!	Not	x=6 y=3!(x==y) returns true

## ❖ Operator String

Operasi dalam string meliputi penggabungan dua buah string dan fungsi-fungsi untuk memanipulasi string. **Operator . dan .=**

## 2.2 Tipe Data

Tipe data dalam PHP digunakan untuk menentukan jenis data yang akan disimpan dalam suatu variabel. PHP mempunyai empat tipe data dasar yaitu: integer, floating point, character dan boolean.

**Integer** merupakan tipe data yang mencakup semua bilangan bulat. Range bilangan integer adalah antara -2.147.483.647 sampai dengan 2.147.483.647.

Contoh integer:

```
$biaya = 1000;
$a = 1234;
$a = -123;
```

**Floating point** merupakan tipe data yang mencakup semua bilangan desimal (bilangan yang memiliki angka di belakang koma). Range bilangan floating point antara 1e-308 sampai dengan 1e308 (10<sup>-308</sup> s.d. 10<sup>308</sup>).

Contoh floating point:

```
$a = 2.345;
$a = 3.2e4;
```

**Character** merupakan tipe yang digunakan untuk menyimpan data-data yang berupa karakter (satu huruf). Penulisan nya biasanya diapit dengan tanda kutip satu ('...').

---

Contoh character:

```
$abjad = 'e';
```

```
$abjad = 'n';
```

Jika karakter-karakter berkumpul membentuk suatu kata, kalimat atau paragraf maka tipe tadi tidak bisa lagi disebut sebagai character tapi bertipe string.

**String** merupakan tipe data tersendiri dan tidak dapat dikelompokkan menjadi tipe data dasar. Penulisannya biasa diapit dengan tanda kutip dua (“...”).

Contoh string:

```
$nama = "Karina";
```

```
$salamat = "Rasamala Raya";
```

Tipe yang terakhir adalah **boolean**. Boolean menyimpan data –data yang bernilai logika, benar atau salah, 0 atau 1, ya atau tidak .

### Latihan 1.

```
<?php
$a = 1;
$b = 2;
$c = $a.$b;
//nilai $c sekarang 12, menggabungkan angka 1 dan 2
$d= $c + 1 ;
echo $d;
// nilai $d 13,
$e = "Number";
$f = $e.$d;
echo $f;
//nilai $f menjadi Number13, a kombinasi $e dan $f
?>
```

### Latihan 2 :

```
<?php
// inisiasi variable yang digunakan
// nama peralatan
$brg1 = "Buku";
$brg2 = "Mouse";
$brg3 = "FlashDisk";
$brg4 = "Pulpen";
// harga per unit peralatan
$harga1 = 17500;
```

```
$harga2 = 30000;
$harga3 = 70000;
$harga4 = 22300;

// jumlah peralatan yang ada
$jmlbrg1 = 2;
$jmlbrg2 = 5;
$jmlbrg3 = 1;
$jmlbrg4 = 3;
// total harga per jenis peralatan
$th1 = $jmlbrg1 * $harga1;
$th2 = $jmlbrg2 * $harga2;
$th3 = $jmlbrg3 * $harga3;
$th4 = $jmlbrg4 * $harga4;
// hitung grand total nilai peralatan
$tharga = $th1 + $th2+ $th3 + $th4;
// besar diskon
$diskon = 5;
// jumlah total diskon yang diberikan
$tdiskon = ($diskon * $tharga)/100;
// jumlah yang harus dibayar
$tdibayar = $tharga - $tdiskon;
?>

<html>
<head>
<title>Daftar Peralatan Yang Dibeli</title>
</head>
<style TYPE="text/css">
    body {
        font-size: 14pt;
    }
    table {
        font-size: 25pt;
    }
</style>
<body>
<center>
<font face="com ic sans serif" size=5 color="blue">Contoh Perhitungan dengan PHP</font>
<table border="1" cellspacing="0" cellpadding="3">

<tr>
<td colspan="4" align="center" valign="middle">
<b>Daftar Pemesanan Peralatan Kantor</b>
</td>
```

---

```

</tr>
<tr>
<td><b>Nama Peralatan</b></td>
<td><b>Jumlah</b></td>
<td><b>Harga Satuan</b></td>
<td><b>Jumlah Harga</b></td>
</tr>
<?php
// Mulai untuk mengisi tabel daftar dengan data yang
ada
?>
<tr>
<td align="left"><?php echo $brg1; ?></td>
<td align="right"><?php echo $jmlbrg1; ?></td>
<td align="right"><?php echo $harga1; ?></td>
<td align="right"><?php echo $th1; ?></td>
</tr>
<tr>
<td align="left"><?php echo $brg2; ?></td>
<td align="right"><?php echo $jmlbrg2; ?></td>
<td align="right"><?php echo $harga2; ?></td>
<td align="right"><?php echo $th2; ?></td>
</tr>
<tr>
<td align="left"><?php echo $brg3; ?></td>
<td align="right"><?php echo $jmlbrg3; ?></td>
<td align="right"><?php echo $harga3; ?></td>
<td align="right"><?php echo $th3; ?></td>
</tr>
<tr>
<td align="left"><?php echo $brg4; ?></td>
<td align="right"><?php echo $jmlbrg4; ?></td>
<td align="right"><?php echo $harga4; ?></td>
<td align="right"><?php echo $th4; ?></td>
</tr>
<tr>
<td colspan="3" align="right">Total Harga</td>
<td align="right"><?php echo $tharga; ?></td>
</tr>
<tr>
<td colspan="3" align="right">
Diskon <?php echo "( $diskon % )"; ?></td>
<td align="right"><?php echo $tdiskon; ?></td>
</tr>
<tr>
<td colspan="3" align="right">Jumlah harus
dibayar</td>

```

---

---

```
<td align="right"><?php echo $tdibayar; ?></td>
</tr>
</table>
</center>
</body>
</html>
```

### Latihan 3.

Buatlah tampilan dibawah ini

Nilai I		Nilai II	
<input type="text"/>	<div><div>+</div><div>▼</div></div>	<input type="text"/>	<input type="button" value="submit"/>
	<div><div>+</div><div>-</div><div>*</div><div>/</div></div>		

Lakukanlah perhitungan dan tampilkan hasilnya dibawahnya masih dihalaman yang sama, boleh juga dikembangkan tampilkan di halaman baru.

---

## BAB III STRUKTUR KONTROL

### 3.1 Struktur Seleksi

#### ❖ IF

Bentuk umum :

```
if(condition)
{
    statement
}
```

```
if (condition)
{
    statement 1
}else{
    statement 2
}
```

```
if (condition)
{
    statement 1
}elseif
(condition){
    statement 2
}
```

#### Latihan 1 :

```
<?
$ball = 'purple';
    //Coba ubah ke "yellow", "blue", "green", "purple",
atau warna lain
if ($ball == 'red') {
    $redbox = $ball;
} elseif ($ball == 'yellow') {
    $yellowbox = $ball;
} elseif ($ball == 'blue') {
    $bluebox = $ball;
} elseif ($ball == 'green') {
    $greenbox = $ball;
} elseif ($ball == 'purple') {
    $purplebox = $ball;
} else {
    $colorlessbox = $ball;
}
echo "red box : $redbox <br>\n";
echo "yellow box : $yellowbox <br>\n";
echo "blue box : $bluebox <br>\n";
echo "green box : $greenbox <br>\n";
echo "purple box : $purplebox <br>\n";
echo "colorless box : $colorlessbox <br>\n";
?>
```



---

## Latihan 2.

```
<html>
<head>
<title>contoh Penggunaan IF</title>
</head>
<body>
<form>
    Besar Pembelian :
    <input type=text name=total_beli><br><br>
    <input type=submit value="Tentukan Diskon">
</form>
<?php
    if (isset($total_beli))
    {
        $total_beli=intval($total_beli);
        $diskon=0;
        if($total_beli>=200000)
            $diskon=0.1;
        else if ($total_beli>=100000)
            $diskon= 0.05;
        else
            $diskon=0.01;

        $diskon=$diskon * intval($total_beli);
        printf("Diskon = %s <br>\n", $diskon);
        printf("Pembayaran = %s <br>\n", $total_beli-
        $diskon);
    }
?>
</body>
</html>
```

## ❖ SWITCH CASE



```
switch(variable) {
case nilai 1 :
    perintah yang dijalankan ;
    break;
case nilai 2 :
    perintah yang dijalankan ;
    break;
...
default :
    perintah yang dijalankan ;
    break;
}
```

---

### Latihan 3.

```
<?php
    $destination="Tokyo";
    Switch ($destination){
        case "Las Vegas":
            echo "Bring an extra $500";
            break;
        case "Amsterdam":
            echo "Bring an open mind";
            break;
        case "Egypt":
            echo "Bring 15 bottles of SPF 50 Sunscreen";
            break;
        case "Tokyo":
            echo "Bring lots of money";
            break;
        case "Caribbean Islands":
            echo "Bring a swimsuit";
            break;
    }
?>
```

