

# DASAR PEMROGRAMAN PHP

## A. PENGENALAN

### Mengenal PHP

PHP merupakan kependekan dari Hypertext Preprocessor yaitu bahasa pemrograman web yang dapat disisipkan dalam skrip HTML. PHP bekerja disisi server. Tujuan dari bahasa ini adalah membantu para pengembangan web untuk membuat web dinamis dengan cepat.

### Aplikasi yang diperlukan

Agar dapat menjalankan PHP kita harus menyediakan perangkat lunak berikut:

- Web Server (Apache, IIS, Personal Web Server/PWS)
- PHP Server
- Database Server (MySQL, Interbase, MS SQL, dll)

Kita tidak perlu menginstal satu-satu dari aplikasi-aplikasi di atas, karena biasanya telah tersedia paket aplikasi yang telah menyediakan ketiganya dalam satu paket aplikasi seperti Xampp atau Appserv.

Jika kita menggunakan Xampp, maka skrip PHP kita tulis di dalam folder htdocs. Namun, jika kita menggunakan Appserv, maka skrip PHP di tulis di dalam folder www.

## B. PENULISAN SINTAX PHP

### Sintax dasar PHP

Skrip PHP dituliskan di antara tanda berikut:

```
<?php
.....
?>
```

### Menyimpan PHP

File PHP harus disimpan dengan extensi .php dan disimpan di server. Jika disimpan dengan format .html, maka tidak dapat diproses dan ditampilkan di web server seperti apa adanya, yaitu berupa kode-kode.

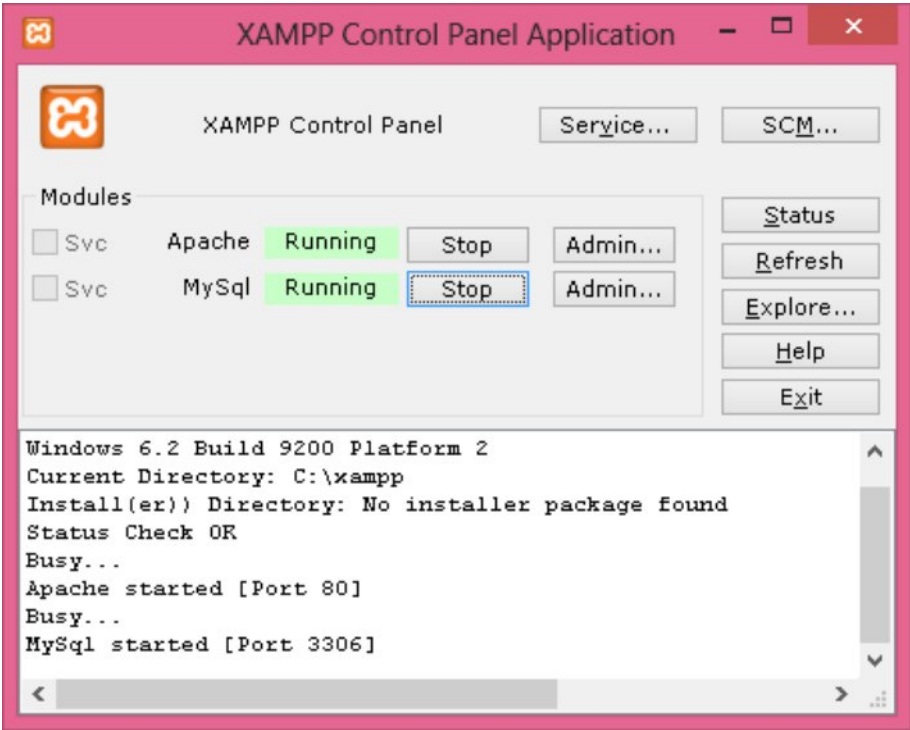
### Semicolon (;)

Setiap baris skrip PHP harus diakhiri dengan tanda semicolon (;). Jika tidak, maka akan menampilkan pesan error.

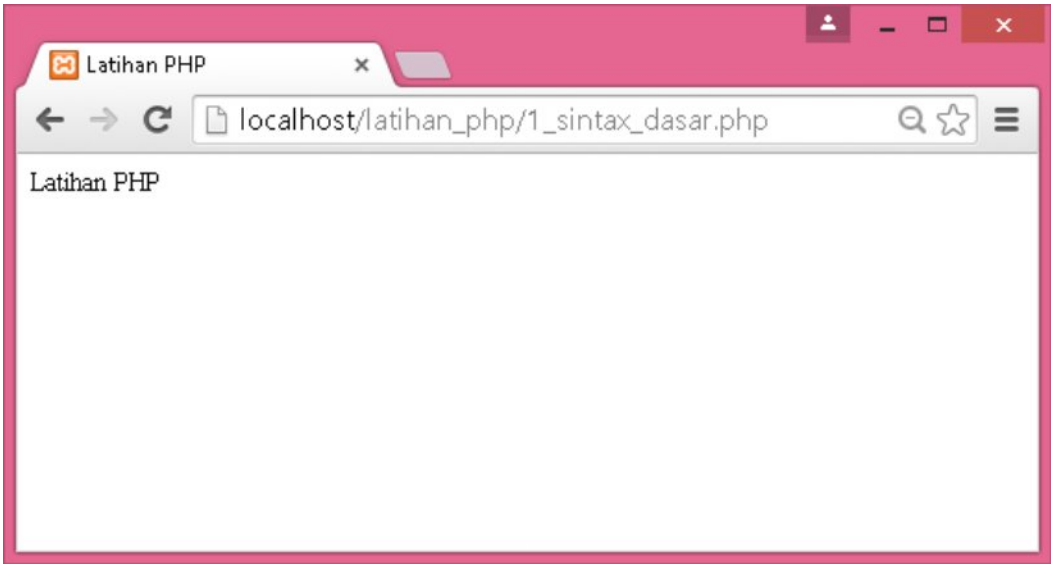
```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4     <title>Latihan PHP</title>
5 </head>
6 <body>
7     <?php
8         echo "Latihan PHP";
9     ?>
10 </body>
11 </html>
```

Menampilkan file PHP

Untuk menampilkan hasil skrip PHP di atas, tidak dapat dilakukan seperti saat membuka file HTML yaitu doble klik pada file. File PHP hanya bisa dibuka dengan memanggilnya melalui web browser dan mengaktifkan service Apache dan MySQL terlebih dahulu. Cara memanggilnya cukup dengan mengetikkan localhost/nama\_folder/nama\_file.php pada address bar browser.



Gambar 1 Mengaktifkan service Apache dan MySQL



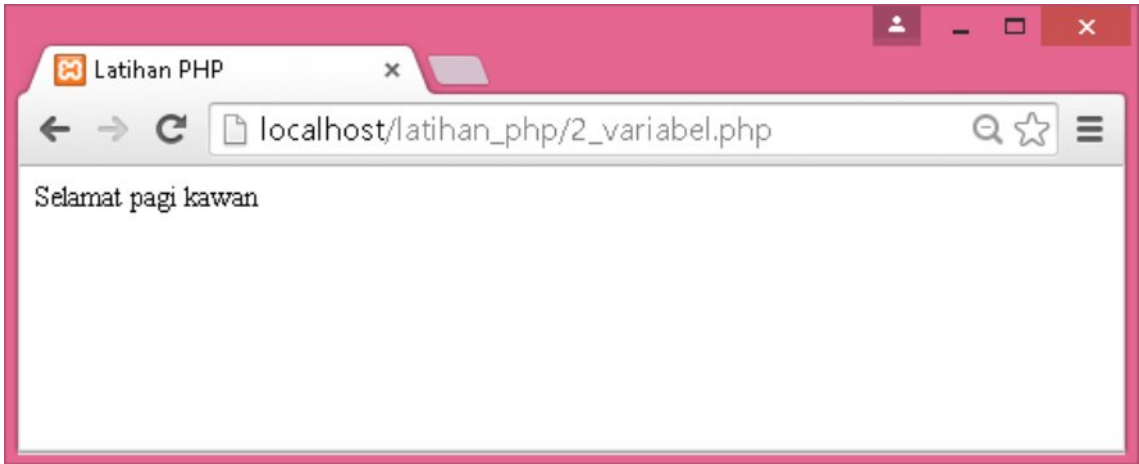
Variabel

Jika kita ingin menyimpan suatu nilai berupa angka atau string yang nilainya dapat berubah-ubah, maka kita dapat membuat variabel terlebih dahulu. Membuat variabel tidak bisa asal kata, tetapi harus memenuhi aturan berikut:

- Nama variabel harus diawali dengan huruf atau underscore ( \_ )
- Nama variabel hanya boleh dituliskan dengan huruf, angka dan underscore
- Nama variabel yang terdiri lebih dari satu kata, dapat dipisahkan dengan underscore
- Nama variabel tidak boleh menggunakan kata yang merupakan bagian dari skrip PHP

Penulisan nama variabel harus diawali dengan tanda dolar (\$).

```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4      <title>Latihan PHP</title>
5  </head>
6  <body>
7      <?php
8          $kata = "Selamat pagi kawan";
9          echo $kata;
10     ?>
11 </body>
12 </html>
```



Menampilkan tulisan

Sebelumnya kita sudah pernah menampilkan tulisan yaitu dengan menggunakan echo. Tulisan yang ditampilkan dengan echo dapat berupa tulisan biasa atau skrip HTML. Ada beberapa hal penting yang perlu diperhatikan saat menggunakan echo yaitu sebagai berikut:

- Jangan menuliskan titik dua (") dalam text yang ditulis dengan echo
- Jika terpaksa harus menuliskan titik dua, maka berikan tanda slash (\) sebelum titik dua.
- Sebagai ganti tanda titik dua, Anda juga dapat menggunakan petik tunggal (').

```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4      <title>Latihan PHP</title>
5  </head>
6  <body>
7      <?php
8          echo " <p align=\"center\"> ini rata tengah </p> ";
9          echo " <p align='right'> in rata kanan </p> ";
10         ??
11     </body>
12 </html>
```

Komentar dengan PHP

Menuliskan komentar di dalam skrip PHP dapat dilakukan dengan dua cara. Jika komentar hanya satu baris, maka dapat menggunakan dua garis miring (//) atau tanda pagar (#). Jika komentar lebih dari satu baris, maka komentar diawali dengan tanda /\* dan diakhiri dengan tanda \*/.

```
1  <?php
2      #ini adalah komentar
3
4      // ini juga komentar
5
6      /* ini komentar
7         jika lebih dari satu baris */
```

C. OPERATOR

Setiap bahasa pemrograman mesti memiliki operator. Operator digunakan untuk memanipulasi atau melakukan proses perhitungan pada suatu nilai. Ada beberapa jenis operator dalam PHP yaitu:

- Operator aritmatika
- Operator perbandingan
- Operator logika

Operator Aritmatika

Operator aritmatika merupakan operator yang digunakan untuk melakukan perhitungan aritmatika.

Operator	Arti	Contoh
+	Penjumlahan	2 + 3 = 5
-	Pengurangan	6 – 3 = 3
*	Perkalian	4 * 3 = 12
/	Pembagian	9 / 3 = 3
%	Modulus / Sisa bagi	10 % 4 = 2

Operator Perbandingan

Operator perbandingan merupakan operator yang digunakan untuk membandingkan 2 nilai atau variabel. Hasil dari operator ini adalah true atau false.

Operator	Arti	Contoh	Hasil
==	Sama dengan	2 + 3 == 6	False
!=	Tidak sama dengan	2 + 3 != 6	True
>	Lebih besar dari	2 + 3 > 4	True
<	Lebih kecil dari	2 + 3 < 4	False
<=	Lebih kecil dari atau sama dengan	2 + 3 <= 5	True
>=	Lebih besar dari atau sama dengan	2 + 3 >= 5	True

Operator Logika

Operator logika digunakan untuk menggabungkan dua pernyataan atau lebih. Hasil dari

Operator	Arti	Keterangan
and	Dan	Hasilnya benar jika kedua pernyataan benar
or	Atau	Hasilnya benar jika salah satu pernyataan atau keduanya benar
!	tidak	Hasilnya benar jika pernyataan salah

D. MODULARITAS

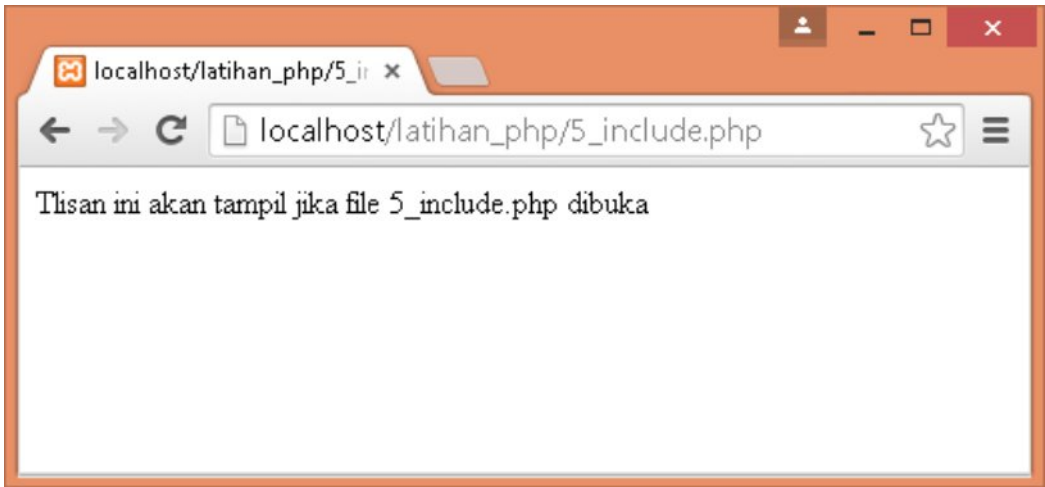
Dalam PHP kita dapat menyisipkan file PHP lain ke dalam suatu file PHP. Untuk menyisipkan file PHP lain, kita dapat menggunakan include() atau require(). Perbedaan dari kedua skrip tersebut terlihat jika terjadi error pada php. Jika menggunakan include() maka skrip setelah skrip yang error tetap dijalankan, sedangkan jika menggunakan require() skrip setelah skrip yang error tidak dieksekusi.

File 5\_modul.php

```
1 <?php
2     echo "Tlisan ini akan tampil jika file 5_include.php dibuka";
3 ?>
```

File 5\_include.php

```
1 <?php
2     include("5_modul.php");
3 ?>
```



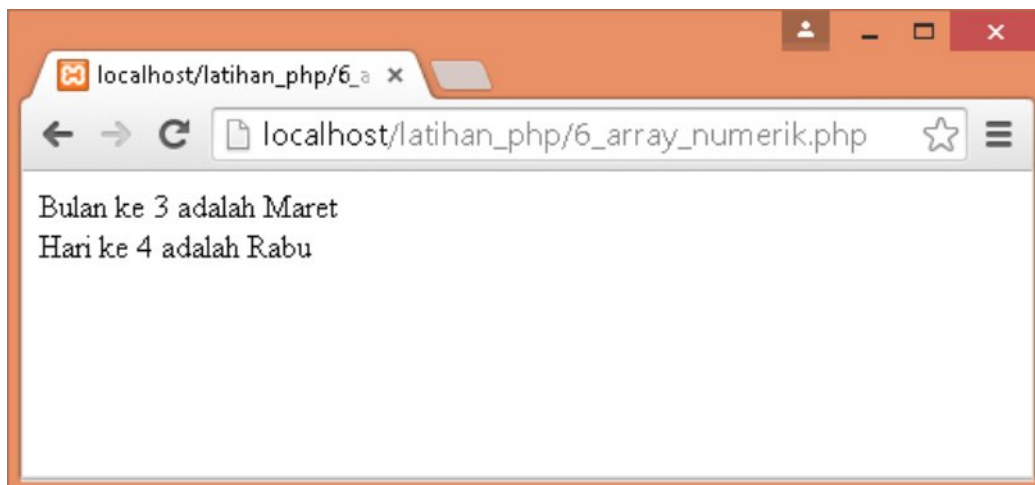
C. ARRAY

Array merupakan sebuah variabel yang dapat menyimpan nilai lebih dari satu. Array dapat berindex numerik yaitu diakses dengan menyebutkan angka index yang dimulai dari nol.

```

1 <?php
2 $bulan = array("Januari","Februari","Maret","April","Mei","Juni","Juli",
3               "Agustus","September","Oktober","November","Desember");
4
5 echo "Bulan ke 3 adalah $bulan[2]";
6
7
8 $hari[0] = "Minggu";
9 $hari[1] = "Senin";
10 $hari[2] = "Selasa";
11 $hari[3] = "Rabu";
12 $hari[4] = "Kamis";
13 $hari[5] = "Jum'at";
14 $hari[6] = "Sabtu";
15
16 echo "<br> Hari ke 4 adalah $hari[3]";
17

```



Array juga dapat berindex berupa string atau text. Array seperti ini biasa disebut array asosiatif.

```

1 <?php
2 // cara 1
3 $profil = array("nama" => "Nabil", "alamat" => "Tegal", "usia" => "4");
4
5 echo "Nama saya $profil[nama], saya berasal dari $profil[alamat],
6       usia saya $profil[usia] tahun";
7
8 // cara 2
9 $profil['nama'] = "Afaf";
10 $profil['alamat'] = "Banjarnegara";
11 $profil['usia'] = 5;
12
13 echo "<br>Nama saya $profil[nama], saya berasal dari $profil[alamat],
14       usia saya $profil[usia] tahun";
15