### Latihan 4.

```
<html>
<head>
<title>PenggunaanSwitch - Case</title>
</head>
<body>
Hari ini :
<?Php
    $nama_hari = date("l");
    Switch ($nama_hari)
    {
        Case "Sunday" ;
        Print("Minggu");
        print "Waktu untuk istirahat";
        Break;
        Case "Monday" ;
        Print("Senin <br>");
        print "Meeting awal minggu jam 08.00";
        Break;
        Case "Tuesday" ;
        Print("Selasa <br>");
        print "Pembukaan Workshop Diklat";
        Break;
```

```
        Case "Wednesday" ;
            Print("Rabu <br>");
            print ("Seminar Launchig Window Vista di JHCC");
            Break;
        Case "Thrusday" ;
            Print("Kamis <br>");
            print "Pertemuan dengan Mahasiswa";
            Break;
        Case "Friday" ;
            Print("Jum'at <br>");
            print "Jogging bersama";
            Break;
        Default :
            Print("Sabtu <br>");
            print "Survey harga ke Dusit, Mangga Dua";
    }
?>
</body>
</html>
```

## 3.2 Struktur Perulangan

### ❖ WHILE Loop

```
while(kondisi) {
    operasi program ;
}
```

### ❖ Do - While loop

```
do {
    proses program ;
}while($syarat);
```

### ❖ For loop

```
for( nilai awal loop, kondisi , penambahan ) {
    operasi program ;
}
```

### ❖ For Each

```
foreach ($array as $value)
{
    operasi program;
}
```

#### Latihan 5

```
<?php
$brush_price = 5;
$counter = 10;

echo "<table border=\"1\" align=\"center\">";
echo "<tr><th>Quantity</th>";
echo "<th>Price</th></tr>";
while ( $counter <= 100 ) {
    echo "<tr><td>";
    echo $counter;
    echo "</td><td>";
    echo $brush_price * $counter;
    echo "</td></tr>";
    $counter = $counter + 10;
}
echo "</table>";
?>
```

#### Latihan 6

```
<?php
$i = 0;
echo 'This code will run at least once because i
default value is 0.<br>';
do {
echo 'i value is ' . $i . ', so code block will run.
<br>';
++$i;
} while ($i < 10);
?>
```

#### Latihan 7.

```
<html>
<head>
<title>Penggunaan For</title>
</head>
<body>
<?Php
    For ($bil = 1; $bil <25; $bil++)
        Print("$bil <br>\n");
?>
</body>
</html>
```

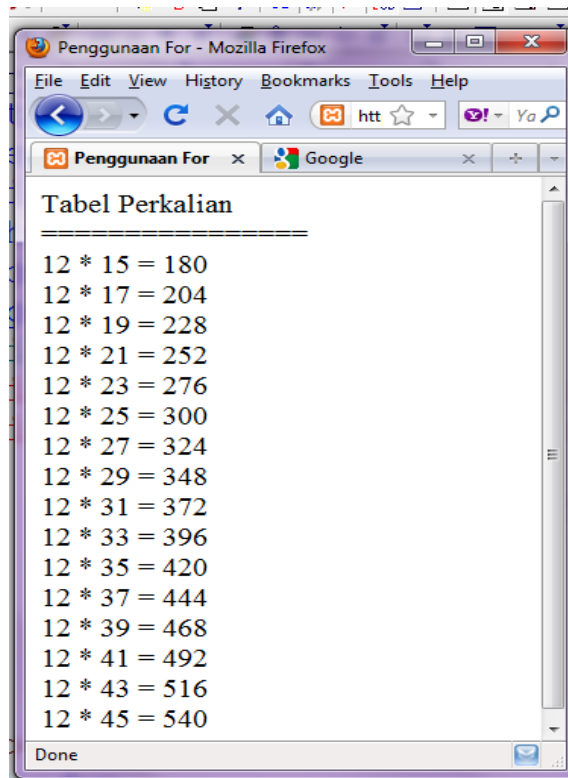
#### Latihan 8.

```
<?php
for($i = 1 ;$i < 11 ;$i++) {
if ($i % 2 == 0 ) continue ;
else echo $i ;
}
?>
```

#### Latihan 9.

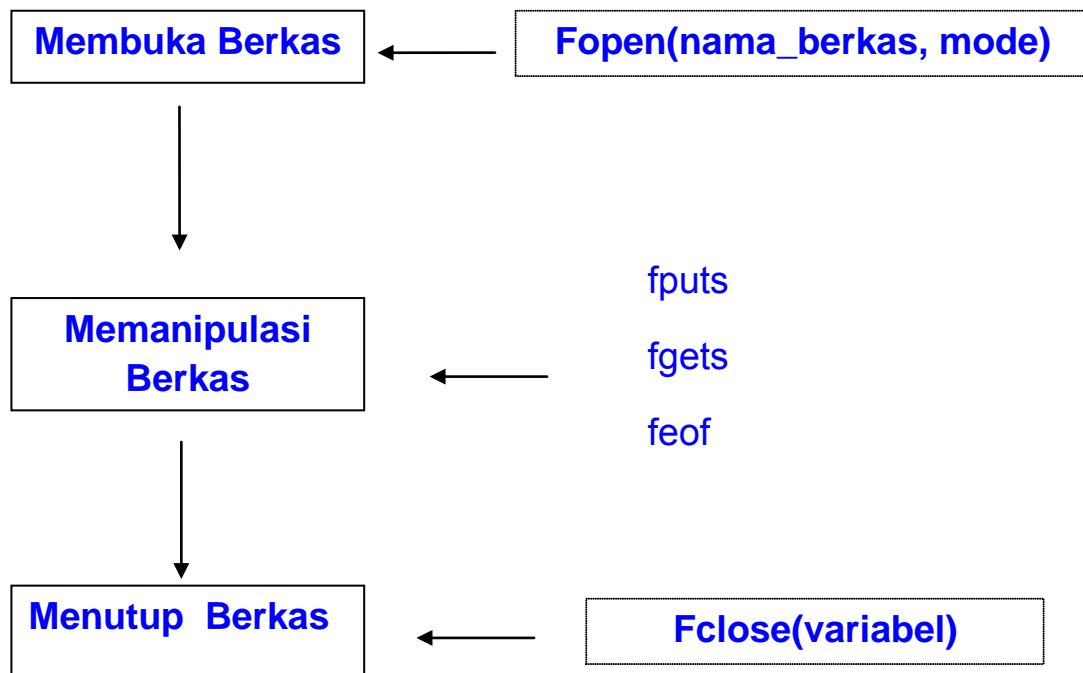
```
<?php
$arr=array("senin", "selasa", "rabu");
foreach ($arr as $hari){
echo "Hari: " . $hari . "<br>";
}
?>
```

- ❖ Dari latihan diatas apa perbedaan perintah break dan continue
- ❖ Dengan menggunakan perintah looping buatlah tabel perkalian, misalnya dengan tampilan sebagai berikut :



## BAB IV MANIPULASI BERKAS

### 4.1. Prinsip Pemrosesan Berkas



### 4.2. Mode Dalam Berkas

Mode	Keterangan
r	Berkas hanya bisa dibaca. Penunjuk berkas akan diletakkan di awal berkas
r+	Berkas dibuka dengan mode baca dan tulis. Penunjuk berkas diletakkan di awal berkas
w	Mode perekaman data. Berkas akan diciptakan kalau belum ada, kalau sudah ada isinya akan dihapus
w+	Berkas dibuka dengan mode baca dan tulis. Berkas akan diciptakan kalau belum ada, kalau sudah ada isinya akan

---

	dihapus
A	Mode ini untuk penambahan data. Pada saat berkas dibuka, penunjuk berkas diletakkan pada akhir berkas. Apabila berkas belum ada akan diciptakan
a+	Berkas dibuka dengan mode baca dan tulis, penunjuk berkas diletakkan pada akhir berkas. Apabila berkas belum ada akan diciptakan

```
fputs(file,string,length)
fgets(file,length)
```

#### Latihan 1.

Ketikkan pada notepad kalimat dibawah ini.

```
Hello, this is a test file.
There are three lines here.
This is the last line.
```

Simpan dengan nama : test1.txt

#### Latihan 2.

```
<?php
$file = fopen("test1.txt","r");
echo fgets($file);
fclose($file);
?>
```

Simpan dengan nama fgets1.php

#### Latihan 3.

```
<?php
$file = fopen("test1.txt","r");
while(! feof($file))
{
    echo fgets($file). "<br />";
}
fclose($file);
?>
```



Simpan dengan nama fgets2.php

### Latihan 3.

```
<html>
<head>
<title>Contoh Counter</title>
</head>
<body>
  <?php
    $nama_file="counter.dat";
    If (file_exists($nama_file))
    {
      $berkas = fopen($nama_file,"r");
      $pencacah = (integer)trim(fgets($berkas, 255));
      $pencacah++;
      Fclose($berkas);
    }
    Else
      $pencacah = 1;
      // simpan pencacah
      $berkas = fopen($nama_file,"w");
      Fputs($berkas, $pencacah);
      Fclose($berkas);

      // tulis ke halaman web
      Print("Anda pengunjung ke-$pencacah <br>\n");    ?>
  </body>
</html>
```

Simpan dengan nama counter.php

---

## BAB V ARRAY

### 5.1 Insialisasi Array

Array merupakan kumpulan beberapa data yang disimpan dalam satu variabel. Jadi, berbeda dengan variabel sebelumnya yang hanya menyimpan satu nilai saja, array dapat menampung lebih dari satu nilai.

<b>Charlie</b>	<b>Ariel</b>	<b>Budiman</b>
----------------	--------------	----------------

\$nama

Tiap nilai dalam array (disebut elemen) di akses dengan menggunakan index. PHP mengenal dua macam index, yaitu index numerik dan index asosiatif. Index numerik menggunakan angka untuk menandai tiap elemen, sedangkan index asosiatif memberi nama untuk masing-masing elemen.