```



E. STATEMENT CONTROL

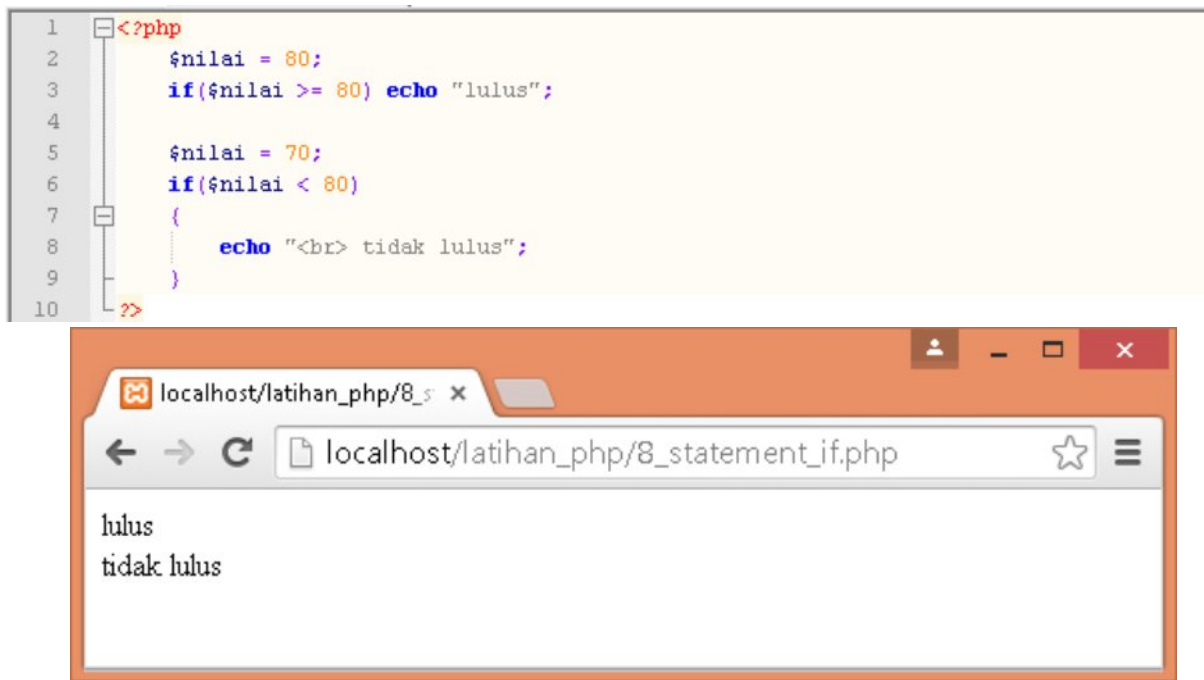
Statement If

Statent ini digunakan untuk menjalankan suatu program tertentu jika sebuah syarat terpenuhi. Format dari statement ini adalah:

```
If(syarat) statement;
```

Format di atas hanya berlaku jika statement hanya satu baris. Jika statment lebih dari satu, maka formatnya menjadi:

```
If(syarat)
{
Statement 1;
Statement 2;
...
}
```

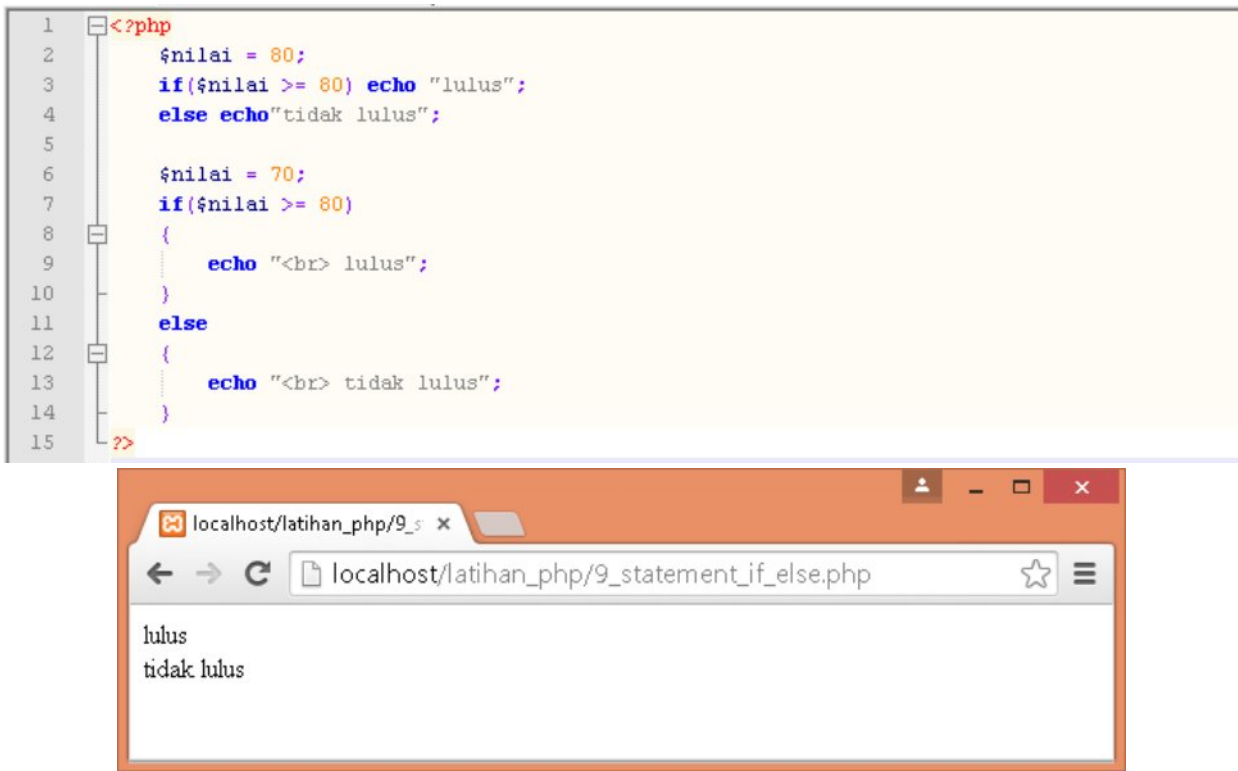


Statement if ... else

Ketika dalam kondisi tertentu, kita ingin menjalankan program tertentu jika suatu syarat tidak terpenuhi, maka kita dapat menggunakan statement ini. Formatnya sebagai berikut:

```
if(syarat)
{
    Statement jika benar;
}
Else
{
    Statement jika salah;
}
```

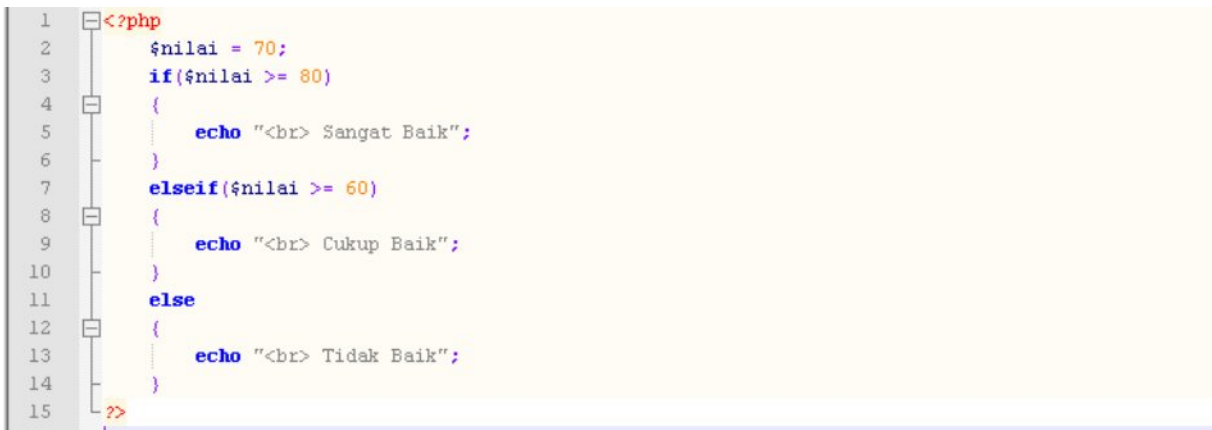




Statement if...elseif... else

Statement kontrol ini digunakan jika ketika syarat pertama tidak terpenuhi ada syarat kedua, ketiga atau seterusnya yang menjadi syarat alternatif. Formatnya sebagai berikut:

```
if(syarat)
{
    Statement jika syarat1 benar;
}
elseif(syarat2)
{
    Statement jika syarat2 benar;
}
else
{
    Statement jika semua syarat salah;
}
```