#### Inisialisasi Array

**\$kota = array {"Yogya","Solo","Bandung","Bogor"}**

```
$kota[]="Yogya";
$kota[]="Solo";
$kota[]="Bandung";
$kota[]="Bogor";
```

Yogya	Solo	Bandung	Bogor
0	1	2	3

#### Contoh penulisan array

##### Cara 1 :

```
<?Php
    $personal = array {
        "nama" => "Salman",
        "kendaraan" => "Minibus",
        "rumah" => "Perumnas"
    };
    Print $personal["nama"] . "\n";
```

```
Print $personal["kendaraan"] . "\n";
Print $personal["rumah"]."\\n";
?>
```

### Cara 2 :

```
<?Php
$personal["nama"] ="Salman";
$personal["kendaraan"] ="Minibus";
$personal["rumah"] ="Perumnas";

Print $personal["nama"] . "\n";
Print $personal["kendaraan"] . "\n";
Print $personal["rumah"]."\\n";
?>
```

### Count() dan Sizeof()

Kedua fungsi ini memiliki kegunaan yang sama, yaitu menghitung jumlah elemen yang dimiliki oleh sebuah array.

Sintaksnya adalah : count(\$array)

Sizeof(\$array)

### Latihan 1.

```
<?php
$a[0] =1;
$a[1] =3;
$a[2] =5;
$jumlah =count($a);
print "Jumlah array a = $jumlah <br>";
// variabel $jumlah akan bernilai 3
$b["buah"] = "semangka";
$b["sayur"] ="wortel";
$b["daging"] ="ayam";
$b["utama"] ="nasi";
$jumlah = sizeof($b);
print "Jumlah array b = $jumlah <br>";
// variabel $jumlah akan bernilai 4
?>
```

### Latihan 2.

```
<html>
<body>

<?php
$x=array("one","two","three");
foreach ($x as $value)
{
    echo $value . "<br />";
}
?>

</body>
</html>
```

### Latihan 3.

```
<?
$UsiaKaryawan["Lisa"] = "28";
$UsiaKaryawan["Jack"] = "16";
$UsiaKaryawan["Ryan"] = "35";
$UsiaKaryawan["Rachel"] = "46";
$UsiaKaryawan["Grace"] = "34";
foreach($UsiaKaryawan as $Nama => $umur)
{
    echo "Nama Karyawan: $Nama, Usia: $umur"." th <br>";
}
?>
```

## 5.2 Fungsi dalam Array

### Fungsi is\_array()

Fungsi `is_array()` digunakan untuk memeriksa apakah sebuah variabel merupakan array atau bukan. Jika sebuah variabel merupakan array, fungsi ini akan menghasilkan nilai `true` dan jika bukan array akan menghasilkan nilai `false`. Sintaksnya adalah sebagai berikut:

### Is\_array(variabel)

---

#### Latihan 4.

```
<HTML>
<HEAD>
<TITLE> Penggunaan Is Array </TITLE>
</HEAD>
<BODY>

<?php
$var = array(1,2,3,4,5,6,7);
$scan = is_array($var);
if ($scan == false) {
    $status = "bukan";
} else {
    $status = "";
}
echo "\$var = array(1,2,3,4,5,6,7) ";
echo "<br>";
echo "Variabel \$var $status merupakan array";
?>

</BODY>
</HTML>
```

#### Fungsi list()

Fungsi list() digunakan untuk mengambil komponen-komponen array sebagai variabel-variabel yang terpisah. Sintaksnya adalah sebagai berikut:

List(\$item1, \$item2, ... , \$itemn) = array

Jumlah variabel \$item harus sama dengan atau kurang dari jumlah komponen yang dimiliki array.

---

#### Latihan 5.

```
<HTML>
<HEAD>
<TITLE> Penggunaan List </TITLE>
</HEAD>
<BODY>

<?php
$program = array('Bobo','Doraemon','Spiderman');
list($Majalah, $Komik, $Film) = $program;
echo "Jenis Buku & Hiburan :";
echo "<br />";
```

```
echo "Cerpen : $Majalah";  
echo "<br />";  
echo "Cerita Bergambar : $Komik";  
echo "<br />";  
echo "Bioskop : $Film";  
?>  
  
</BODY>  
</HTML>
```

#### Latihan 6.

```
<HTML>  
<HEAD>  
<TITLE> Penggunaan List 2 </TITLE>  
</HEAD>  
<BODY>  
  
<?php  
$program = array('Bobo','Doraemon','Spiderman');  
list($Majalah, , $Film) = $program;  
echo "Jenis Buku & Hiburan :";  
echo "<br />";  
echo "Cerpen : $Majalah";  
echo "<br />";  
echo "Bioskop : $Film";  
?>  
  
</BODY>  
</HTML>
```

#### Fungsi split()

Fungsi split digunakan untuk memecah suatu string menjadi array berdasarkan karakter pemisah tertentu. Sintaksnya adalah sebagai berikut:

```
split(karakter, teks, [batas])
```

Karakter adalah karakter yang digunakan untuk memisahkan array.

Teks adalah string yang akan dipecah menjadi array.

Batas adalah jumlah komponen yang akan dihasilkan.

---

### Latihan 7.

```
<HTML>
<HEAD>
<TITLE> Penggunaan Split </TITLE>
</HEAD>
<BODY>

<?php
$tanggal = "17-05-2010";
list($hari, $bulan, $tahun) = split("-", $tanggal);
echo "Hari = $hari";
echo "<br />";
echo "Bulan = $bulan";
echo "<br />";
echo "Tahun = $tahun";
?>

</BODY>
</HTML>

list($bulan, $hari, $tahun) = split("/.-]", $tanggal);
```

Fungsi split() identik dengan fungsi explode().

### Fungsi join()

Fungsi ini pada prinsipnya adalah kebalikan fungsi split(), yaitu digunakan untuk mengumpulkan komponen-komponen array menjadi satu string. Sintaksnya adalah sebagai berikut:

```
join(karakter, array)
```

Karakter adalah karakter yang digunakan untuk “merekatkan” komponen-komponen array.

---

### Latihan 8.

```
<HTML>
<HEAD>
<TITLE> Penggunaan Join </TITLE>
</HEAD>
<BODY>
<?php
$var = array('18', '11', '2010');
```

```
$tanggal = join("/", $var);  
echo "$tanggal";  
?>  
</BODY>  
</HTML>
```

Fungsi join() identik dengan fungsi implode().

### **Fungsi in\_array()**

Fungsi in\_array() digunakan untuk memeriksa apakah suatu nilai tertentu terdapat sebagai komponen di dalam sebuah array. Sintaksnya adalah sebagai berikut:

```
in_array(cari, array, [tipe])
```

Cari adalah nilai yang akan dicari apakah terdapat pada sebuah array. Tipe adalah boolean yang mengindikasikan apakah tipe data akan disertakan dalam pencarian. Jika suatu nilai ditemukan tapi tipe data tidak sama, maka fungsi in\_array() dianggap gagal. Fungsi ini bersifat case-sensitive.

### **Latihan 9.**

```
<HTML>  
<HEAD>  
<TITLE> Penggunaan In Array </TITLE>  
</HEAD>  
<BODY>  
  
<?php  
$program = array("HTML", "PHP", "CSS", "JavaScript");  
print_r($program);  
$cari = "HTML";  
if (in_array($cari, $program)) {  
    echo "Program Basis Web $cari ada di dalam array";  
} else {  
    echo "Program Basis Web $cari tidak ada di dalam  
array";  
}  
?>  
  
</BODY>  
</HTML>
```



### Latihan 10.

```
<HTML>
<HEAD>
<TITLE> Penggunaan In Array dengan Type Data </TITLE>
</HEAD>
<BODY>
<?
$tipe = array('1.10', 5.0, 1.13);
if (in_array('5.0', $tipe, TRUE)) {
    echo "String \"5.0\" ada di dalam array";
} else {
    echo "String \"5.0\" tidak ada di dalam array";
}
echo "<br />";
if (in_array(1.13, $tipe, TRUE)) {
    echo "Bilangan 1.13 ada di dalam array";
} else {
    echo "Bilangan 1.13 tidak ada di dalam array";
}
?>
</BODY>
</HTML>
```

---

## BAB VI FUNCTION

### 6.1 Penulisan Function

Fungsi pada dasarnya terdiri dari blok-blok program yang dapat dipanggil, kemudian dimasukkan langsung ke dalam kode program utama, dan memanggil nama fungsinya. Hal penting dari fungsi adalah fungsi dapat menerima value dari luar, atau dikenal dengan parameter, value ini kemudian dapat diolah dan hasilnya dapat digunakan di luar fungsi.

Manfaat yang dapat diambil ketika menggunakan fungsi adalah :

1. Kode program lebih mudah dibaca
2. Jika terdapat kesalahan mudah dikoreksi
3. Perintah atau prosedur dalam program dapat langsung diubah dari fungsinya, tanpa harus mengubah program utama, dengan demikian akan meningkatkan efisiensi dalam membuat program.

```
function namafungsi ($parameter1, $parameter2)
{
    body fungsi;
}
```

**Syarat memberikan nama fungsi :**

1. Awali setiap pembuatan fungsi dengan kata function
2. Tidak boleh sama dengan function yang telah ada dalam PHP
3. Nama fungsi dapat berupa angka, huruf, garis bawah atau campuran antara huruf dan angka, namun huruf pertama tidak boleh angka
4. Tidak boleh diawali dengan angka
5. Penamaan fungsi tidak bersifat case sensitif

Jenis fungsi dalam PHP terbagi menjadi 3 jenis yaitu :

1. Built-In
2. User Defined Function (UDF)
3. Eksternal

---

**Fungsi Built-In** adalah fungsi yang sudah disediakan oleh PHP dan pemrogram dapat langsung menggunakannya, contoh fungsi ini adalah :

- ✓ Fungsi Matematik : sin, cos, tan, asin, dan sebagainya
- ✓ Fungsi Tanggal dan Waktu : date, checkdate
- ✓ Fungsi String : strlen, strpos, strtolower, strtoupper, substr, dan lain sebagainya

**Fungsi UDF** merupakan fungsi yang dibuat sendiri oleh pemrogram. Fungsi ini dapat diletakkan dimana saja dalam script PHP, tapi sebaiknya diletakkan paling atas agar memudahkan dalam proses debugging dan tahap pembuatan dan pengembangan program.