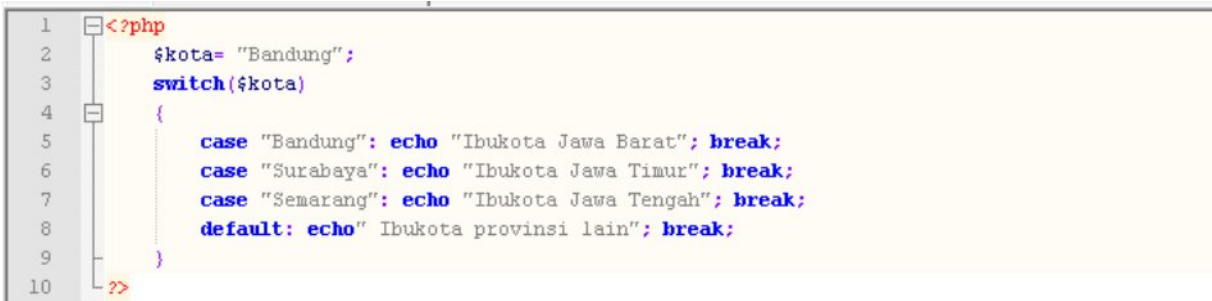


Statement switch

Statement ini digunakan untuk menjalankan program pada setiap kemungkinan nilai dari sebuah variabel. Formatnya sebagai berikut:

```
switch($variabel)
{
    case "option": statement;
    break;
    case "option": statement;
    break;
    default: statemtn;
    break;
}
```

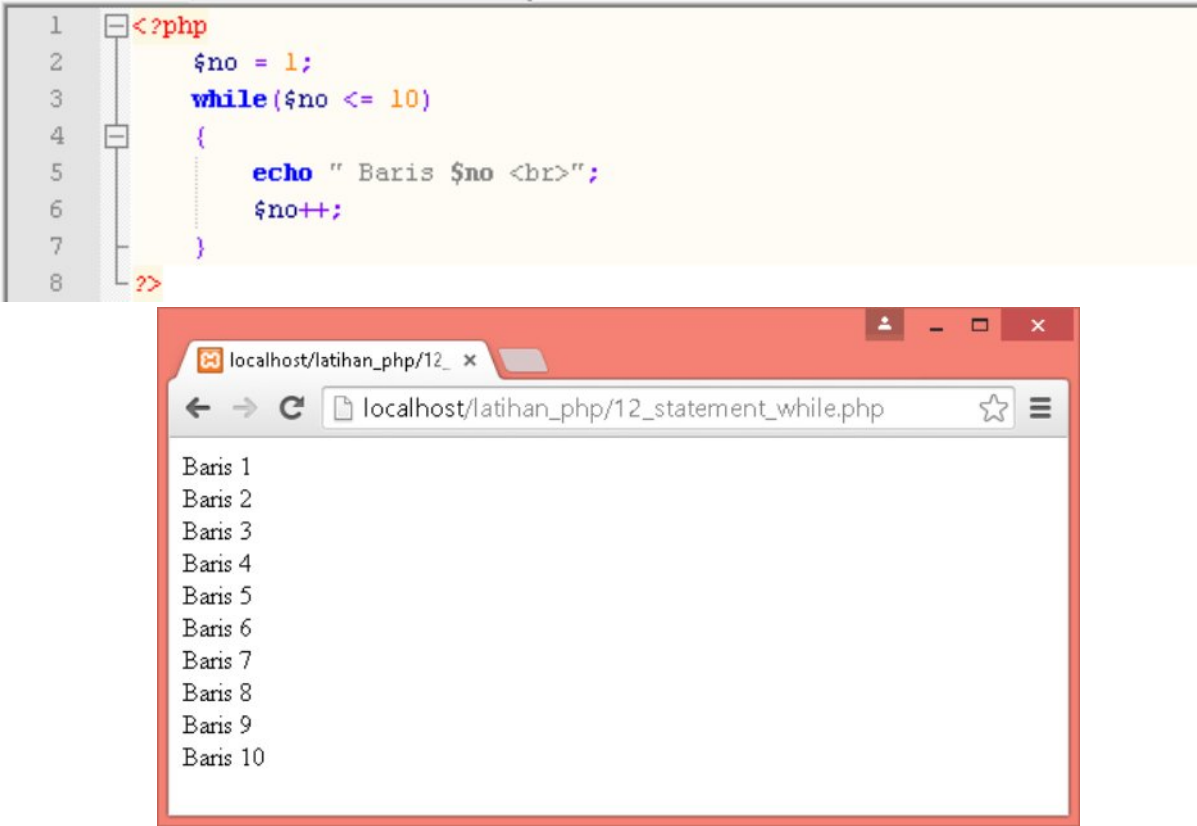
default dijalankan jika tidak ada option yang terpenuhi.



Statement while

While digunakan untuk menjalankan program secara berulang-ulang selama syarat masih terpenuhi. Format penulisanya sebagai berikut:

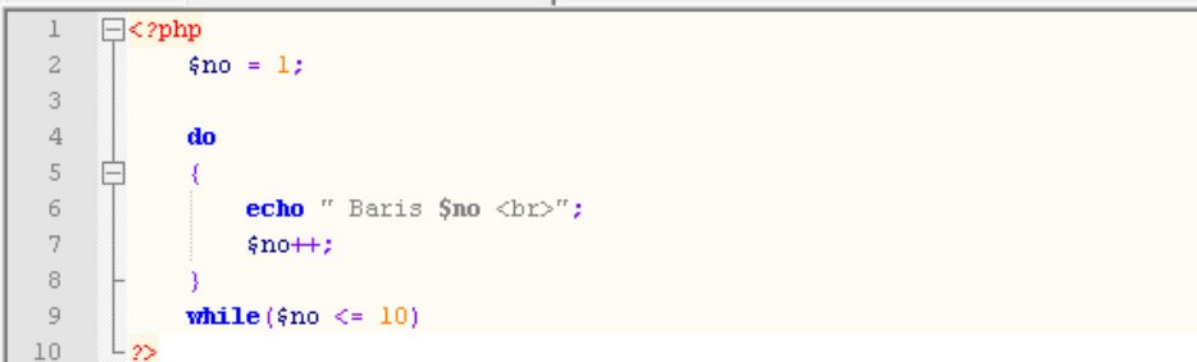
```
while(syarat)
{
    Statement;
}
```

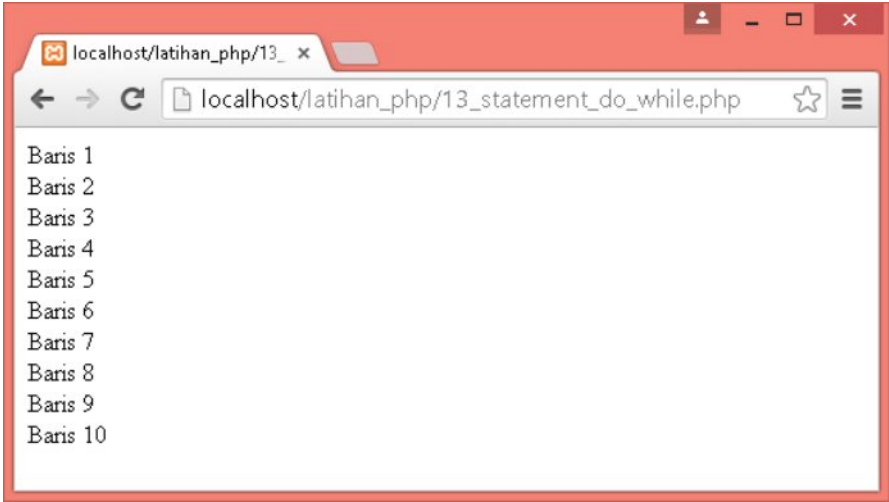


Statement do ... while

Statement ini hampir sama dengan statement while, tetapi syarat ditulis di akhir. Format penulisan sebagai berikut:

```
do
{
    Statement;
}
while(syarat);
```

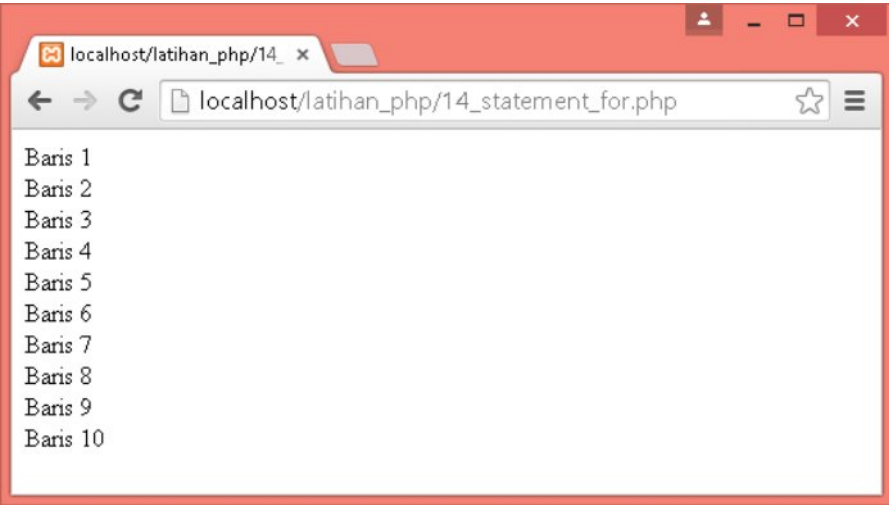




Statement for

Pada prinsipnya statement ini sama dengan while, tetapi pada statement ini batas awal dan akhir pengulangan langsung ditentukan. Formatnya sebagai berikut:

```
for(inisialisasi counter; syarat; increment/decrement counter)
{
    Statement;
}
```

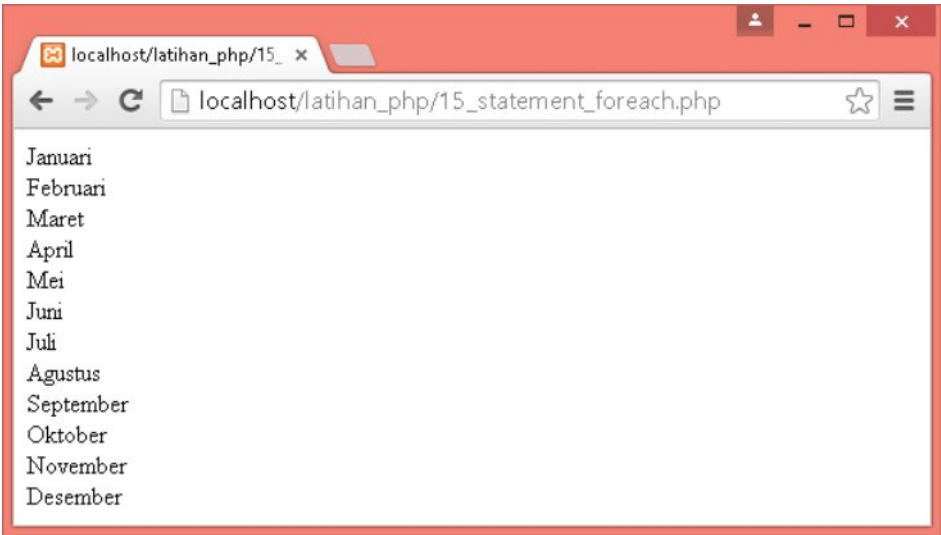


Statement foreach

Statement ini digunakan untuk melakukan perulangan data array.

```
Foreach(variabel_array as kunci)
{
    Statement;
}
```

```
1 <?php
2 $bulan = array("Januari","Februari","Maret","April","Mei","Juni","Juli",
3               "Agustus","September","Oktober","November","Desember");
4
5 foreach($bulan as $nama_bulan)
6 {
7     echo "$nama_bulan <br>";
8 }
9 ?>
```



F. GET DAN POST

GET digunakan untuk menangkap data yang dikirim oleh form dengan method get. Tetapi karena semua data yang dikirim ditampilkan pada URL (address bar), maka jarang digunakan pada form. GET lebih sering digunakan untuk mengirimkan data melalui link.

POST digunakan untuk menangkap data yang dikirim oleh form. Karena data yang dikirim tidak ditampilkan di URL, maka POST lebih aman dibanding GET, sehingga lebih sering digunakan pada form.

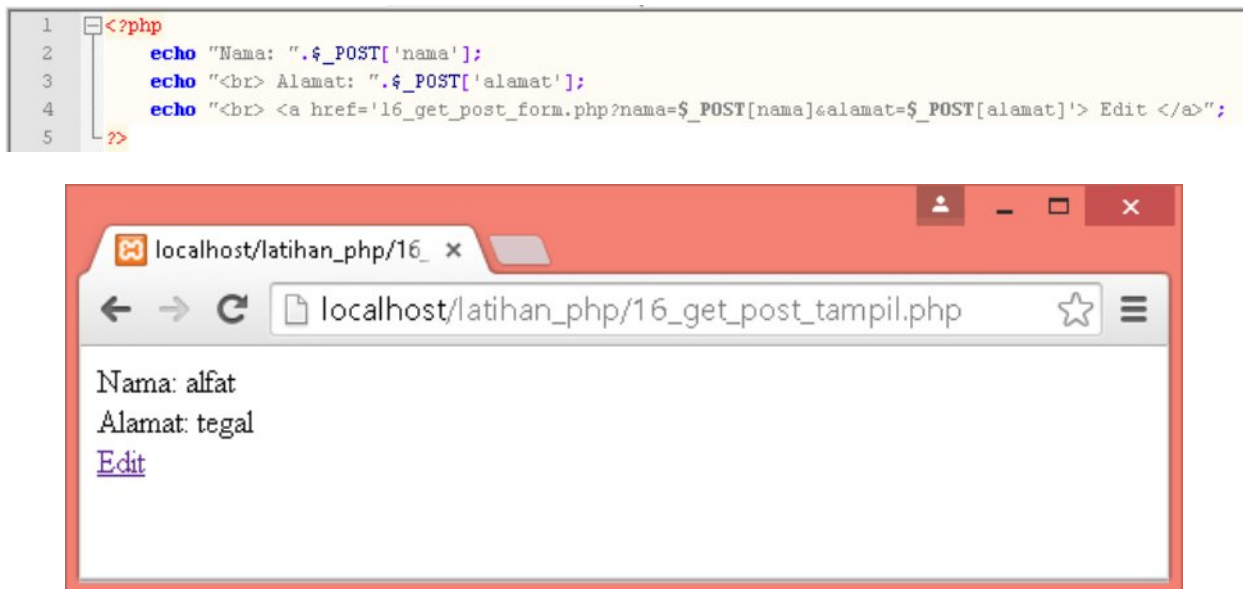
File 16\_get\_post\_form.php

```
1 <form action="16_get_post_tampil.php" method="post">
2     Nama: <input type="text" name="nama"
3           value="<?php if(isset($_GET['nama'])) echo $_GET['nama']; ?>"> <br>
4     Alamat: <input type="text" name="alamat"
5            value="<?php if(isset($_GET['alamat'])) echo $_GET['alamat']; ?>"> <br>
6     <input type="submit" value="Simpan">
7 </form>
```



Jika tombol simpan diklik pada form di atas, maka akan membuka file 16\_get\_post\_tampil.php.

File 16\_get\_post\_tampil.php

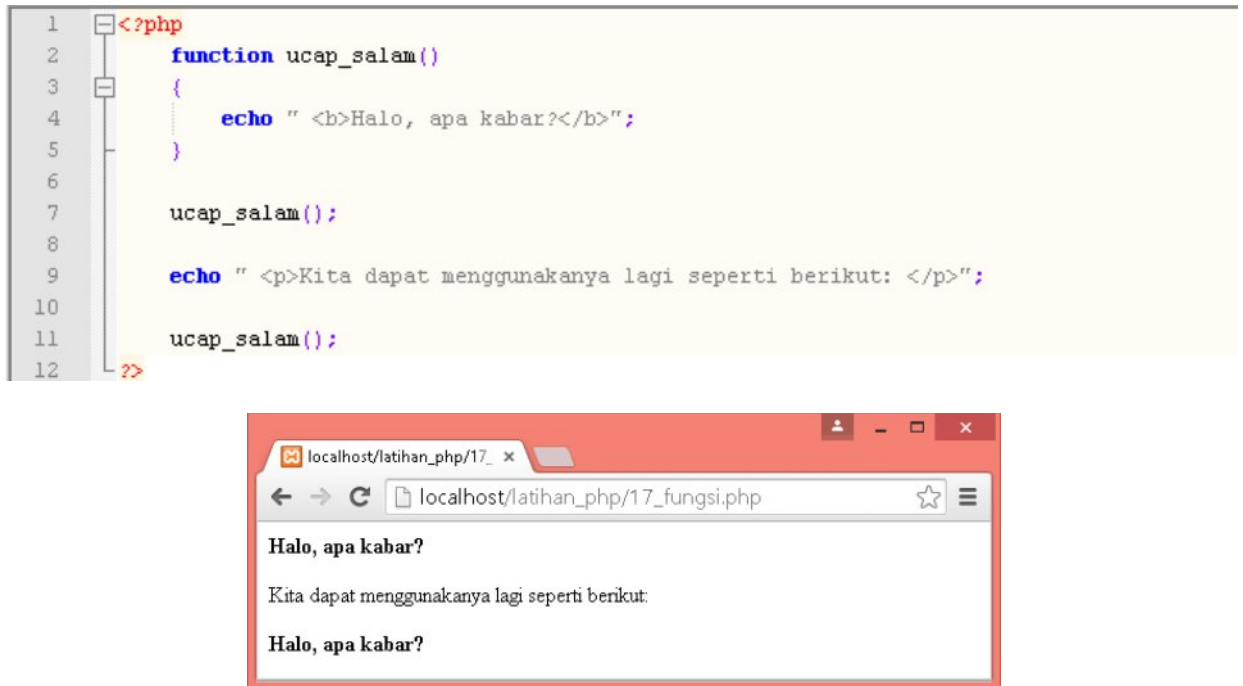


Jika link Edit diklik, maka data nama dan alamat yang dikirim ditangkap menggunakan GET pada file 16\_get\_post\_form.php.