**Fungsi External** merupakan daftar fungsi yang belum diaktifkan dalam keseluruhan modul PHP, fungsinya diletakkan dalam library atau file yang terpisah dari modul PHP. Fungsi external ini dapat diaktifkan secara otomatis dengan meregristasikan pada file PHP.INI.

### Latihan 1.

```
<?php
function basic($argument)
{
    echo $argument;
}
basic('hello world!'); // outputs 'hello world!'
?>
```

### Latihan 2.

```
<html>
<head><title>Contoh Penggunaan UDF</title></head>
<body>
<!-- Menentukan Form Input -->
<form >
Masukkan Bilangan Pertama : <br>
<input type="text" name="A" size=10> <br>
Masukkan Bilangan Kedua : <br>
<input type="text" name="B" size=10> <br>
<input type="submit" value="hitung">
</form>
<!--membandingkan 2 buah bilangan yang diinput-->
```

```
<?
$a=$_post["A"];
$b=$_post["B"];
Function jumlah($A,$B)
{
    $jumlahbil=$A + $B;
    Return $jumlahbil;
}

Function kurang($A,$B)
{
    $kurangbil=$A - $B;
    Return $kurangbil;
}

Function kali($A,$B)
{
    $kalibil=$A * $B;
    Return $kalibil;
}

Function bagi($A,$B)
{
    $bagibil=$A / $B;
    Return $bagibil;
}

Echo "<br>";
Echo ("Bilangan Pertama : ");
Echo $A;
Echo "<br>";
Echo ("Bilangan Kedua : ");
Echo $B;
Echo "<br> <br>";

Echo "Hasil Penjumlahan 2 buah bilangan ";
Echo "<br>";
$jumlahbil=&jumlah($A,$B);
Printf( "Penjumlahan antara : %d + %d = %d
", $A, $B, $jumlahbil);
Echo "<br><br>";
Echo "Hasil Pengurangan 2 buah bilangan ";
Echo "<br>";
$kurangbil=&kurang($A,$B);
Printf( "Pengurangan antara : %d - %d = %d
", $A, $B, $kurangbil);
Echo "<br><br>";
```

---

```
Echo "Hasil Perkalian 2 buah bilangan ";
Echo "<br>";
$kalibil=&kali($A,$B);
Printf( "Perkalian antara :  %d * %d = %d ", $A,
$B, $kalibil);
Echo "<br><br>";
Echo "Hasil Pembagian 2 buah bilangan ";
Echo "<br>";
$bagibil=&bagi($A,$B);
Printf( "Pembagian antara :  %d / %d = %d
", $A, $B, $bagibil);
Echo "<br><br>";
?>
</body>
</html>
```

### Latihan 3.

```
<?php
function repeat($text, $num = 10)
{
    echo "<ol>\r\n";
    for($i = 0; $i < $num; $i++)
    {
        echo "<li>$text </li>\r\n";
    }
    echo "</ol>";
}
// calling repeat with two arguments
repeat("I'm the best", 15);
// calling repeat with just one argument
repeat("You're the man");
?>
```

## 6.2 Jenis-jenis fungsi Buit-In

### ✓ Fungsi Tanggal

Fungsi date dapat digunakan untuk memperoleh tanggal atau jam pada sistem (server).

Syntax : **Date(string\_format)**

Tabel di bawah ini memperlihatkan bagaimana format waktu yang akan dihasilkan jika nilai argumen yang dimasukkan sebagai berikut :

Argumen	Nilai hasil
a	"am" atau "pm"
A	"AM" atau "PM"
d	Tanggal sekarang dari "01" sampai dengan "31"
D	Nama hari dari "Sun" sampai dengan "Sat"
F	Nama bulan dari "January" sampai dengan "December"
g	Jam sekarang dari "1" sampai dengan "12"
G	Jam sekarang dari "0" sampai dengan "23"
h	Jam sekarang dari "01" sampai dengan "12"
H	Jam sekarang dari "00" sampai dengan "23"
i	Menit sekarang dari "00" sampai dengan "59"
I	Daylight saving time : "1" jika True, "0" jika false
J	Tanggal sekarang dari "1" sampai dengan "31"
I	Nama hari dari "Sunday" sampai dengan "Saturday"
L	Tahun kabisat : "1" jika True, "0" jika false
m	Kode bulan dari "01" sampai dengan "12"
M	Nama bulan dari "Jan" sampai dengan "Dec"
N	Kode bulan dari "1" sampai dengan "12"

---

R	Format tanggal RFC 822
s	Detik sekarang dari "00" sampai dengan "59"
S	Akhiran "th" atau "nd"
t	Jumlah hari pada bulan sekarang
T	Format timezone pada komputer, misalnya "Pacific Standard Time"
w	Kode hari dari "0" (Minggu) sampai dengan "7" (Sabtu)
y	Tahun sekarang dalam 2 digit
Y	Tahun sekarang dalam 4 digit
z	Tanggal dalam tahun dari "1" sampai dengan "365"

#### Latihan 4.

```
<HTML>
<HEAD>
<TITLE> Tanggal </TITLE>
</HEAD>
<BODY>
<font size="10px">

<?php
echo "Sekarang tanggal ";
echo date('d-F-Y');
echo "<br>dan jam ";
echo date('h:i:s A');
?>

</FONT>
</BODY>
</HTML>
```

### Fungsi getdate()

Fungsi getdate() digunakan untuk mengambil nilai waktu lokal sekarang atau waktu timestamp dan waktu memasukkannya ke dalam array asosiatif. Sintaksnya adalah sebagai berikut:

```
getdate([timestamp])
```

Elemen array yang dapat dipergunakan adalah sebagai berikut:

1. "minutes" = menit
2. "seconds" = detik
3. "mday" = hari dalam satu bulan
4. "hours" = jam, dalam format 24 jam.
5. "wday" = hari dalam satu minggu, numeris, 0 untuk minggu hingga 6 untuk sabtu
6. "mon" = bulan, numeris.
7. "year" = tahun, numeris.
8. "yday" = hari dalam satu tahun, misalnya "299"
9. "weekday" = hari dalam satu minggu, tekstual penuh, misalnya "Friday"
10. "month" = bulan, tekstual penuh, misalnya "January"

#### Latihan 5.

```
<HTML>
<HEAD>
<TITLE> Getdate </TITLE>
</HEAD>
<BODY>
<center>
<h1>
<?php
$sekarang = getdate();
$bulan = $sekarang['month'];
$hari = $sekarang['mday'];
$tahun = $sekarang['year'];
$jam = $sekarang['hours'];
if ($jam <= 11) {
    echo "Selamat Pagi";
} elseif ($jam > 11 and $jam <= 15) {
    echo "Selamat Siang";
} elseif ($jam > 15 and $jam <= 18) {
    echo "Selamat Sore";
} elseif ($jam > 18) {
    echo "Selamat Malam";
```



```
}
?>
</h1>
<h2> Selamat datang</h2>
<h3> Sekarang adalah tanggal <?php echo "$hari $bulan
$tahun"; ?></h3>
</BODY>
</HTML>
```

### Fungsi String

- **Strlen(string)** : mengetahui jumlah karakter dalam string
- **Strtoupper(string)** : mengubah huruf kecil menjadi huruf besar
- **Strtolower(string)** : mengubah huruf besar menjadi huruf kecil
- **Ucfirst(string)** : mengubah huruf pertama string menjadi huruf besar
- **Ucwords(string)** : mengubah huruf pertama setiap kata menjadi huruf besar
- **Ltrim(string)** : menghapus spasi diawal string
- **Rtrim(string)** : menghapus spasi diakhir string
- **Trim(string)** : menghapus spasi baik diawal atau diakhir string
- **Substr(string, awal, jumlah)** : memotong string dimulai dari posisi *awal* sebanyak *jumlah* karakter
- **Substr\_count(string, substring)** : menghitung jumlah substring dalam string
- **Strpos(string, karakter)** : mengetahui posisi suatu karakter dalam string
- **Addslashes(string, parameter)** : menambahkan karakter backslash
- **Stripslashes(string)** : menghilangkan karakter backslash
- **Strip\_tags(string [, string tags yang tidak dihilangkan])** : menghilangkan kode tag HTML
- **Htmlentities(string)** : menghentikan proses penerjemahan tag html oleh browser
- **Strrev(string)** : membalik karakter yang terdapat pada string.
- **Str\_replace(yang diganti, pengganti, tujuan)** : mengganti string dengan string lain.
- **Strpos(tujuan, pencari)** : mengetahui posisi string dalam string

### **Fungsi Memanggil File Lain**

- **Fungsi require()**  
**Require("nama file.php")**
- **Fungsi include()**  
**Include("nama file.php")**
- **Fungsi include\_once()**  
**Include\_once("nama file.php")**

---

## BAB VII DATABASE

### 7.1 Operasi Database

Operasi data dalam database umumnya terdiri dari :

- ✓ Membuka koneksi dengan database server
- ✓ Memilih dan membuka database yang diinginkan
- ✓ Mengirim perintah untuk mengambil/menampilkan/mengubah/menghapus data.
- ✓ Mengakses hasil pengambilan data
- ✓ Mengakhiri koneksi

Query dikelompokkan menjadi tiga :

- ✓ Query untuk mengelola database
  - Membuat database  
`CREATE DATABASE <nama database>`
  - Menghapus database  
`DROP DATABASE <nama database>`
  - Membuat tabel  
`CREATE TABLE <nama tabel> (  
  <nama kolom> <type data>,  
  <nama kolom> <type data>,  
  ... ,  
  <nama kolom> <type data>  
) ;`
  - Memodifikasi tabel
  - Menghapus tabel  
`DROP TABLE <nama tabel>;`
  - Menambah user
  - Mengatur permission
  - Menghapus user

✓ Query untuk mengakses data dalam satu tabel

▪ Memasukkan data INSERT

```
INSERT INTO <nama tabel> (<nama kolom 1>, < nama
kolom 2>, ..)
VALUES (<isi kolom 1>, <isi kolom 2>, ..);
```

▪ Memodifikasi data UPDATE

```
UPDATE <nama tabel>
SET <nama kolom>=<isi kolom>
WHERE <kriteria>
```

▪ Mengambil data SELECT

```
SELECT <nama kolom>
FROM <nama tabel>
WHERE <kriteria>
```

▪ Menghapus data DELETE

```
DELETE FROM <nama tabel> WHERE <kriteria>
```

✓ Query yang melibatkan lebih dari satu tabel

Query banyak tabel digunakan untuk menghubungkan data di lebih dari satu tabel dengan menggunakan JOIN

## 7.2 Fungsi PHP – MySql

### ❖ MYSQL\_CONNECT()

Digunakan untuk menghubungkan ke server MySQL. Fungsi ini merupakan fungsi awal yang akan dijalankan sebelum melakukan fungsi-fungsi lain. Selain `mysql_connect()` terdapat juga fungsi pe nhubung ke server mysql lain , yaitu `mysql_pconnect()`.

Bentuk umum :

```
mysql_connect (host, user, password) ;
atau
mysql_pconnect (host, user, password) ;
```

---

## KONEKSI MYSQL

### Latih 1.

```
<?php
// hostname or ip of server
$servername='localhost';

// username and password
$dbusername='root';
$dbpassword='';
$link=mysql_connect
($servername,$dbusername,$dbpassword)
or die ( " Not able to connect to server ");
if ($link)
{
    echo "ok....koneksi berhasil";
}
?>
```

### ❖ MYSQL\_QUERY()

Berfungsi untuk memasukkan perintah-perintah SQL, baik yang bersifat DDL atau DML. Dengan demikian perintah-perintah SQL tersebut dapat dijalankan pada browser.

Bentuk umum :

```
mysql_query(perintah-perintah SQL);
```

## Membuat Database

### Latih 2.

```
<?
mysql_connect("localhost","root","");
$dbname="lat_dbase";
$ccek=mysql_query("CREATE DATABASE $dbname") or
die("Couldn't Create Database: $dbname");
if($ccek){
    echo "Database $dbname berhasil dibuat";
}
?>
```

### ❖ MYSQL\_SELECT\_DB()

Mysql\_select\_db() berfungsi untuk mengaktifkan database yang telah tersedia pada mysql server.

Bentuk umum :

```
mysql_select_db(nama_database);
```

### Membuat Tabel

#### Latih 3.

```
<?
mysql_connect("localhost","root",""); //koneksi
mysql_select_db("lat_dbase"); // mengaktifkan database
//membuat tabel
$sql = "CREATE TABLE tbl_mhs
(
mhsID int NOT NULL AUTO_INCREMENT,
PRIMARY KEY(mhsID),
FirstName varchar(15),
LastName varchar(15),
Age int
) ";
mysql_query($sql);
// input data
$input=mysql_query("insert into
tbl_mhs(FirstName,LastName,Age)
values('Anjar','Prabowo',25)");
?>
```

### Menyisipkan Data

#### Latih 4.

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("lat_dbase", $con);
```

---

```
mysql_query("INSERT INTO tbl_mhs (FirstName, LastName,
Age)
VALUES ('Karina', 'Suwandi', '29')");

mysql_query("INSERT INTO tbl_mhs (FirstName, LastName,
Age)
VALUES ('Glenn', 'Gandari', '32')");

mysql_close($con);
?>
```

### ❖ MYSQL\_DB\_QUERY()

Jika pada `mysql_query()` hanya berfungsi untuk memasukkan perintah-perintah SQL saja, berbeda halnya dengan `mysql_db_query()` yang dapat menyatukan dua fungsi, yaitu `mysql_select_db()` dan `mysql_query()`. Jika telah menggunakan fungsi `mysql_db_query()` tidak perlu lagi menggunakan fungsi `mysql_select_db()`.

Bentuk umum penulisannya:

```
mysql_db_query(nama_database, perintah-perintah SQL);
```

### ❖ MYSQL\_FETCH\_ROW()

Berfungsi untuk mengambil data per record atau per baris dalam suatu table. Yang perlu diperhatikan sifat pengambilan record pada fungsi `mysql_fetch_row()` adalah dengan system urutan nomor index field.

## Menampilkan Data

### Latih 5.

```
<?php
mysql_connect("localhost","root","");
// mysql_select_db("lat_dbase");
$hasil=mysql_db_query("lat_dbase","select * from
tbl_mhs");
While($data=mysql_fetch_row($hasil))
{
    echo "$data[0] $data[1] $data[2]<br>";
}
?>
```

---

### ❖ MYSQL\_FETCH\_ARRAY()

Mysql\_fetch\_array() mempunyai fungsi yang sama dengan mysql\_fetch\_row(), yaitu untuk mengambil data per-record. Yang membedakan adalah jika pada fungsi mysql\_fetch\_row() menggunakan nomor urutan index, tetapi pada fungsi mysql\_fetch\_array() yang digunakan adalah nama fieldnya.

#### Latih 6.

```
<?php
mysql_connect("localhost","root","");
mysql_select_db("lat_dbase");
$hasil=mysql_query("select * from tbl_mhs");
While($data=mysql_fetch_array($hasil))
{
    echo "$data[FirstName] $data[LastName]
$data[Age]<br>";
}
?>
```

### ❖ MYSQL\_NUM\_ROWS()

Digunakan untuk mengetahui berapa jumlah record terkini.

Bentuk umum :

```
Mysql_num_rows(variable);
```

#### Latih 7.

```
<?
mysql_connect("localhost","root","");
mysql_select_db("lat_dbase");
$hasil=mysql_query("select * from tbl_mhs");
$hit=mysql_num_rows($hasil);
echo "jumlah record $hit";
?>
```

### ❖ MYSQL\_CLOSE()

Menutup hubungan ke database MySql

Bentuk Umum : mysql\_close(variabel koneksi)



Memasukkan data dengan Form :

#### Latihan form.php

```
<html>
<body>

<form action="insert3.php" method="post">
Firstname: <input type="text" name="firstname">
Lastname: <input type="text" name="lastname">
Age: <input type="text" name="age">
<input type="submit">
</form>

</body>
</html>
```

#### Latihan insert3.php

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db("lat_dbase", $con);

$sql="INSERT INTO tbl_mhs (FirstName, LastName, Age)
VALUES
('$_POST[firstname]', '$_POST[lastname]', '$_POST[age]') "
;

if (!mysql_query($sql,$con))
{
    die('Error: ' . mysql_error());
}
echo "1 record added";

mysql_close($con)
?>
```

#### Soal Latihan :

1. Buatlah form buku tamu, tentukan field-fieldnya dan simpanlah dalam database
2. Tampilkanlah buku tamu tersebut
3. Cobalah tampilkan setiap halaman 5 record

---

## BAB VIII MANIPULASI DATABASE

### UPDATE

Digunakan untuk mengupdate record pada tabel.

#### Bentuk Umum :

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value
```

#### Latih 1.

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("lat_dbase", $con);

mysql_query("UPDATE tbl_mhs SET Age = '36'
WHERE FirstName = 'Karina' AND LastName = 'Suwandi'");

mysql_close($con);
?>
```

### DELETE

Digunakan untuk menghapus record dengan kriteria tertentu.