G. FUNGSI

Fungsi dasar

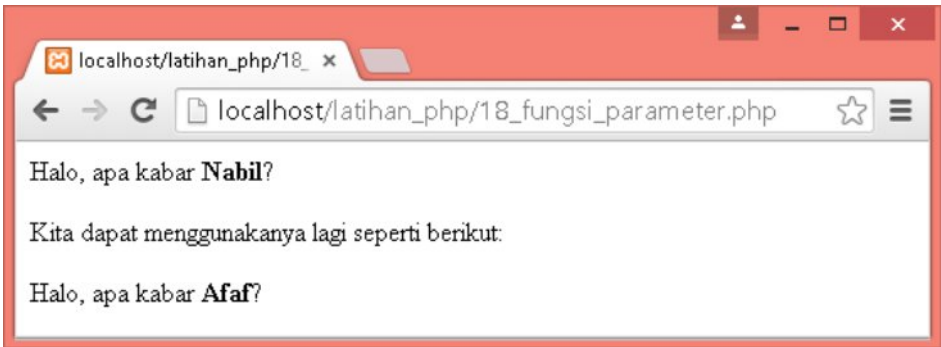
Fungsi digunakan untuk membuat program yang sering digunakan. Program yang sering digunakan cukup ditulis sekali dengan fungsi, fungsi tersebut dapat dipanggil setiap kali program tersebut akan digunakan.



Fungsi dengan parameter

Jika dalam penggunaan fungsi ingin menghasilkan tulisan yang berbeda-beda, tetapi dalam format yang sama, kita dapat menggunakan fungsi yang disertai parameter.

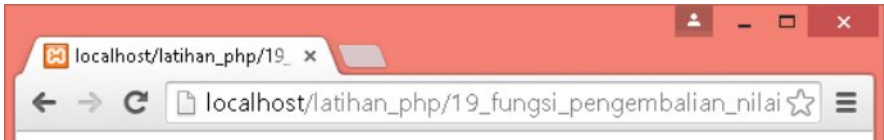
```
1 <?php
2     function ucap_salam($nama)
3     {
4         echo "Halo, apa kabar <b>$nama</b>?";
5     }
6
7     ucap_salam("Nabil");
8
9     echo " <p>Kita dapat menggunakannya lagi seperti berikut: </p>";
10
11    ucap_salam("Afaf");
12 >>
```



Pengembalian nilai

Sebuah fungsi dapat mengembalikan suatu nilai.

```
1 <?php
2     function hitung_luas($panjang, $lebar)
3     {
4         $luas = $panjang * $lebar;
5         return $luas;
6     }
7
8     echo "Luas ruangan adalah ".hitung_luas(10,8);
9
10    echo " <p>Kita dapat menggunakannya lagi seperti berikut: </p>";
11
12    $p = 15;
13    $l = 10;
14    $luas = hitung_luas($p, $l);
15    echo "Luas ruangan adalah $luas";
16
17 >>
```

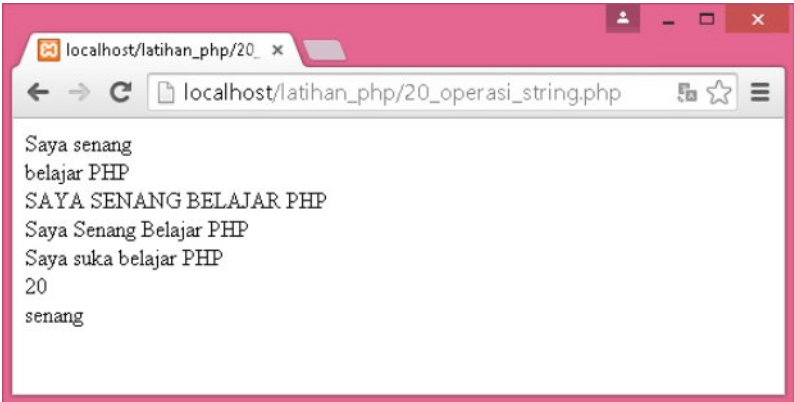


H. OPERASI STRING

Untuk memanipulasi sebuah text/string, kita dapat menggunakan fungsi-fungsi berikut:

Nama Fungsi	Keterangan
strpos()	Untuk mengambil nilai posisi suatu substring di dalam sebuah setring.
str_replace()	Untuk mengganti suatu kata dengan kata yang lain di dalam sebuah string.
strtoupper()	Untuk mengubah string ke huruf besar
strtolower()	Untuk mengubah string ke huruf kecil
ucwords()	Untuk mengubah huruf pertama pada setiap kata menjadi besar
explode()	Untuk memecah suatu string menjadi array
implode()	Untuk mengubah anggota array menjadi string (kebalikan explode())
substr()	Untuk mengambil sebagian string.

```
1 <?php
2     $text = "Saya senang belajar PHP";
3
4     echo substr($text,0,11);
5     echo "<br>";
6     echo substr($text,12);
7     echo "<br>";
8     echo strtoupper($text);
9     echo "<br>";
10    echo ucwords($text);
11    echo "<br>";
12    echo str_replace("senang","suka",$text);
13    echo "<br>";
14    echo strpos($text,"P");
15    $kata = explode(' ', $text);
16    echo "<br>";
17    echo $kata[1];
18
```





## I. OPERASI TANGGAL

### Fungsi date()

Fungsi ini digunakan untuk mencetak tanggal dengan format tertentu. Format tanggal ditulis di dalam kurung. Beberapa huruf yang dapat digunakan untuk menulis format tanggal yaitu sebagai berikut:

#### Time:

- a : am atau pm
- A : AM atau PM
- g : jam tanpa 0 di depan, bernilai antara 1 – 12
- G : jam tanpa 0 di depan, bernilai antara 0 – 23
- h : jam dengan 0 di depan, bernilai antara 1 – 12
- H : jam dengan 0 di depan, bernilai antara 00 – 23
- i : menit dengan 0 di depan, bernilai antara 00 – 59
- s : detik dengan 0 di depan, bernilai antara 00 – 59

#### Day:

- d : hari dalam bulan (tanggal) dengan 0 di depan, bernilai antara 01 – 31
- j : hari dalam bulan (tanggal) tanpa 0 di depan, bernilai antara 1 – 31
- D : hari dalam minggu (singkatan), bernilai antara Sun – Sat
- l : hari dalam minggu, bernilai antara Sunday – Saturday
- w : hari dalam minggu tanpa 0 di depan, bernilai antara 0 – 6
- z : hari dalam tahun tanpa 0 di depan, bernilai antara 0 – 365

#### Month:

- m : bulan dengan 0 di depan, bernilai antara 01 – 12
- n : bulan tanpa 0 di depan, bernilai antara 1 – 12
- M : singkatan dari bulan, bernilai antara Jan – Dec
- F : nama bulan lengkap, bernilai antara January – December
- t : jumlah hari dalam sebulan, bernilai antara 28 – 31

#### Year:

- L : 1 jika melompati tahun dan 0 jika tidak
- Y : tahun dengan 4 digit
- y : tahun dengan 2 digit (00 – 99)

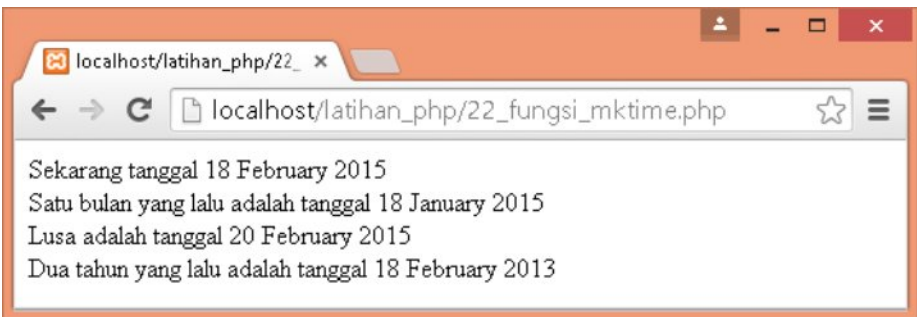


Fungsi mktime()