#### Bentuk Umum :

```
DELETE FROM table_name
WHERE some_column = some_value
```

---

### Latih 2.

```
<?php
$con = mysql_connect("localhost","root","");

if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("lat_dbase",$con);

mysql_query("DELETE FROM tbl_mhs WHERE
LastName='Prabowo'");
?>
```

### Menambah, mengedit dan menghapus artikel

#### Membuat Koneksi

```
<?
$dbhost = "localhost";
$dbuser = "root";
$dbpass = "";
$dbname = "artikel_db";

//lakukan koneksi dengan mysql
$connection = mysql_connect($dbhost,$dbuser,$dbpass);
if(!$connection)
{
    echo "Tidak dapat terhubung dengan database";
    exit;
}

//pilih database
$pilih_db = mysql_select_db($dbname);
if(!$pilih_db)
{
    echo "Tidak dapat memilih database";
    exit;
}
?>
```

### Membuat Form Isian Artikel

```
<?
include "koneksi.php";
?>
<h1>Form Berita</h1>
<form name=article method=post action=article.php>
  <table width="100%" border="0" cellspacing="0"
cellpadding="0">
    <tr>
      <td width="18%">Judul</td>
      <td width="2%">:</td>
      <td width="80%"><input type="text" name="title"
size="30" class="masukan"></td>
    </tr>
    <tr>
      <td>Penulis</td>
      <td>:</td>
      <td><input type="text" name="author" size="30"
class="masukan"></td>
    </tr>
    <tr>
      <td>Lead</td>
      <td>:</td>
      <td><textarea name="abstraksi" rows="4"
cols="45"></textarea></td>
    </tr>
    <tr>
      <td>Content</td>
      <td>:</td>
      <td><textarea name="content" rows="8" cols="45"
></textarea></td>
    </tr>
    <tr>
      <td>&nbsp;</td>
      <td>&nbsp;</td>
      <td>&nbsp;</td>
    </tr>
    <tr>
      <td>&nbsp;</td>
      <td>&nbsp;</td>
      <td><input type="submit" name="Submit"
value="Add" class="tombol">
      <input type="reset" name="batal" value="Reset"
class="tombol"></td>
    </tr>
```

```
</table>
</form>
```

### Proses Penambahan Artikel

```
<?
include "koneksi.php";

$judul= $_POST['title'];
$penulis = $_POST['author'];
$lead = $_POST['abstraksi'];
$isip = $_POST['content'];
$time=date("d M Y, H:i");

$lead = str_replace("\r\n", "<br>", $lead);
$content = str_replace("\r\n", "<br>", $content);

$query = "INSERT INTO articles
(judul, penulis, lead, content, waktu)
values('$judul', '$penulis', '$lead', '$isip', '$time')";
$result = mysql_query($query);

if($result){
    echo "<h3 align=center>Proses penambahan artikel
berhasil</h3>";
    echo "<A href=\"tampil_articles.php\">List</A>";
}else{
    echo "<h2 align=center>Proses penambahan artikel
tidak berhasil</h2>";
}
?>
```

### Membuat Tampilan Artikel

```
<?php
include "koneksi.php";
$perintah="SELECT * FROM articles ORDER BY
articleID DESC";
$hasil=mysql_db_query($dbname,$perintah);
echo("
<h2>List Artikel</h2>
<br><UL>
");

while ($row=mysql_fetch_array($hasil))
{
    echo("

```

---

```

        <LI>$row[1] &nbsp;$row[2] &nbsp;$row[waktu]
        &nbsp;<a
        href=\"edit_article.php?articleID=$row[0]\">Edit</
        a>
        &nbsp;<a
        href=\"delete.php?articleID=$row[0]\">Hapus</a></L
        I><br>");
    }
    echo("</table>");
    echo "<br><a href=\"add_article.php\">Tambah</a>";
    echo "<br><a href=\"adminpanel.php\">Admin
    Panel</a>";
?>

```

### Membuat Edit artikel

```

<?
include "koneksi.php";
$articleID = $_GET['articleID'];
// Query satu record artikel dari tabel sesuai nilai
$articleID
$perintah="SELECT * FROM articles WHERE articleID
=\"$articleID\"";
$hasil=mysql_db_query($dbname,$perintah);
$row=mysql_fetch_array($hasil);
$time=date("d M Y, H:i");

?>
<h1>Form Berita</h1>
<form name=article method=post
action=update_article.php>
    <input type="hidden" name="ID" value="<? echo
"$row[articleID]" ?>">
    <table width="100%" border="0" cellspacing="0"
cellpadding="0">
        <tr>
            <td width="18%">Judul</td>
            <td width="2%">:</td>
            <td width="80%"><input type="text" name="title"
size="50" class="masukan"
            value="<? echo "$row[judul]" ?>"></td>
        </tr>
        <tr>
            <td>Penulis</td>
            <td>:</td>
            <td><input type="text" name="author" size="50"
class="masukan"

```

---

```

        value="<? echo "$row[penulis]" ?>"></td>
    </tr>
    <tr valign="top">
        <td>Lead</td>
        <td>:</td>
        <td><textarea name="abstraksi" rows="4"
cols="50">
            <? echo "$row[lead]" ?></textarea></td>
    </tr>
    <tr valign="top">
        <td>Content</td>
        <td>:</td>
        <td><textarea name="content" rows="4" cols="50">
            <? echo "$row[content]" ?></textarea></td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
        <td>
            <input type="submit" name="masuk" value="Update"
class="tombol">
            <input type="reset" name="hapus" value="Cancel"
class="tombol"></td>
    </tr>
</table>
</form>

```

### Membuat Proses Edit Artikel

```

<?
include "koneksi.php";

$time=date("d M Y, H:i");
//$lead = str_replace("\r\n", "<br>", $lead);
//$content= str_replace("\r\n", "<br>", $content);
$update="UPDATE articles SET judul='$title',
penulis='$author', lead='$abstraksi',
content='$content', waktu='$time' WHERE articleID
='$ID'";
$hasil=mysql_db_query($dbname, $update);

```

```
if ($hasil) {
    echo "Artikel berhasil di update<br>";
    echo "<a href=\"tampil_articles.php\">List</a>";
} else {
    echo "Artikel gagal di update";
}
?>
```

### Membuat Delete Artikel

```
<?php
include "koneksi.php";
$articleID = $_GET['articleID'];

$perintah="DELETE FROM articles WHERE articleID
=\"$articleID\"";
$hasil= mysql_query ($perintah);

if ($hasil) {
    echo "Artikel berhasil dihapus<br>";
    echo "<a href=\"edit_articles.php\">Back</a>";
} else {
    echo "Komentar gagal dihapus. Kemungkinan terjadi
kegagalan koneksi
ke database MySQL.";
}
?>
```



---

## BAB IX SESSION DAN COOKIES

### 9.1 Session

Session merupakan hal yang cukup penting dalam aplikasi berbasis web. Dengan session memungkinkan programmer menyimpan informasi user secara semi-permanen, artinya selama masa tertentu informasi akan tersimpan. Penyimpanan isi variabel session berada di server, jadi tidak bisa diakses secara langsung oleh *client*.

Dalam aplikasi berbasis web, session banyak digunakan sebagai autentikasi login. Dengan session memungkinkan programmer mengatur siapa saja yang bisa mengakses suatu halaman. Misalnya saja, untuk melihat halaman kotak surat pada email, kita harus login terlebih dahulu. Dalam proses login antara lain akan terjadi pembuatan suatu session yang akan dibawa oleh user di setiap halaman. Di halaman kotak surat, session tersebut akan diperiksa. Jika session benar maka user dipersilahkan membuka halaman kotak surat, namun jika salah maka user tidak bisa membuka halaman kotak surat dan biasanya akan diminta untuk login terlebih dahulu. Itulah sebabnya, user tidak bisa mengakses halaman kotak surat secara langsung tanpa melakukan login.

Dalam penanganan session terdapat beberapa proses yang perlu diperhatikan :

- ☐ Proses pembuatan session
- ☐ Proses pemeriksaan session
- ☐ Proses penghapusan session

#### Membuat Session : session1.php

```
<?php
/*****
Halaman ini merupakan halaman contoh penciptaan
session. Perintah session_start() harus ditaruh di
perintah pertama tanpa spasi di depannya. Perintah
session_start() harus ada pada setiap halaman yang
berhubungan dengan session
*****/
session_start();
```

---

```
if (isset ($_POST['Login'])) {
$user = $_POST['user'];
$pass = $_POST['pass'];
//periksa login
if ($user == "rahadian" && $pass = "123") {
//menciptakan session
$_SESSION['login'] = $user;
//menuju ke halaman pemeriksaan session
echo "<h1>Anda berhasil LOGIN</h1>";
echo "<h2>Klik <a href='session2.php'>di sini
(session2.php)</a>
untuk menuju ke halaman pemeriksaan session";
}
} else {
?>
<html>
<head>
<title>Login here...</title>
</head>
<body>
<form action="" method="post">
<h2>Login Here...</h2>
Username : <input type="text" name="user"><br>
Password : <input type="password" name="pass"><br>
<input type="submit" name="Login" value="Log In">
</form>
</body>
</html>
```

### **Pemeriksaan Session : session2.php**

```
<?php
/*****
Halaman ini merupakan contoh halaman pemeriksaan
session. Pemeriksaan session biasanya dilakukan jika
suatu halaman memiliki akses terbatas. Misalnya harus
login terlebih dahulu.
*****/
session_start();
//pemeriksaan session
if (isset($_SESSION['login'])) {
//jika sudah login
//menampilkan isi session
echo "<h1>Selamat Datang ". $_SESSION['login']
."</h1>";
```

---

```
echo "<h2>Halaman ini hanya bisa diakses jika Anda
sudah login</h2>";
echo "<h2>Klik <a href='session3.php'>di sini
(session3.php)</a> untuk LOGOUT</h2>";
} else {
//session belum ada artinya belum login
die ("Anda belum login! Anda tidak berhak masuk ke
halaman
ini.Silahkan login <a href='session1.php'>di
sini</a>");
}
?>
```

### **Penghapusan Session : session3.php**

```
<?php
/*****
Halaman ini merupakan halaman logout, dimana kita
menghapus session yang ada.
*****/
session_start();
if (isset($_SESSION['login'])) {
unset ($_SESSION);
session_destroy();
//
echo "<h1>Anda sudah berhasil LOGOUT</h1>";
echo "<h2>Klik <a href='session1.php'>di sini</a> untuk
LOGIN kembali</h2>";
echo "<h2>Anda sekarang tidak bisa masuk ke halaman
<a href='session2.php'>session2.php</a> lagi</h2>";
}
?>
```

## **9.2 Cookies**

Seperti halnya session, cookies juga merupakan sebuah konsep penyimpanan informasi user. Hanya saja, jika session tempat penyimpanan berada di server, cookies berada di client. Oleh karena itu, konsep cookies sebaiknya jangan digunakan untuk menyimpan informasi login user seperti username, password dsb. Selain user bisa melihat informasi yang disimpan, user juga bisa *men-disable*

---

cookies itu sendiri. Jika cookies di *-disable*, maka program yang memanfaatkan cookies tentunya tidak akan berjalan dengan baik.