Fungsi ini digunakan untuk membuat timestamp dengan format sintaks sebagai berikut:

Mktime(hour, minute, second, month, day, year);

```
1 <?php
2 $sekarang = mktime(12, 13, 40, date('m'), date('d'), date('Y'));
3 echo "Sekarang tanggal ".date('d F Y', $sekarang);
4
5 $bulanlalu = mktime(0, 0, 0, date('m')-1, date('d'), date('Y'));
6 echo "<br>Satu bulan yang lalu adalah tanggal ".date('d F Y', $bulanlalu);
7
8 $lusa = mktime(0, 0, 0, date('m'), date('d')+2, date('Y'));
9 echo "<br>Lusa adalah tanggal ".date('d F Y', $lusa);
10
11 $tahun = mktime(0, 0, 0, date('m'), date('d'), date('Y')-2);
12 echo "<br>Dua tahun yang lalu adalah tanggal ".date('d F Y', $tahun);
13
```



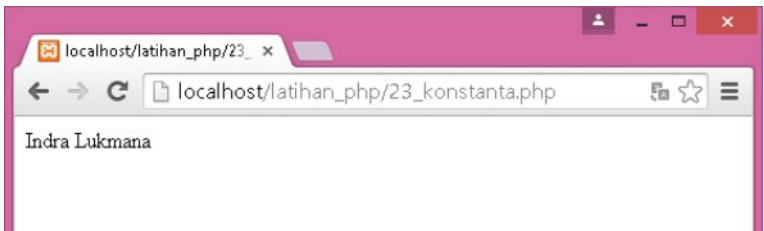
J. KONSTANTA

Konstanta sebenarnya hampir sama dengan variabel, namun dalam penggunaannya, konstanta tidak perlu menggunakan tand dolar (\$). Untuk membuat konstanta menggunakan fungsi define(), dengan format:

define(nama\_konstanta, nilai konstanta);

Untuk mengecek apakah ada konstanta telah didefinisikan atau belum menggunakan fungsi defined().

```
1 <?php
2 define("NAMA", "Indra Lukmana");
3
4 if(defined("NAMA")) echo NAMA;
5
```

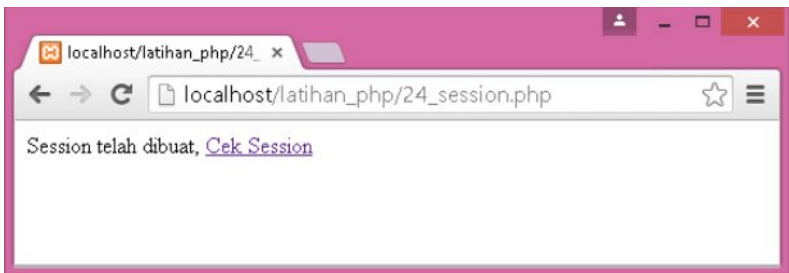


K. SESSION

Sama halnya dengan variabel, session juga digunakan untuk menyimpan suatu nilai. Bedanya jika session telah dibuat dalam suatu file, session dapat diakses oleh file lain. Session biasanya digunakan untuk pembuatan login user. Dengan session, seseorang dapat dicek apakah sudah login atau belum.

Sebelum menggunakan session, kita harus menuliskan fungsi `session_start()`. Sedangkan untuk menghapus session, dapat dilakukan dengan fungsi `session_destroy()`.

```
1 <?php
2     session_start();
3
4     $_SESSION['username'] = "Administrator";
5     $_SESSION['password'] = "Rahasia";
6
7     echo "Session telah dibuat,
8         <a href='24_session_cek.php'> Cek Session </a>";
9 
```



Jika link Cek Session di atas diklik, akan menuju file lain yang menampilkan data session. Walaupun dalam file yang berbeda, nilai session dapat terbaca, berbeda dengan variabel yang hanya bisa digunakan dalam file yang sama. Berikut skrip file yang menggunakan session:

```
1 <?php
2     session_start();
3
4     if(!empty($_SESSION['username']) and !empty($_SESSION['password']))
5     {
6         echo "Username: " . $_SESSION['username'];
7         echo "<br> Password: " . $_SESSION['password'];
8         echo "<br> <a href='24_session_hapus.php'> Hapus Session </a>";
9     }
10    else
11    {
12        echo "Session telah dihapus";
13        echo "<br> <a href='24_session.php'>Buat Session</a>";
14    }
15
16 
```

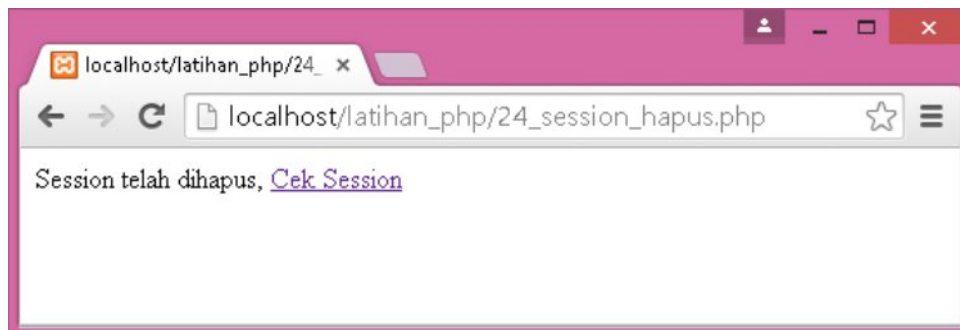


Jika link Hapus Session diklik, akan menuju file yang menghapus session. Berikut isi skrip file yang menghapus session:

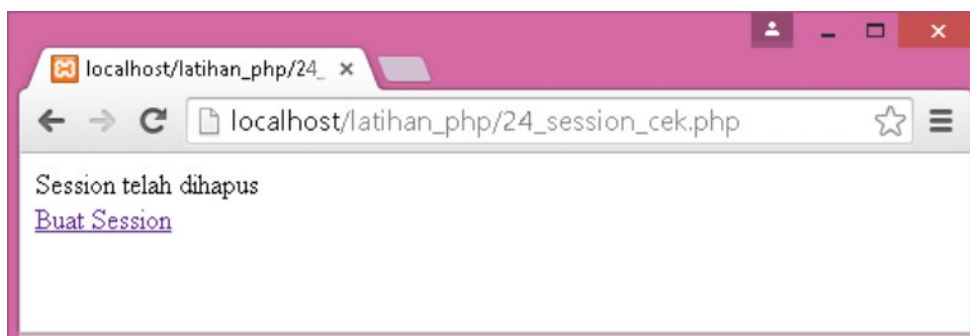
```

1 <?php
2     session_start();
3
4     session_destroy();
5
6     echo "Session telah dihapus,
7         <a href='24_session_cek.php'> Cek Session </a>";
8     ?>

```



Cek lagi session dengan mengklik link Cek Session.



## L. COOKIE

Hampir mirip dengan SESSION, namun data COOKIE akan terhapus setelah batas waktu yang telah ditentukan. Untuk membuat COOKIE caranya dengan menggunakan fungsi `setcookie(nama, nilai, batas akhir)`. Batas akhir COOKIE dihitung dengan satuan detik. Untuk mengecek COOKIE dengan skrip `$_COOKIE[nama_cookie]`.

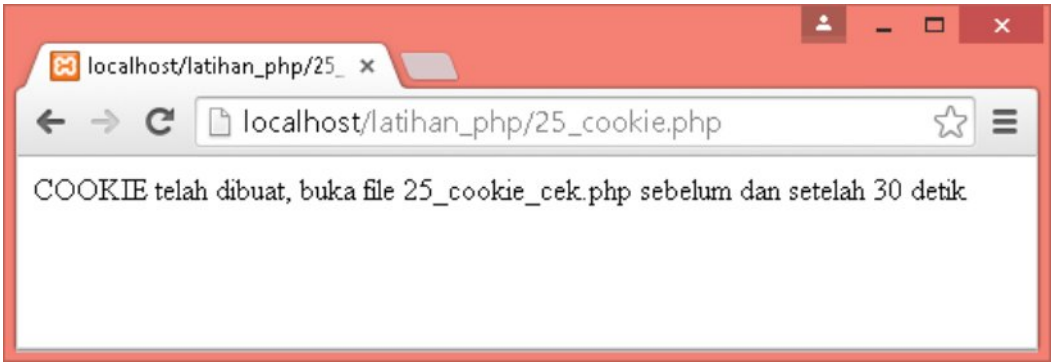
Kita akan buat 2 buah file, file `25_cookie.php` digunakan untuk membuat cookie dan `25_cookie_cek.php` digunakan untuk mengecek data cookie.