Cookies sendiri biasanya dipakai dalam aplikasi *shopping cart*. Biasa digunakan untuk menyimpan sementara, produk-produk yang dipilih oleh pengunjung pada saat berbelanja. Dalam penanganan cookies juga terdapat beberapa proses yang perlu diperhatikan :

- ☐ Proses pembuatan cookies
- ☐ Proses pemeriksaan cookies
- ☐ Proses penghapusan cookies

#### **Pembuatan Cookies : cookie1.php**

```
<?php
$value = 'rahadian';
$value2 = 'rahadi ramelan';
setcookie("username", $value);
setcookie("namalengkap", $value2, time()+3600); /*
expire in 1
hour */
echo "<h1>Ini halaman pengesetan cookie</h1>";
echo "<h2>Klik <a href='cookie2.php'>di sini</a> untuk
pemeriksaan cookies</h2>";
?>
```

#### **Pemeriksaan Cookies : cookie2.php**

```
<?php
if(isset($_COOKIE['username'])) {
echo "<h1>Cookie 'username' ada. Isinya : " .
$_COOKIE['username'];
} else {
echo "<h1>Cookie 'username' TIDAK ada.</h1>";
}
if(isset($_COOKIE['namalengkap'])) {
echo "<h1>Cookie 'namalengkap' ada. Isinya : " .
$_COOKIE['namalengkap'];
} else {
echo "<h1>Cookie 'namalengkap' TIDAK ada.</h1>";
}
echo "<h2>Klik <a href='cookie1.php'>di sini</a> untuk
penciptaan cookies</h2>";
echo "<h2>Klik <a href='cookie3.php'>di sini</a> untuk
```

```
penghapusan cookies</h2>";  
?>
```

### **Penghapusan Cookies : cookie3.php**

```
<?php  
// set the expiration date to one hour ago  
setcookie ("username", "", time() - 3600);  
setcookie ("namalengkap", "", time() - 3600);  
echo "<h1>Cookie Berhasil dihapus.</h1>";  
echo "<h2>Klik <a href='cookie01.php'>di sini</a> untuk  
penciptaan cookies</h2>";  
echo "<h2>Klik <a href='cookie02.php'>di sini</a> untuk  
pemeriksaan cookies</h2>";  
?>
```

---

## BAB X HANDLING ERRORS

**Beberapa pesan kesalahan yang terjadi :**

**Pesan Error:**

[view sourceprint?](#)

```
1.Warning: mysql_fetch_array(): supplied argument is not  
2.a valid MySQL result resource in ....
```

atau

[view sourceprint?](#)

```
1.Warning: mysql_num_rows(): supplied argument is not  
2.a valid MySQL result resource in ...
```

**Penyebab:**

Error di atas disebabkan ada query SQL yang salah. Bisa jadi karena sintaks perintah query SQL nya salah, misalnya kurang tanda petik untuk menyatakan string. Selain itu bisa juga terjadi karena nama field dan tabel yang salah penulisannya.

**Solusi:**

Coba cek kembali statement query SQL. Pastikan tidak ada lagi kesalahan dalam penulisannya. Untuk proses debugging query SQL (mengetahui letak kesalahannya), bisa menambahkan statement `die(mysql_error())` pada perintah `mysql_query()` nya. Contoh:

[view sourceprint?](#)

```
1.$query = ".....";  
2.mysql_query($query) or die(mysql_error());  
3..  
4..
```

---

**Pesan Error:**

[view sourceprint?](#)

```
1.Warning: session_start() [function.session-start]:  
2.Cannot send session cookie - headers already sent by  
3.(output started at ... ) in ... on line ...
```

atau

[view sourceprint?](#)

```
1.Warning: session_start() [function.session-start]:  
2.Cannot send session cache limiter - headers already sent  
3.(output started at ...) in ... on line ...
```

### Penyebab:

Error di atas biasanya terjadi pada penggunaan session. Sewaktu memberikan perintah `session_start()`, terdapat baris perintah di atasnya yang menampilkan string atau kode html ke browser. Sebagai contoh, perhatikan script berikut ini:

[view sourceprint?](#)

```
1.<?php  
2.echo "Hallo...";  
3.session_start();  
4.?>
```

atau

[view sourceprint?](#)

```
1.<html>  
2.<head>  
3.    <title>....</title>  
4.</head>  
5.<?php  
6.session_start();  
7.?>
```

Perhatikan script di atas, maka terdapat string yang di-echo-kan atau terdapat kode html yang di letakkan sebelum `session_start()`. Nah... i inilah yang menyebabkan error. Selain contoh di atas, error juga bisa muncul dari script yang berbentuk seperti ini

[view sourceprint?](#)

```
1.<?php  
2.include "header.php";  
3.session_start();  
4.?>
```

Meskipun sebelum `session_start()` tidak ada echo atau kode html, namun error akan tetap muncul apabila isi dari script header.php terdapat echo atau kode html juga, misalnya

---

### header.php

[view sourceprint?](#)

```
1.<?php
2.echo "hallo";
3.?>
```

#### Solusi:

Pastikan sebelum perintah `session_start()` tidak terdapat perintah `echo` atau kode html apapun.

---

#### Pesan Error:

[view sourceprint?](#)

```
1.Warning: Cannot modify header information - headers already sent
  by
2.(output started at ...) in ...
```

#### Penyebab:

Peringatan di atas biasanya muncul ketika melakukan redirecting menggunakan perintah

[view sourceprint?](#)

```
1.header("Location: ...");
```

Seharusnya, sebelum perintah `header("Location: ...")` tidak boleh ada string apapun yang ditampilkan di browser, meskipun itu kode HTML (sama seperti penyebab error di `session_start()` sebelumnya). Sebagai contoh perhatikan script berikut ini:

[view sourceprint?](#)

```
1.<?php
2.echo "<p>Hallo Apa kabar?</p>";
3.header("Location: test.php");
4.?>
```

Pada script di atas, sebelum perintah `header()` terdapat string yang ditampilkan ke browser. Inilah yang mengakibatkan muncul peringatan tersebut. Bagaimana dengan script berikut ini?



[view sourceprint?](#)

```
1.<?php
2.$a = 10;
3.if ($a < 0) echo "Nilai A negatif";
4.else header("Location: test.php");
5.?>
```

Apakah muncul Warning atau tidak? Jika di coba, maka tidak muncul Warning nya meskipun sebelum `header()` terdapat string yang di tampilkan ke browser. Mengapa demikian? meskipun di situ terdapat perintah untuk menampilkan string ke browser, namun perintah tersebut tidak pernah dijalankan karena nilai `$a` nya positif.

**Solusi:**

Pastikan sebelum perintah `header("Location: ...")` tidak ada kode HTML atau perintah `echo` apapun. Jika meng-include-kan suatu script di atas `header("Location: ...")` pastikan dalam script yang di-include-kan tersebut tidak ada kode HTML atau `echo`.

---

**Pesan Error:**

[view sourceprint?](#)

```
1.Warning: mysql_connect() [function.mysql-connect]:
2.Access denied for user 'root'@'localhost' (using password: YES)
in ...
```

**Penyebab:**

Error di atas di sebabkan karena koneksi yang gagal ke MySQL. Penyebab gagalnya bisa jadi disebabkan nama user atau password yang salah.

**Solusi:**

Pastikan username dan password koneksi ke MySQL sudah benar ketika menggunakan perintah koneksi `mysql_connect("namahost", "namauser", "passworduser");`

---

---

**Pesan Error:**

[view sourceprint?](#)

```
1.Parse error: syntax error, unexpected T_STRING,  
2.expecting T_OLD_FUNCTION or T_FUNCTION or T_VAR or `}' in ...
```

**Penyebab:**

Penyebab pastinya, saya kurang mengetahuinya. Namun biasanya ini terjadi karena penggunaan versi PHP yang rendah (versi di bawah PHP 5) untuk menjalankan script PHP yang menggunakan OOP.

**Solusi:**

Coba upgrade versi PHP, atau gunakan PHP dengan versi yang lebih tinggi.

---

**Pesan Error:**

[view sourceprint?](#)

```
1.Parse error: syntax error, unexpected T_ELSE in ...
```

**Penyebab:**

Error tersebut muncul sebagai akibat penulisan sintaks IF ... ELSE yang tidak sesuai aturan. Perhatikan contoh penggunaan IF ELSE yang salah di bawah ini

[view sourceprint?](#)

```
1.<?php  
2.$a = 5;  
3.if ($a > 0) $status = "A lebih besar dari 0"  
4.else ($a < 0) $status = "A lebih kecil dari 0"  
5.else $status = "A sama dengan 0"  
6.??>
```

Contoh di atas terdapat kesalahan yaitu setiap akhir statement tidak diberikan tanda titik koma (;), dan pada bagian `else ($a < 0)` kurang IF, seharusnya

[view sourceprint?](#)

```
1.<?php  
2.$a = 5;  
3.if ($a > 0) $status = "A lebih besar dari 0";  
4.else if ($a < 0) $status = "A lebih kecil dari 0";  
5.else $status = "A sama dengan 0";  
6.??>
```

### Solusi:

Pastikan struktur statement yang digunakan berbentuk seperti ini

[view sourceprint?](#)

```
1.<?php
2.if (syarat1) statement1;
3.else if (syarat2) statement2;
4.else if (syarat3) statement3;
5..
6..
7.else statementX;
8.?>
```

atau

[view sourceprint?](#)

```
01.<?php
02.if (syarat1)
03.{
04.    statement;
05.    statement;
06.    .
07.    .
08.}
09.else if (syarat2)
10.{
11.    statement;
12.    statement;
13.    .
14.    .
15.}
16..
17..
18.else
19.{
20.    statement;
21.    statement;
22.    .
23.    .
24.}
25.?>
```

---

### Pesan Error:

[view sourceprint?](#)

```
1.Fatal error: Call to undefined function: ...
```

### Penyebab:

Error tersebut disebabkan karena nama function yang digunakan tidak ada atau tidak dikenal oleh PHP. Jika function tersebut adalah built in function, misalnya `str_split()` bisa jadi hal ini disebabkan pada versi PHP yang digunakan belum ada function tersebut. Jika function tersebut adalah function buatan sendiri, bisa jadi lupa menyisipkan ke dalam script atau meng-include-kannya.

### Solusi:

Jika function yang tidak dikenal adalah built in function, maka lakukan upgrade versi PHP yang lebih tinggi. Sedangkan jika function yang tidak dikenal adalah buatan sendiri, pastikan sudah meng-includekan di dalam scriptnya sehingga bisa dikenal. Contohnya:

#### function.php

[view sourceprint?](#)

```
1.<?php
2.function jumlah($a, $b)
3.{
4.    return $a + $b;
5.}
6.?>
```

#### script.php

[view sourceprint?](#)

```
1.<?php
2.include "function.php";
3.echo jumlah(9, 10);    // akan menghasilkan 19
4.?>
```

atau

#### script.php

[view sourceprint?](#)

```
1.<?php
2.function jumlah($a, $b)
3.{
4.    return $a + $b;
5.}
6.
7.echo jumlah(9, 10);    // akan menghasilkan 19
8.?>
```

### Pesan Error:

[view sourceprint?](#)

```
1.Parse error: parse error, unexpected $end
```

### Penyebab:

Error di atas disebabkan karena kurangnya tanda penutup kurung kurawal.

Contohnya:

[view sourceprint?](#)

```
1.<?php
2.if ($a > 1)
3.{
4.    echo "Nilai A lebih dari 1";
5.?>
```

atau

[view sourceprint?](#)

```
1.<?php
2.while ($a > 1)
3.{
4.    .
5.    .
6.    .
7.?>
```

Selain itu, bisa pula error tersebut disebabkan karena efek perbedaan penggunaan tanda pembuka PHP `<?php` dan `<?.` Contohnya :

[view sourceprint?](#)

```
01.<?php
02.if (syarat)
03.{
04.    statement1;
05.    statement2;
06.?>
07.
08.<?
09.    statement3;
10.    statement4;
11.}
12.?>
```

Perhatikan contoh script di atas, bahwa pada blok statement IF dipecah menjadi 2 blok, namun perhatikan bahwa untuk blok kedua, tanda pembukanya menggunakan `<?`. Saja, terkadang suatu server hanya membolehkan tanda

---

pembuka script PHP menggunakan `<?php`, sehingga untuk blok kedua tidak dianggap sebagai script, dan akibatnya statement `IF` nya tidak ada tanda `}` penutup.

### Solusi:

Pastikan tanda kurung kurawal penutup tidak lupa menuliskannya. Jika ternyata kasus yang dihadapi sama seperti contoh kasus terakhir (perbedaan tanda pembuka script PHP), konsistensikan penulisan tanda pembukanya menggunakan `<?php` supaya aman.

---

### Pesan Error:

[view sourceprint?](#)

```
1.Notice: Undefined index: ...
```

atau

[view sourceprint?](#)

```
1.Notice: Undefined variable: ...
```

### Penyebab:

Sebenarnya itu bukan error, namun sekedar peringatan saja bahwa ada variabel atau index array yang belum terdefinisi sebelumnya. Di beberapa versi PHP peringatan tersebut muncul, namun di versi yang lain tidak muncul. Berikut ini contohnya:

[view sourceprint?](#)

```
1.<?php
2.if ($a > 0)
3.{
4. ....
5.}
6.?>
```

Perhatikan script di atas, variabel `$a` langsung diberikan statement pengecekan `IF`, padahal nilai `$a` ini tidak ada sebelumnya, di beberapa versi PHP, akan muncul peringatan

[view sourceprint?](#)

```
1.Notice: Undefined variable: a
```

Namun, bila nilai \$a ini sudah didefinisikan sebelumnya misalnya

[view sourceprint?](#)

```
1.<?php
2.$a = 10;
3.if ($a > 0)
4.{
5. ....
6.}
7.?>
```

maka peringatan di atas tidak akan muncul lagi. Peringatan yang hampir sama juga bisa muncul untuk contoh berikut ini

[view sourceprint?](#)

```
1.<?php
2.if ($_GET['test'] == 0)
3.{
4. ....
5.}
6.?>
```

yang dikarenakan nilai \$\_GET['test'] belum terdefinisi sebelumnya atau belum ada nilainya (belum ada request GET untuk parameter 'test').

### Solusi:

Sebenarnya peringatan di atas bisa saja diabaikan karena tidak berpengaruh pada kinerja script atau proses yang terjadi di dalamnya. Namun, jika ingin menghilangkan peringatan tersebut caranya adalah dengan memodifikasi parameter

[view sourceprint?](#)

```
1.error_reporting = ...
```

pada file konfigurasi **php.ini**

Tambahkan nilai "& ~E\_NOTICE" pada nilai parameternya, contoh:

[view sourceprint?](#)

```
1.error_reporting = E_ALL & ~E_NOTICE
```

---

Maksud dari parameter di atas adalah, PHP akan menampilkan Pesan error kecuali NOTICE. Jika sudah, silakan merestart Apache nya.

---

**Pesan Error:**

[view sourceprint?](#)

```
1.Fatal error: Maximum execution time of ... seconds exceeded in ...
```

**Penyebab:**

Error di atas disebabkan karena lama eksekusi sebuah script PHP telah mencapai batas maksimumnya. Jika menggunakan Apache atau XAMPP, default *maximum execution time* nya diset 30 detik. Jika lama eksekusi script yang dibuat melebihi 30 detik, maka akan muncul error di atas.

**Solusi:**

Untuk meningkatkan atau menambahkan *maximum execution time* pada file `php.ini`, dengan cara carilah parameter

[view sourceprint?](#)

```
1.max_execution_time = ...
```

Tambahlah nilai lainya, misalnya menjadi 9999 detik, sehingga nilai parameter nya menjadi

[view sourceprint?](#)

```
1.max_execution_time = 9999
```

Selanjutnya, simpan kembali `php.ini` nya, lalu restart Apache nya. Kalau perlu, lakukan analisis algoritma pada script apabila dirasa tidak efisien sehingga mengakibatkan running time yang lama.

**Pesan Error:**

Bukan Error, tapi kegagalan dalam proses upload file ketika membuat script upload file.



**Penyebab:**

Kegagalan dalam proses upload file itu disebabkan oleh ukuran file yang terlalu besar melebihi batas maksimum yang diperbolehkan oleh PHP sesuai di file php.ini.

**Solusi:**

Ubah batas maksimum file size untuk upload ini dengan mengubah parameter

[view sourceprint?](#)

```
1.upload_max_filesize = ...
```

dan

[view sourceprint?](#)

```
1.post_max_size = ...
```

pada file php.ini

Berikut ini contoh pengubahan batas maksimum upload file hingga 80 Mb.

[view sourceprint?](#)

```
1.upload_max_filesize = 80M
```

dan

[view sourceprint?](#)

```
1.post_max_size = 80M
```

Jika sudah melakukan pengubahan, lakukan restart Apache nya, jangan lupa pula untuk menambah komponen input

[view sourceprint?](#)

```
1.<input type="hidden" name="MAX_FILE_SIZE" value="80000000" >
```

pada form upload nya.

Solusi di atas hanya akan bisa menyelesaikan permasalahan upload jika file diupload ke folder atau direktori tertentu yang ada di server. Namun, jika file diupload ke database server, maka solusi di atas tidak menyelesaikan masalah karena adanya batasan limit file size pada tipe data yang memungkinkan untuk menyimpan data file, misalnya tipe data blob, medium blob, atau long blob.

---

## DAFTAR PUSTAKA

1. <http://www.w3schools.com/php/default.asp>
2. <http://www.freewebmasterhelp.com/tutorials/phpmysql/>
3. <http://php.about.com/>
4. <http://www.howstuffworks.com/dns.htm>
5. <http://www.co.cc/>
6. <http://www.w3c.org>
7. Dasar Pemrograman Web Dinamis Menggunakan PHP, Abdul Kadir, Andi Yogyakarta, 2002