File `25_cookie.php`

```

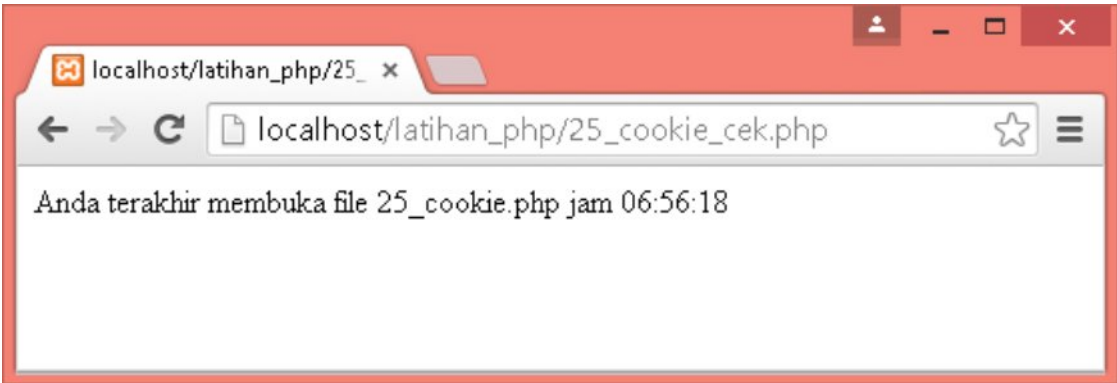
1 <?php
2     $batas = time() + 30;
3
4     setcookie("kunjunganTerakhir", date('H:i:s'), $batas);
5

```



File 25\_cookie\_cek.php

```
1 <?php
2     if(isset($_COOKIE['kunjunganTerakhir']))
3     {
4         echo "Anda terakhir membuka file 25_cookie.php
5             jam ".$_COOKIE['kunjunganTerakhir'];
6     }
7     else
8     {
9         echo "Anda terakhir membuka file 25_cookie.php
10            30 detik yang lalu";
11     }
12
13 ?>
```



Jika dibuka setelah 30 detik, hasilnya menjadi seperti berikut:



M. PHP DAN MYSQL

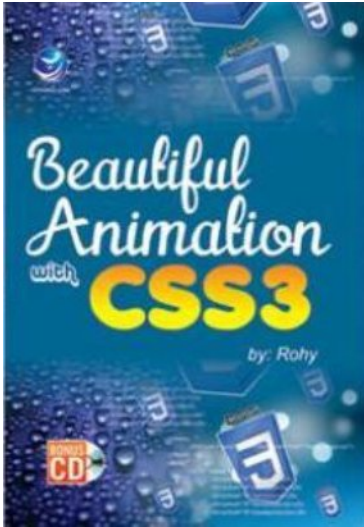
MySql merupakan database yang paling sering digunakan untuk menyimpan data website yang dibuat menggunakan PHP. Beberapa sintaks yang digunakan untuk memanipulasi database diantaranya sebagai berikut:

Sintaks	Keterangan
mysql_connect()	Untuk mengkoneksikan ke database. Isi dalam kurung secara berurutan adalah nama host, nama user mysql, password maysql.
mysql_select_db()	Untuk memilih database yang digunakan untuk menyimpan data. Tanda kurung diisi dengan nama database.
mysql_query()	Untuk menuliskan query mysql. Tanda kurung diisi dengan query mysql.
mysql_fetch_array()	Untuk mengambil data dari database, dan dijadikan array assosiatif. Tanda kurung diisi variabel yang menyimpan query mysql.
mysql_num_rows()	Untuk menghitung jumlah baris yang dihasilkan dari perintah mysql. Tanda kurung diisi variabel yang menyimpan query mysql.
mysql_error()	Untuk menampilkan pesan error yang dihasilkan oleh perintah msyql.

Contoh-contoh dari perintah-perintah di atas dapat langsung dilihat pada buku “Web Programming is Easy”.



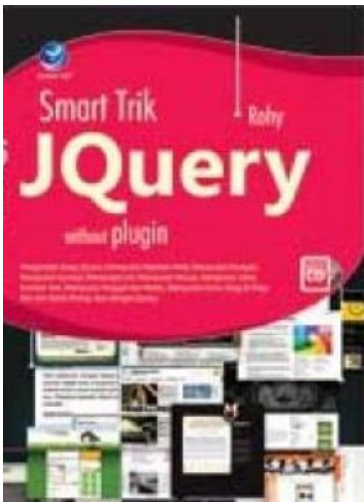
BUKU-BUKU KARYA PENULIS



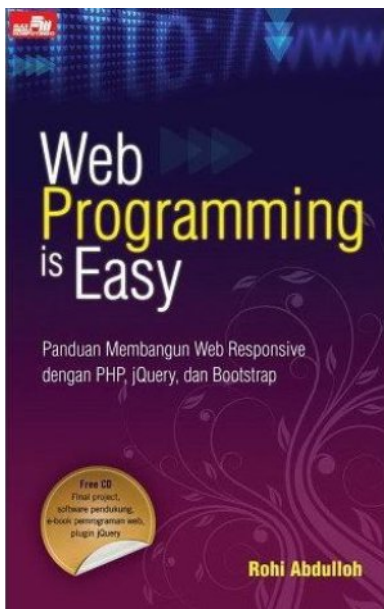
Judul : **Beatiful Animation with CSS3**  
Ukuran/Halaman : 16 x 23 cm / 122 halaman  
Penerbit : Andi Publisher  
Tahun Terbit : 2014  
Keterangan Singkat : Membahas 40+ tutorial animasi dengan CSS3 yang sering digunakan pada desain website ,diantaranya: manipulasi background, menu, tab, accordion, image viewer, image slider, social button, dsb.



Judul : **Be Creative with HTML5 Canvas**  
Ukuran/Halaman : 14 x 21 cm / 204 halaman  
Penerbit : Elex Media Komputindo  
Tahun Terbit : 2014  
Keterangan Singkat : Membahas penggunaan HTML5 Canvas mulai dari bentuk dasar, animasi, video, audio, menangani event, penerapan sifat-sifat fisika, hingga tutorial membuat jam analog, diagram (chart), aplikasi painting, graphic editor dan photo editor.



Judul : **Smart Trik JQuery without Plugin**  
Ukuran/Halaman : 14 x 21 cm / 248 halaman  
Penerbit : Andi Publisher  
Tahun Terbit : 2015  
Keterangan Singkat : Membahas 60+ tutorial jQuery tanpa menggunakan plugin, diantaranya : manipulasi gambar, tabel, tanggal & waktu, teks,



Judul : **Web Programming Is Easy**

Ukuran/Halaman : 14 x 21 cm / 224 halaman

Penerbit : Elex Media Komputindo

Tahun Terbit : 2015

Keterangan Singkat : Membahas panduan mudah membuat website bagi pemula mulai dari membuat halaman admin, halaman front-end, trik pengembangan website, menggunakan jQuery, desain responsive dengan Bootstrap, konversi ke mysql, hingga upload ke hosting.



Judul : **Super Dahsyat Responsive Web Design dengan Foundation 5**

Halaman : 192 halaman

Penerbit : Asfa Solution

Tahun Terbit : 2015

Keterangan Singkat : Membahas panduan membuat website responsive dengan foundation 5 mulai dari dasar-dasar foundation hingga penerapannya pada desain website baik halaman administrator maupun halaman pengunjung.



Judul : **Trik Rahasia Menjadi Master Wordpress Handal**

Halaman : 180 halaman

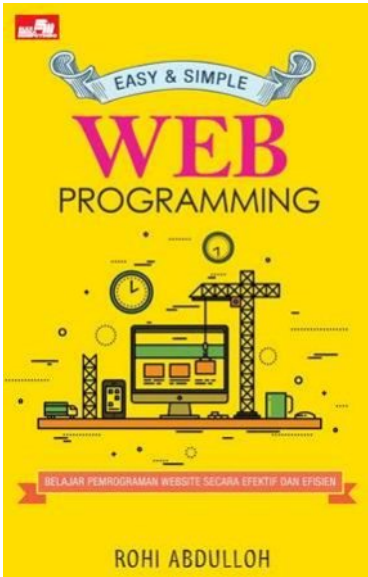
Penerbit : Asfa Solution

Tahun Terbit : 2015

Keterangan Singkat : Membahas pemrograman pada wordpress mulai dari dasar



widget, hingga membuat plugin sendiri.



Judul	: <b>Easy &amp; Simple Web Programming</b>
Halaman	: 228 halaman
Penerbit	: Elex Media Komputindo
Tahun Terbit	: 2016
Keterangan Singkat	: Edisi kedua dari buku "Web Programming Is Easy" dengan penambahan 2 bab yaitu mengatasi error pada PHP dan menggunakan widget dari internet seperti google map, komentar facebook, tweet pada twitter, statistik pengunjung, polling, dan sebagainya.